

Mihaela Giurgulescu
Valeriu Benedicth Giurgulescu



Informatică și TIC

Clasa a VIII-a



Acest manual este proprietatea Ministerului Educației și Cercetării.

Acest manual școlar este realizat în conformitate cu Programa școlară aprobată prin Ordinul ministrului educației naționale nr. 3393/28.02.2017.

119 – număr unic de telefon la nivel național pentru cazurile de abuz împotriva copiilor

116.111 – numărul de telefon de asistență pentru copii

Mihaela Giurgulescu
Valeriu Benedicth Giurgulescu



Informatică și TIC

Clasa a VIII-a



Manualul școlar a fost aprobat de Ministerul Educației și Cercetării prin ordinul de ministru nr. 4500/10.07.2025.
Manualul este distribuit elevilor în mod gratuit, atât în format tipărit, cât și în format digital, și este transmisibil timp de patru ani școlari, începând din anul școlar 2025-2026.

Inspectoratul Școlar
Școala/ Colegiul/ Liceul

ACEST MANUAL A FOST FOLOSIT DE:

Anul	Numele elevului	Clasa	Anul școlar	Aspectul manualului*			
				format tipărit		format digital	
				la primire	la predare	la primire	la predare
1							
2							
3							
4							

- * Pentru precizarea aspectului manualului se va folosi unul dintre următorii termeni: **nou, bun, îngrijit, neîngrijit, deteriorat.**
- Cadrele didactice vor verifica dacă informațiile înscrise în tabelul de mai sus sunt corecte.
 - Elevii nu vor face niciun fel de însemnări pe manual.

Referenți științifici:
Lector univ. dr. Alexandra Băicoianu, Facultatea de Matematică și Informatică, Universitatea „Transilvania” din Brașov
Prof. grd. I Simona Mihaela Popa, Colegiul Național „Gheorghe Lazăr”, București

Coordonator redacție: Cătălin Georgescu
Editor-coordonator: Ionuț Popa
Redactare: Diana Onulescu
Corectură: Elena Chiruță
Copertă și tehnoredactare: Florin Paraschiv
Layout: Faber Studio
Înregistrări și procesare sunet, activități digitale interactive și platformă e-learning: Infomedia Pro;
website: <https://infomediapro.ro>
Credite foto: Dreamstime

ISBN 978-630-358-013-5

Pentru comenzi puteți contacta Departamentul Difuzare
C.P. 12, O.P. 63, cod 014880, sector 1, București
Tel.: 021 796 73 83, 021 796 73 80
Fax: 021 369 31 99
www.art-educational.ro

Toate drepturile asupra acestei lucrări sunt rezervate Editurii Art Klett.
Nicio parte a acestei lucrări nu poate fi reprodusă, stocată ori transmisă, sub nicio formă (electronic, mecanic, fotocopiare, înregistrare sau altfel), fără acordul prealabil scris al editurii.

Cuvânt-înainte

Manualul este conceput din dorința de a vă ajuta să folosiți cât mai bine tehnologia, pentru a vă bucura din plin de beneficiile acesteia, și cu speranța că va reprezenta pentru voi un material de calitate, frumos și ușor de utilizat.

Lucrarea este structurată în trei părți: prima parte conține teme legate de utilizarea programelor de calcul tabelar, cu detalierea programului MS Excel, a doua parte este dedicată realizării de site-uri web, cu evidențierea modalității de lucru colaborativ, iar ultima parte este dedicată realizării de algoritmi și programe informatice pentru lucrul cu șirurile de valori numerice, utilizând limbajul de programare C++ și mediul de dezvoltare integrat Code::Blocks. Nu lipsește nici realizarea unor simulări în mediul Open Roberta Lab, utilizând un robot didactic virtual, NXT.

Lecțiile conțin aplicații practice care favorizează învățarea prin descoperire. De asemenea, exercițiile propuse la fiecare lecție conțin exemple de rezolvare, pentru o înțelegere mai ușoară a enunțului și se rezolvă pe baza informațiilor transmise în manual. Pe parcursul lucrării, la finalul fiecărei părți sunt prevăzute activități de recapitulare, pentru a consolida cunoștințele acumulate, și de autoevaluare, pentru ca fiecare dintre voi să poată să-și aprecieze gradul de înțelegere a noțiunilor. Există și o recapitulare finală care conține itemi corespunzători noțiunilor învățate pe parcursul întregului an școlar. Toate exercițiile cuprinse în autoevaluări și în recapitularea finală sunt rezolvate la sfârșitul manualului.

Și pentru că sunteți într-un an important pentru voi, anul tranziției spre liceu, vă îndemnăm să țintiți cât mai sus în tot ceea ce faceți, pentru că „nu vei zări niciodată un curcubeu dacă te uiți întotdeauna în jos” (Charlie Chaplin).

Autorii

Prezentarea manualului

Varianta digitală a manualului este similară cu cea tipărită, având în plus peste 90 de AMII, activități multimedia interactive de învățare, cu rolul de a spori valoarea cognitivă. Activitățile multimedia interactive de învățare sunt de trei feluri și sunt simbolizate pe parcursul manualului astfel:

 **Activitate statică**, de ascultare activă și de observare dirijată a unei imagini semnificative

 **Activitate animată**, filmuleț sau scurtă animație

 **Activitate interactivă**, de tip exercițiu sau joc, în urma căreia elevul are feedback imediat

Alte butoane folosite în varianta digitală:

 Cuprins manual

 Fereastră de afișare a rezultatelor

 Ajutor general manual

 Micșorarea/mărirea imaginii

 Salt la începutul/finalul manualului

 Pagină înainte/pagină înapoi

 Mărire pe tot ecranul

 Schimbarea modului de afișare

 Includerea unei notițe

 Introducerea de marcaje colorate

 Informații despre utilizator

Ce propune acest manual

Manualul propune ghidarea elevilor, asemenea unui tutorial interactiv și prietenos, în lumea fascinantă a tehnologiei informației și comunicațiilor.

Acesta este structurat în trei mari unități de învățare și răspunde viziunii și cerințelor noului curriculum. Prima unitate acoperă integral, în cele 6 lecții existente, domeniul de conținut din programa școlară *Calcul tabelar*. Urmează a doua unitate, structurată în 4 lecții, ce prezintă în detaliu conținutul despre *Pagini web*, în timp ce ultima unitate dezvoltă, în 12 lecții, conținutul despre *Algoritmi și programare*. Nu puteau lipsi recapitulările sumative și autoevaluările, marcate distinct pentru fiecare unitate, precum și o recapitulare finală, urmată de răspunsuri la multe dintre problemele și exercițiile propuse. Toate unitățile manualului oferă numeroase activități interdisciplinare însoțite de indicații și modele de rezolvare. Sunt propuse, de asemenea, și metode complementare de evaluare: *observarea sistematică a activității și comportamentului elevilor* (vezi *Fișă de autoevaluare a activității și comportamentului*), *investigația*, *proiectul*, *portofoliul*, *autoevaluarea*.

Specificul disciplinei implică utilizarea unor metode didactice activ-participative care sunt bogat sugerate de activitățile prezentate în acest manual, astfel încât să fie atinse toate cele trei competențe generale vizate.

Manualul este structurat în 3 unități de învățare



Structura unităților de învățare

Lecții de predare-învățare

Modul de abordare modern facilitează învățarea și asigură, totodată, stimularea creativității elevilor care vor folosi cunoștințele dobândite în realizarea unor miniproiecte, atât individuale, cât și colaborative, pe teme de interes caracteristice vârstei lor (respectând creditarea informației și drepturile de autor). Principalele rubrici ale lecțiilor sunt marcate diferit și consecvent pe parcursul manualului.

Descoperiți prezintă, într-o manieră plăcută și graduală, introducerea conținutului nou de învățat. Sunt oferite numeroase exemple, iar conținuturile noi sunt prezentate cu suportul multor imagini reprezentative, ce sunt explicate în detaliu, unde este cazul.

Exersați propune itemi ce vor consolida învățarea, din care nu lipsesc temele de portofoliu, dezbaterile în clasă, lucrul în echipe sau individual, activitățile de documentare/ investigare și realizarea unor eseuri scurte sau a unor sarcini interesante.

Amintiți-vă și Rețineți sunt alte rubrici care apar pe parcursul lecțiilor, atunci când conținutul prezentat necesită reamintirea unor noțiuni sau când se evidențiază conținutul important, de bază pentru achizițiile viitoare.

Aplicați și investigați asigură aplicarea noțiunilor noi, elevii fiind implicați, astfel, în mod activ în procesul de învățare. Investigațiile contribuie la evaluarea nivelului de înțelegere, dezvoltând gândirea critică și transformând cunoștințele în instrumente de gândire activă.

Știați că...? prezintă unele informații sau curiozități despre aplicațiile descrise în manual. Astfel, cei interesați să afle mai multe o pot face apelând la cuvintele-cheie din lecții.

Portofoliu

Pe parcursul lecțiilor sunt recomandate mai multe teme de **dezbateri** și pentru **portofoliul** elevilor. Metode moderne de învățare-evaluare, acestea sunt prezentate în cadrul rubricii **Exersați**. Proiectele au atât caracter informativ, cât și formativ. Se propun teme pentru documentări și investigații, manualul oferind o serie generoasă de exemple despre cum putem folosi diverse aplicații online sau instalate pe propriile calculatoare.

Recapitulare

La fiecare final de capitol, înainte de testul de autoevaluare aferent, se află câte un test de recapitulare. Probele concepute conțin selecții de itemi obiectivi, semiobiectivi și subiectivi, cu nivel de la simplu la complex. Recapitularea vizează atât corpul de noțiuni învățate, cât și capacitățile asociative, de interpretare și aplicare a cunoștințelor în contexte noi.

Autoevaluare

Autoevaluările sunt însoțite de punctajele de notare și de răspunsuri care se află la finalul manualului, elevii putând astfel să se verifice singuri.

Manualul propune și o recapitulare finală, din toată materia parcursă de-a lungul anului școlar. Exercițiile și problemele propuse sunt astfel concepute încât să asigure o bună verificare a cunoștințelor acumulate, precum și operaționalizarea acestora în situații noi, variate. Și aici, elevii au la dispoziție, la sfârșitul manualului, răspunsurile la exercițiile propuse, pentru a se putea autoevalua cu ușurință, obținând un feedback imediat.

Amintiți-vă Amintiți-vă de robotul virtual NCT pe care l-ați folosit pentru a crea jocul, în stilul Scratch. Ne amintim că programul robotului se numește NEPO.

Descoperiți Descoperiți noțiunile noi de programare și de robotica. Robotul NCT, robotul educațional realizat de compania LEGO, este un robot virtual care poate fi controlat prin intermediul unui program de programare.

Aplicați și investigați Aplicați și investigați noțiunile noi de programare și de robotica. Robotul NCT, robotul educațional realizat de compania LEGO, este un robot virtual care poate fi controlat prin intermediul unui program de programare.

Exersați Exersați noțiunile noi de programare și de robotica. Robotul NCT, robotul educațional realizat de compania LEGO, este un robot virtual care poate fi controlat prin intermediul unui program de programare.

Portofoliu Portofoliu este o colecție de lucrări realizate de elevi, care pot fi prezentate în clasă sau în concursuri.

Utilizarea algoritmilor în robotica

Program robotului

Algoritm de funcționare

Exersați

Portofoliu

Formarea unei rețele de calcul. Tipuri de date. Formule

Rețineți

Știați că...?

Aplicați și investigați

Exersați

Portofoliu

Cuprins și competențe generale și specifice

	Pag.	Lecții
	9	Să ne amintim! Scurtă recapitulare a cunoștințelor de până acum
UNITATEA 1 Calcul tabelar	12	L1: Utilizarea foilor de calcul tabelar
	14	L2: MS Excel – elemente de interfață
	18	L3: Operații principale cu un registru de calcul
	21	L4: Formatarea unui registru de calcul. Tipuri de date. Formule
	26	L5: Modalități de utilizare a registrelor de calcul. Calcul tabelare, funcții
	29	L6: Modalități grafice de utilizare a registrelor de calcul
	32	Recapitulare
	33	Autoevaluare
UNITATEA 2 Pagini web	36	L1: Pagini web – exemple, analiză, structură
	40	L2: Aplicații specializate pentru crearea paginilor web – elemente de interfață și facilități
	45	L3: Crearea unei pagini web. Publicarea unui site de o pagină
	51	L4: Construirea unui site Internet cu mai multe pagini. Lucrul colaborativ
	54	Recapitulare
	55	Autoevaluare
UNITATEA 3 Algoritmi	58	L1: Structura decizională (alternativă) – recapitulare
	62	L2: Structura repetitivă – recapitulare
	66	L3: Utilizarea algoritmilor în robotică
	68	L4: Șiruri de valori. Generarea șirurilor de numere cu proprietăți simple
	70	L5: Generarea șirurilor de numere pe baza unor reguli
	72	L6: Operații cu șiruri de valori: citire, parcurgere, afișare
	74	L7: Verificarea unor proprietăți ale elementelor șirului de valori
	76	L8: Numărarea elementelor care îndeplinesc o condiție
	78	L9: Verificarea îndeplinirii unor proprietăți de către toate elementele unui șir
	81	L10: Căutarea unui element în șir de valori. Verificarea existenței unui anume element
	83	L11: Minimul și maximul dintr-un șir de valori
	86	L12: Suma și produsul elementelor unui șir de valori
	88	Recapitulare
	89	Autoevaluare
	90	Proiect: Liceul preferat
	91	Recapitulare finală
	94	Răspunsuri (selectiv)

Competențe specifice

1.1, 3.1

1.2, 3.2

2.1, 2.2, 3.3

1.1, 1.2, 2.1, 2.2, 3.1, 3.2, 3.3

Competențe generale și specifice

1. Utilizarea responsabilă și eficientă a tehnologiei informației și comunicațiilor
 - 1.1. Utilizarea foilor de calcul tabelar în vederea rezolvării unor situații problemă simple
 - 1.2. Utilizarea unui editor dedicat pentru realizarea unor pagini web cu diverse teme
2. Rezolvarea unor probleme elementare prin metode intuitive de prelucrare a informației
 - 2.1. Identificarea șirurilor de valori în diferite contexte de prelucrare în vederea construirii algoritmilor
 - 2.2. Rezolvarea unor probleme simple prin construirea unor algoritmi de prelucrare a șirurilor de valori
3. Elaborarea creativă de miniproiecte care vizează aspecte sociale, culturale și personale, respectând creditarea informației și drepturile de autor
 - 3.1. Elaborarea de produse informatice utilizând aplicații de calcul tabelar
 - 3.2. Elaborarea/actualizarea de pagini web conform unor specificații date
 - 3.3. Implementarea algoritmilor într-un mediu de programare



Salutare, drag prieten! Bine te-am regăsit în acest nou an școlar!

Parcurgând manualul, vei putea realiza un **portofoliu** propriu, o „carte de vizită” pentru disciplina **Informatică și TIC**. Portofoliul va fi evaluat la finalizarea fiecărei unități de învățare a manualului. Fiecare material din portofoliu va avea specificată data realizării sale.

Portofoliul tău va cuprinde:

- materialele indicate la fiecare lecție ca făcând parte din portofoliu;
- proiectele realizate;
- investigațiile realizate;
- fișele de autoevaluare a activității și comportamentului;
- alte materiale indicate de profesorul tău;
- alte materiale pe care dorești să le adaugi, legate de noțiunile învățate.

Scopul realizării portofoliului:

- va reflecta munca depusă de tine;
- va confirma faptul că ai parcurs noțiunile și că ai acumulat cunoștințele necesare;
- evaluarea sa va arăta ceea ce este foarte bine realizat și la ce anume mai trebuie lucrat.

Evaluarea portofoliului se va face, în primul rând, de către tine și de profesorul tău, apoi va fi evaluat și apreciat de către părinți sau alte persoane.

Fișă de autoevaluare a portofoliului	Da	Nu
Am adăugat la portofoliu toate materialele indicate.		
Am rezolvat toate sarcinile în întregime.		
Am întâmpinat dificultăți în realizarea anumitor exerciții.		
Am cerut ajutorul colegilor sau prietenilor pentru a rezolva sarcinile de lucru.		
Am cerut ajutorul cadrului didactic în rezolvarea sarcinilor.		

Fișele de autoevaluare a activității și comportamentului se vor completa la finalul fiecărei unități de învățare și se vor atașa portofoliului tău.

Să ne amintim! Scurtă recapitulare a cunoștințelor de până acum



Amintiți-vă

Principalele operații în lucrul cu fișierele și folderele sunt: creare, deschidere, editare, salvare, închidere, selecție, copiere, mutare, ștergere.

Pentru lucrul cu elementele conținute în fișiere, principalele operații sunt: selectare, copiere, mutare, ștergere.

Comenzile corespunzătoare operațiilor de mai sus se găsesc, în general, la majoritatea programelor, sub formă de opțiuni de meniu, în meniurile **File** (*Fișier*) și **Edit** (*Editare*) sau **Home** (*Acasă* – cu sensul de „fila de bază”).

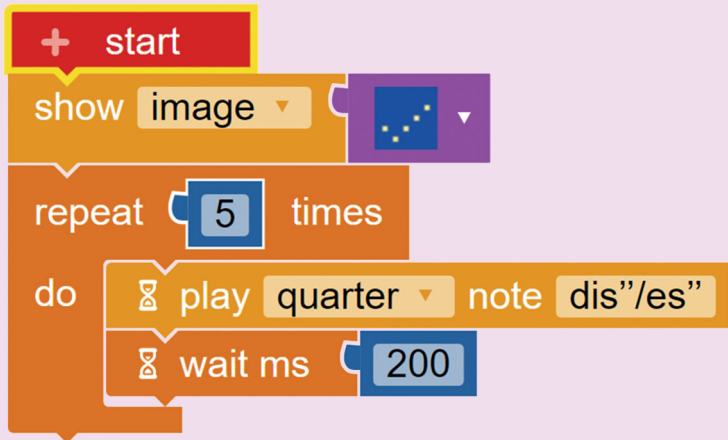
Pentru lucrul mai eficient, puteți folosi comenzile rapide (combinațiile de taste) următoare (valabile atât în editoarele de text, de exemplu, în MS Word, cât și în programele de editare de sunete, în programele de editare video și în editorul de C++):

- **Creare** (*New*) – **Ctrl+N**;
- **Deschidere** (*Open*) – **Ctrl+O**;
- **Salvare** (*Save*) – **Ctrl+S**;
- **Închidere** (*Close* sau *Exit*) – **Alt+F4**;
- **Selectare integrală** (*Select All*) – **Ctrl+A**;
- **Copiere** (*Copy*) – **Ctrl+C**;
- **Mutare** – aceasta este o combinație de două acțiuni: **Decupare** din locația inițială (*Cut*) – **Ctrl+X** –, și **Lipire** în noua locație (*Paste*) – **Ctrl+V**;
- **Ștergere** (*Delete*) – tasta **DEL** sau **Delete** (în funcție de producător, pe tasta respectivă marcajul poate să difere);
- **Anulează** (*Undo*) – **Ctrl+Z** (pentru a reveni la starea dinaintea ultimei acțiuni efectuate).



Exersați

Sisteme de operare	Dați trei exemple de sisteme de operare foarte utilizate în prezent.
Realizarea unui document electronic	Realizați în MS Word o scurtă prezentare electronică, pe tema „Cinci lucruri importante pe care le-am învățat la Informatică și TIC în clasa a VII-a”. Folosiți în prezentare texte și imagini. Titlul să fie centrat și îngroșat.
Siguranța pe Internet	Precizați trei principii de bază care vă asigură o navigare pe Internet lipsită de pericole.
Poșta electronică	Care sunt cele două categorii de programe prin intermediul cărora puteți utiliza poșta electronică?
Reguli de comunicare online	<i>Neticheta</i> impune norme de comportament general în mediul online. Enumerați cinci reguli specifice comunicării în acest mediu.
Operații cu sunete	Care sunt operațiunile de bază pentru a obține un material audio de bună calitate (de exemplu, un interviu de două minute luat unui coleg)?
Aplicații video	Numiți două aplicații dedicate prelucrărilor video.
	Realizați un videoclip cu durata între 30 de secunde și 3 minute, pe o temă ecologică sau educațională (de exemplu, „Parcurile din jurul nostru îngrijite de elevii de la școlile din localitate”), după următorul scenariu: ● să fie o succesiune de secvențe filmate și/sau fotografii pe tema aleasă (se pot folosi inclusiv fotografii ale panourilor expuse în școli, panouri realizate de copii pentru proiectele la care au participat); ● dacă se folosesc fotografii, acestea să fie în număr de cel puțin zece;

	• dacă sunt folosite fotografii sau secvențe filmate luate de pe Internet, se vor folosi numai din domeniul public, liber de drepturi, sau cu acordul scris al autorilor; • să se pună mare, pe ecranul de început, titlul videoclipului; • să se pună efecte de tranziții între cadre; • să se adauge un fundal muzical adecvat, care să se întindă pe durata întregului videoclip. Videoclipul să fie exportat la final în format .mp4.
Aplicații colaborative	Lucru în echipă: Realizați, la nivelul clasei, o carte electronică cu titlul „Povești, povestiri, amintiri”. Parcurgând toate acțiunile următoare, veți obține un minunat proiect realizat împreună: • Mai întâi, alegeți un „manager de proiect”. El va iniția fișierul Word în care cu toții veți introduce câte ceva pentru realizarea cărții. • Salvați fișierul pe Google Drive, cu titlul „Povești, povestiri, amintiri.doc”. • Faceți ceea ce este necesar pentru ca toți elevii clasei să aibă acces la fișierul care va conține cartea: • Asigurați-vă că aveți câte un cont Google pentru fiecare. • Managerul de proiect va da tuturor drept de acces la acel fișier. • Scrieți, fiecare dintre voi, în documentul creat, o poveste, o povestire, o amintire care vi se pare interesantă sau hazlie și merită împărtășită. Puteți adăuga imagini și formata textul așa cum doriți.
GIS	Ce este și la ce se folosește un sistem GIS? Dați un exemplu de utilizare practică în viața de zi cu zi. Plimbați-vă prin zona din jurul casei (pe o rază de 1 km), pe jos sau cu bicicleta, și adăugați pe Google Maps toate cuiburile de barză pe care le găsiți sau orice alte obiective doriți. Așteptați să fie avizate și să devină disponibile toate elementele adăugate, apoi formați grupuri de lucru și analizați ce elemente a adăugat fiecare, precizia localizării și multitudinea amănuntelor din descriere, precum și eleganța descrierii.
Programarea roboților	Explicați ce face Calliope mini în mediul de programare Open Roberta Lab, dacă primește următorul program:  <pre> graph TD Start([+ start]) --> ShowImage[show image] ShowImage --> Repeat[repeat 5 times] Repeat --> Do[do] Do --> PlayNote[play quarter note "dis"/es"] PlayNote --> WaitMs[wait ms 200] WaitMs --> Repeat </pre>
Algoritmi și programare	Realizați în C++ programul care afișează dublul celei mai mici cifre a unui număr întreg introdus de la tastatură. <i>Indicație: Intrare: 345, Ieșire: 6; Intrare: -57, Ieșire: 10.</i> Faceți un program în C++ care calculează și afișează aria și perimetrul unui triunghi echilateral cu latura n, unde n este număr natural mai mic decât 200.

U1

Calcul tabelar



Lecția 1	12-13	Utilizarea foilor de calcul tabelar
Lecția 2	14-17	MS Excel – elemente de interfață
Lecția 3	18-20	Operații principale cu un registru de calcul
Lecția 4	21-25	Formatarea unui registru de calcul. Tipuri de date. Formule
Lecția 5	26-28	Modalități de utilizare a registrelor de calcul. Calcule tabelare, funcții
Lecția 6	29-31	Modalități grafice de utilizare a registrelor de calcul
Recapitulare	32	
Autoevaluare	33	
Concurs	34	Planificarea unui eveniment

1 Utilizarea foilor de calcul tabelar



Descoperiți

▶ **Foile de calcul electronice** reprezintă o modalitate de gestionare a datelor organizate sub forma unor tabele digitale. Numele de **foaie de calcul** provine din contabilitate, domeniu în care se folosesc formulare de hârtie de mari dimensiuni, liniate orizontal și vertical sub formă de tabel.

Utilizarea calculatorului a adus mai multe avantaje în calculul tabelar: a permis construirea ușoară a tabelului (inserare/ștergere de rânduri și coloane), adăugarea la căsuțele din tabele, pe lângă valorile înscrise în ele, a formulor de calcul pentru aceste valori, a specificațiilor de afișare a valorilor etc.

Exemplu practic: faceți un tabel cu materiile voastre de la școală și cu notele dintr-un an școlar. Pentru exemplul de mai jos vom considera 15 materii, câte 5 note la fiecare materie. Vom calcula media pentru fiecare materie (într-o coloană separată) și media generală la final.

Dacă am dori să vedem ce impact ar avea asupra mediei generale o notă mai mare la geografie (de exemplu, 9 în loc de 7), ar trebui să recalculăm media la geografie, apoi să recalculăm media generală.

Imaginați-vă acum că media pentru fiecare materie este calculată automat de calculator, în baza unei formule pe care am introdus-o acolo unde trebuie să apară media, iar pentru media generală se procedează la fel (se calculează automat în baza unei formule). Dacă am schimba nota 7 din tabelul nostru cu nota 9, s-ar modifica imediat și automat atât media la geografie, cât și media generală. Iar noi am vedea rapid ce diferență ar aduce la media generală o notă de 9 în locul notei de 7, fără să mai facem calcule. Această posibilitate este creată, alături de multe altele, de apariția foilor de calcul electronice și a programelor dedicate pentru lucrul cu ele.

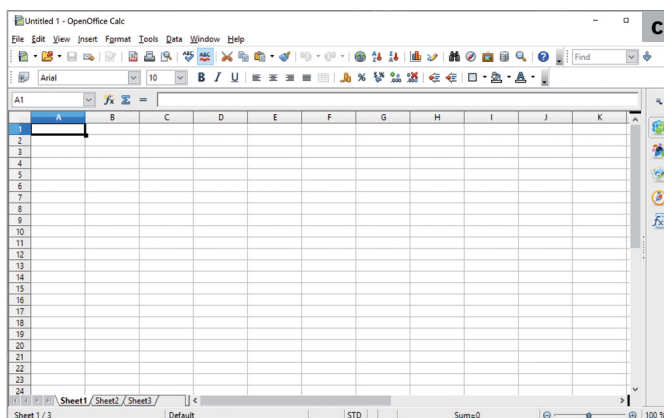
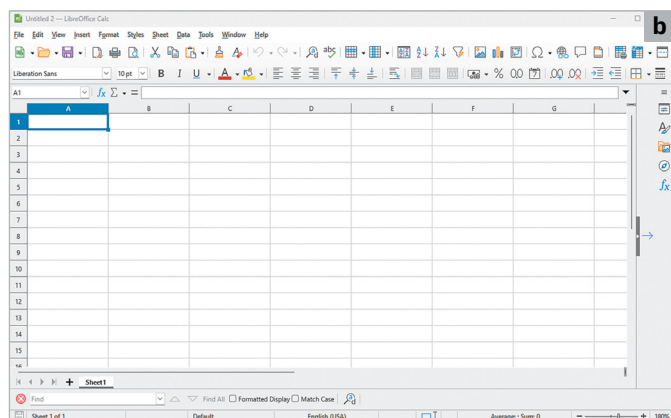
	A	B	C	D	E	F	G
1	MATERIA	NOTA 1	NOTA 2	NOTA 3	NOTA 4	NOTA 5	MEDIA
2	Matematică	10	10	10	9	10	9.80
3	Română	10	10	10	10	10	10.00
4	Istorie	10	10	10	10	10	10.00
5	Fizică	9	9	9	9	9	9.00
6	Desen	10	10	10	10	10	10.00
7	Muzică	10	10	10	10	10	10.00
8	Sport	10	10	10	10	10	10.00
9	Geografie	10	10	7	10	10	9.40
10	Religie	10	10	10	10	10	10.00
11	Limba engleză	10	10	10	10	10	10.00
12	Limba germană	10	10	10	10	10	10.00
13	Chimie	10	10	10	10	10	10.00
14	TIC	10	10	10	10	10	10.00
15	Opțional informatică	10	10	10	10	10	10.00
16	Biologie	9	9	9	9	9	9.00
17							
18						MEDIA GENERALĂ:	9.81

Ce observați la tabelul de mai sus (*imaginea a*)? Pe două dintre margini regăsim niște marcare ca la șah: pe o latură, numere și pe o latură perpendiculară, litere. Acestea ne folosesc atunci când vrem să ne referim sau să ne poziționăm pe o anumită căsuță din tabel. De exemplu, căsuța în care este scris „TIC” este **A14** (intersecția dintre coloana **A** și rândul **14** din tabel), iar cea în care e calculată media generală este **G18**.

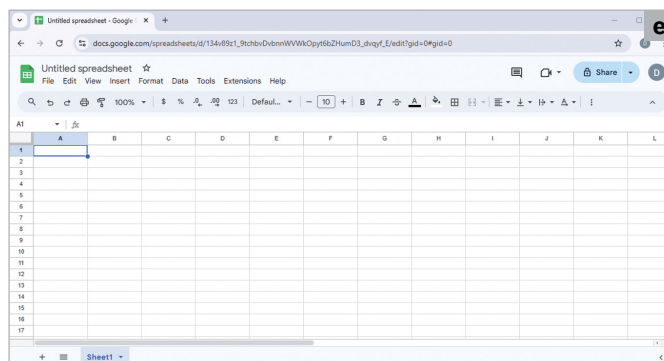
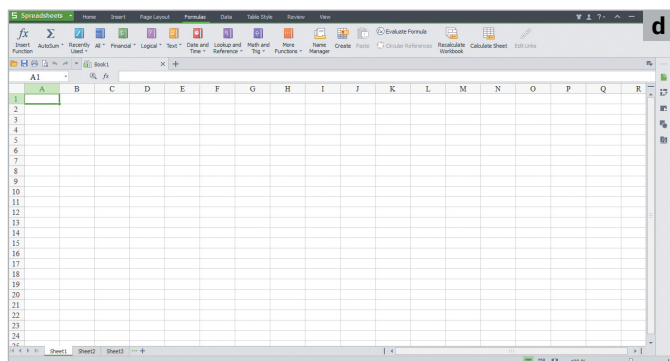
Există o multitudine de programe dedicate lucrului cu foi de calcul, cunoscute și sub numele de **programe de calcul tabelar**. Amintim aici doar câteva dintre cele mai populare:

- **MS Excel** (din pachetul de programe Microsoft Office) sau **Excel Online**
- **WPS Spreadsheets** (din pachetul de programe WPS Office) (*imaginea d*)
- **LibreOffice Calc** (din pachetul de programe LibreOffice) (*imaginea b*)
- **Ashampoo PlanMaker** (din pachetul de programe Ashampoo Office)
- **OpenOffice Calc** (din pachetul de programe Apache OpenOffice) (*imaginea c*)
- **Google Sheets** (Foi de calcul Google) (*imaginea e*)

Fiecare dintre programele specializate în lucrul cu foi de calcul are o serie de particularități care pot fi avantaje sau dezavantaje, în funcție de fiecare dintre noi: **MS Excel** face parte din pachetul de programe MS Office și este, poate, cel



Utilizarea foilor de calcul tabelar



mai popular, astfel că putem găsi cu ușurință ajutor la cei din jurul nostru, fiind disponibil și în limba română; **Calc** (din pachetul LibreOffice) și **Spreadsheets** (din pachetul WPS Office) sunt aplicații gratuite, la îndemâna oricui, și sunt disponibile și în limba română; unele se instalează pe calculator și le aveți la îndemână oricând, indiferent dacă aveți legătură la Internet sau nu, altele sunt disponibile online, nu necesită instalare și nu ocupă spațiu de stocare pe calculatoarele voastre, dar necesită legătură la Internet, altfel nu pot fi accesate.

Știați că...?



În aprilie 1975, Bill Gates împreună cu Paul Allen au înființat compania Microsoft, care a dezvoltat atât cunoscutul sistem de operare Microsoft Windows, cât și pachetul de programe Microsoft Office, din care fac parte Word, PowerPoint, dar și Excel.



Aplicați și investigați

► Căutați pe Internet informații despre cele șase programe menționate anterior. Faceți în caiete un tabel în care să aveți câte o coloană pentru fiecare program și câte o linie în care să analizați aspectele următoare:

1. din ce pachet de programe face parte;
2. este gratuit sau contra cost (dacă utilizarea presupune costuri, treceți prețul în tabel);
3. pentru ce sisteme de operare este disponibil;
4. este disponibil sau nu și în limba română;
5. se instalează sau e disponibil online;
6. acceptă lucrul colaborativ (pot lucra mai multe persoane online pe același fișier);
7. ați auzit de el de la prieteni, colegi sau părinți;
8. ați mai folosit alte programe din același pachet de programe;
9. există versiunea programului și pentru mobile.

Priviți comparativ caracteristicile programelor. Care dintre ele credeți că sunt mai importante? Ce program vi s-ar părea mai potrivit pentru voi?



Exersați

1. Dați exemple de trei programe pentru calcul tabelar.
2. Argumentați de ce este ușor să facem modificări într-o foaie de calcul realizată cu ajutorul unui program specializat.
3. Scrieți, în caiete, **A** (Adevărat) sau **F** (Fals), în funcție de valoarea de adevăr a propozițiilor:
 - a. Un program de calcul tabelar produce fișiere video. **A F**
 - b. Un tabel realizat cu ajutorul unui program de calcul tabelar poate fi modificat ușor și tipărit de nenumărate ori. **A F**
 - c. Într-un program de calcul tabelar se pot introduce și formule pentru a efectua automat anumite calcule. **A F**
4. Care sunt avantajele utilizării foilor de calcul pe calculator? Argumentați.



Portofoliu



Realizați în caiete un tabel cu materiile din clasa a VIII-a, cu media anuală la fiecare materie și cu media generală. Cronometrați-vă și notați timpul necesar.

2 MS Excel – elemente de interfață



Descoperiți

Dintre aplicațiile dedicate lucrului cu foile de calcul vom folosi **Microsoft Office Excel** (prescurtat **MS Excel** sau, simplu, **Excel**), componentă a pachetului de programe Microsoft Office. Un fișier lucrat în această aplicație poate avea una dintre următoarele extensii: **.xls**, **.xlsx**, **.xlsm**. În continuare, pentru exemplificări, va fi utilizată versiunea Microsoft Excel din suita de programe Microsoft 365 Personal, instalată în limba română pe un calculator cu sistem de operare Windows 11.



Mai departe, vom descoperi împreună cele mai importante elemente ale interfeței aplicației MS Excel.

Pornirea aplicației Microsoft Excel

Lansarea în execuție a aplicației se face accesând **Start -> All -> Excel** sau se tastează cuvântul „Excel” în căsuța de căutare (*Search*) de pe **Bara de activități** (*Toolbar*) a interfeței Windows-ului și se apasă **Enter**.

Aplicația MS Excel se deschide cu un ecran de start. De aici putem alege:

- crearea unui document nou, necompletat: se alege **Nou** (*New*), **Registru de lucru necompletat** (*Blank workbook*);
- crearea unui document nou, plecând de la un șablon predefinit: se alege unul dintre șabloanele existente;
- deschiderea unui registru de calcul realizat anterior: se selectează un fișier din lista propusă (cu cele mai recent accesate) sau se caută, în cloud (în contul personal OneDrive) sau pe PC, fișierul dorit.

Structura unui registru de calcul (foaie de calcul, coloană, rând, celulă, adresă de celulă)

Aplicația Microsoft Excel organizează datele în **foi de calcul**, denumite și **foi de lucru**. O **foaie de calcul** (*worksheet*) reprezintă un singur tabel cu mai multe rânduri (linii) etichetate cu numere (**1, 2, ...**) și coloane etichetate cu litere (**A, B, ...**). Numărul de linii și de coloane ale unei foi de calcul se mărește dinamic până la numărul maxim de 1 048 576 de rânduri și 16 384 de coloane.

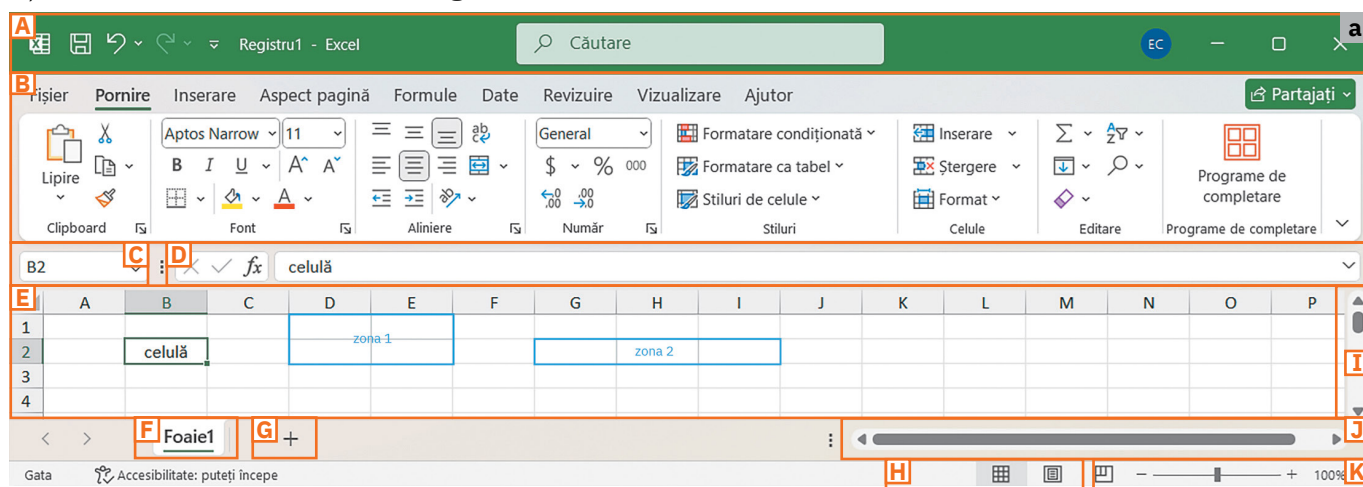
Unitatea elementară adresabilă din tabel este **celula**, aflată la intersecția unui rând și a unei coloane din tabel. În celule se introduc de la tastatură date și formule de calcul pentru obținerea de rezultate. O celulă este accesată prin eticheta coloanei și a rândului la a căror intersecție se află. De exemplu, celula **B2** este celula de la intersecția coloanei **B** și a rândului 2 din tabel (*imaginea a*).

Mai multe celule adiacente, cu coloane sau rânduri alăturate, formează o **zonă**. Celulele din acea zonă se prelucrează simultan, în același mod. O zonă se adresează prin coordonatele celulelor din colțurile sale stânga-sus și dreapta-jos, despărțite prin „:”. De exemplu, în imaginea de mai jos, zona 1 are coordonatele **D1:E2**, zona 2, **G2:I2**.

Mai multe foi de calcul formează împreună un **registru de calcul**, denumit și **registru de lucru** (*workbook*).

Detalierea elementelor de interfață ale aplicației MS Excel

Vom alege să creăm un fișier nou, necompletat. Alegerea făcută inițiază deschiderea în interfața MS Excel a unui fișier care este denumit automat „**Registru1**”. El este acum în lucru, nesalvat încă.



Fereastra interfeței utilizator a Microsoft Excel este alcătuită din următoarele elemente principale:

- A. Bara de titlu (Title Bar)**
- B. Panglica (Ribbon)**
- C. Caseta de nume (Name Box)** – afișează numele celei în care suntem poziționați (în cazul de mai sus, celula este **B2**, marcată cu chenar verde).
- D. Bara de formule (Formula Bar)**
- E. Zona de lucru (foaia de calcul) (Worksheet)**
- F. Eticheta foi de calcul** – aici vom regăsi toate etichetele foilor de calcul create în registrul aflat în lucru – în exemplul de mai sus avem o singură foaie de calcul, numită (implicit) „Foaie1”.
- G. Buton de adăugare a unei noi foi de calcul (New sheet)** într-un registru

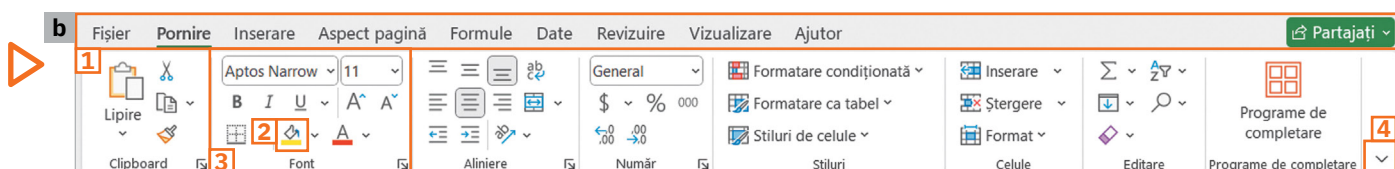
H. Zona de control al modului de vizualizare a tabelului – oferă trei opțiuni: **Normal** (implicit), **Aspect pagină** (Page Layout), **Examinare sfârșit de pagină** (Page Break Preview)

I. Bara de defilare verticală

J. Bara de defilare orizontală

K. Instrumentul de panoramare (Zoom) – permite mărirea/micșorarea zonei vizibile a tabelului. Pune la dispoziție două butoane („+” pentru mărire și „-” pentru micșorare), un potențiomtru care se poate muta cu ajutorul mouse-ului (la stânga pentru micșorare și la dreapta pentru mărire) și o zonă de afișare a nivelului de panoramare (zoom), în partea dreaptă a zonei (în procente).

Panglica – detalieri



Panglica (Ribbon) – include toate comenzile aplicației MS Excel grupate în file, pe tipuri de acțiuni (*imaginea b*).

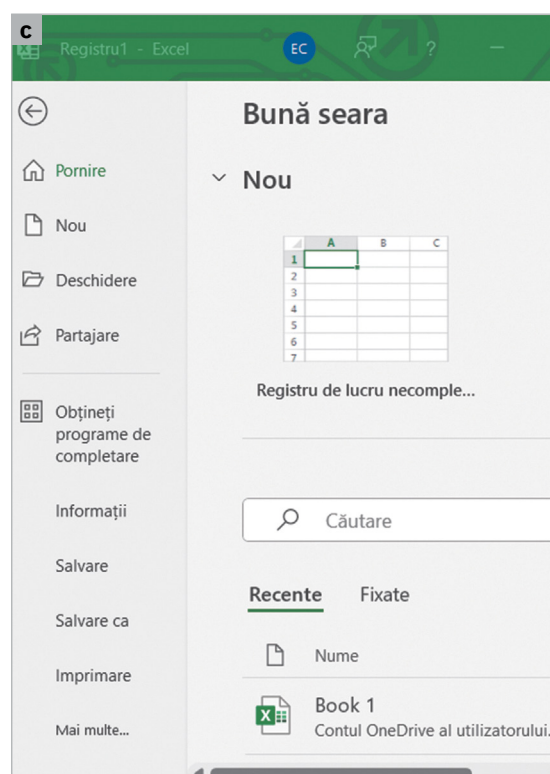
Așadar, în panglică regăsim mai multe file (**1**): **Fișier** (File), **Pornire** (Home), **Inserare** (Insert) etc. Fiecare filă are mai multe **butoane pentru comenzi** (**2**), grupate în **grupuri de comenzi** (**3**) cu nume sugestive (*imaginea de mai sus*).

Schimbăm modalitatea de afișare a **Panglicii**, alegând o opțiune din fereastra ce se deschide la apăsarea butonului (**4**): **Mod ecran complet** (Auto-hide Ribbon), **Afișați numai filele** (Show Tabs), **Afișați întotdeauna Panglica** (Show Tabs and Commands). De exemplu, dacă se alege opțiunea **Afișați numai filele**, atunci se ascunde bara de butoane a filei și rămân vizibile doar numele filelor.

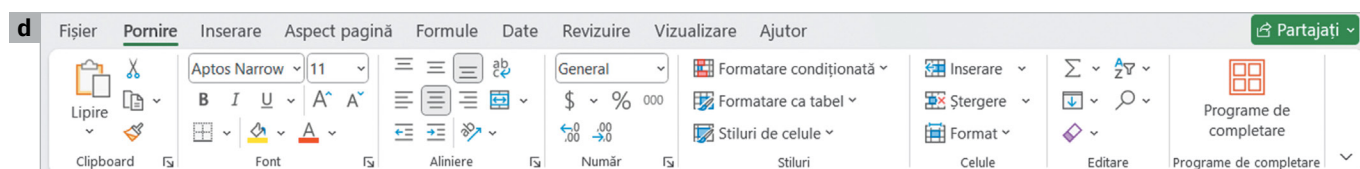
Dacă stați cu mouse-ul deasupra unui buton din interfața Excel, se afișează un text de ajutor cu descrierea funcționalității butonului respectiv.

Principalele file prezente pe panglica MS Excel sunt:

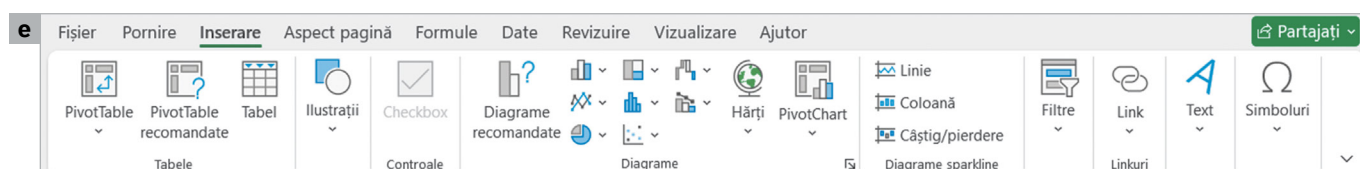
Fila **Fișier** (File) – conține comenzile pentru acțiunile care pot fi realizate asupra fișierului MS Excel. Principalele acțiuni și cele mai uzuale sunt: crearea unui fișier nou, deschiderea unuia existent, salvarea fișierului, exportul acestuia într-un alt format, partajarea, tipărirea și închiderea sa. Tot de aici se pot gestiona opțiunile generale pentru lucrul în Excel. La selectarea acestei file, se deschide o vizualizare nouă, numită **Backstage** (*imaginea c*).



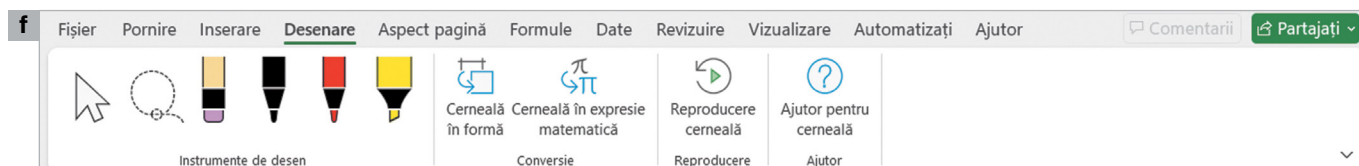
Fila Pornire (Home) – conține comenzile și elementele necesare pentru editarea și formatarea celulelor unui tabel (caracteristici litere, aliniere, stil, contururi de celule, borduri etc.) (*imaginea d*).



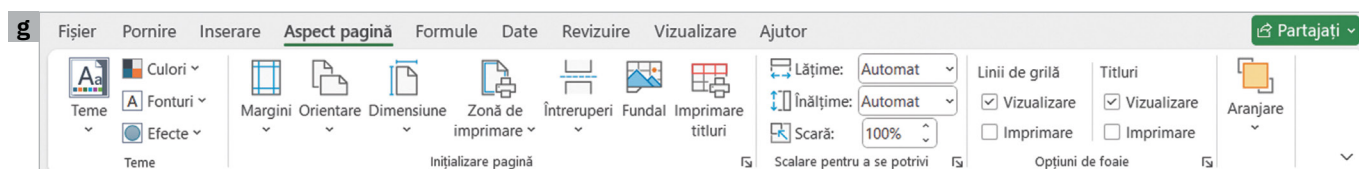
Fila **Inserare** (Insert) – conține comenzile necesare pentru adăugarea altor elemente de conținut într-o foaie de calcul, de exemplu: tabele, ilustrații, diagrame, hărți, conținuturi video, legături către diferite pagini web, texte, simboluri etc. (*imaginea e*).



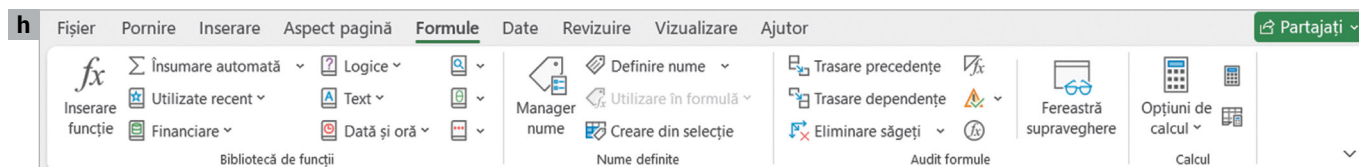
► Fila **Desenare (Draw)** – conține diferite instrumente cu ajutorul cărora se pot insera în foaia de calcul notițe, sublinieri, desene, cu diferite culori, în diferite stiluri. În plus, tot din această filă se pot adăuga și expresii matematice care pot fi chiar scrise de mână, iar recunoașterea acestora se face automat de către program, prin butonul **Cerneală în expresie matematică (Ink to Math)**. Este afișată automat pentru dispozitive cu ecrane tactile sau se poate afișa de la **Fișier (File) > Opțiuni (Options) > Particularizare panglică (Customize Ribbon) > Desenare (Draw)** (imaginea f).



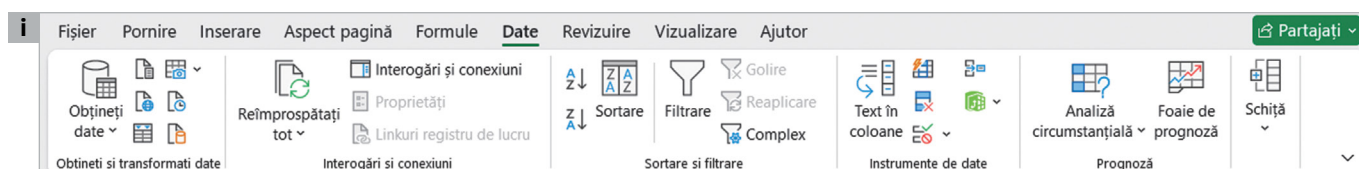
Fila **Aspect pagină (Page Layout)** – conține comenzile pentru configurarea aspectului paginilor documentului. Aici se configurează aspectul fiecărei pagini, marginile, dimensiunea, tipul de orientare: **Portret (Portrait, vertical)** sau **Vedere (Landscape, orizontal)**. Tot aici găsim comenzi pentru aranjarea tabelelor foarte mari, cu multe celule, pentru a fi mai ușor de urmărit, dar și pentru aranjarea (poziționarea, încadrarea, alinierea) diferitelor obiecte incluse în document (de exemplu, imagini) față de celulele care au conținut (numere sau texte) (imaginea g).



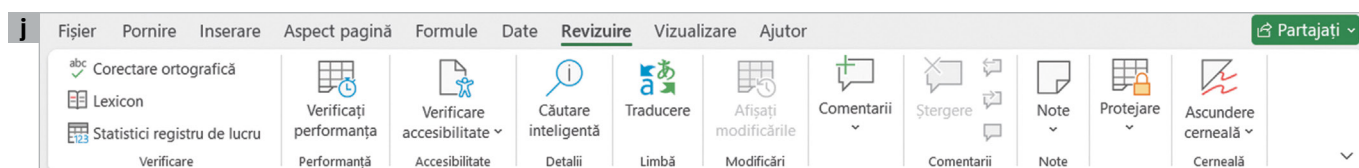
Fila **Formule (Formulas)** – conține comenzi pentru utilizarea diferitelor formule în foaia de calcul, de la însumare automată, până la funcții matematice complexe, trecând prin funcții logice, funcții pentru lucrul cu caractere etc. Tot aici găsim două comenzi foarte utile: **Trasare precedente (Trace Precedents)** și **Trasare dependențe (Trace Dependents)**. Prima afișează săgeți ce indică celulele care afectează valoarea celulei selectate în prezent, iar a doua afișează săgeți care indică celulele afectate de valoarea celulei selectate în prezent (imaginea h).



Fila **Date (Data)** – conține comenzi pentru lucrul cu datele din foaia de calcul. De exemplu, comenzi de sortare a datelor (ordonare alfabetică sau valorică, crescătoare sau descrescătoare), comenzi de filtrare a datelor din celulele selectate sau comenzi de scindare a unei coloane de text în mai multe coloane (puteți să separați o coloană de nume complete în coloane separate pentru nume și prenume) etc. (imaginea i).



+ Fila **Revizuire (Review)** – permite verificarea ortografică a textelor din tabel conform cu limba în care acestea sunt scrise, adăugarea de comentarii, traducerea etc. Ea conține comenzile necesare vizualizării diferitelor statistici legate de registrul de lucru. Protejarea prin parolă a unei foi de calcul sau a unui registru este, de asemenea, posibilă din această filă (imaginea j).

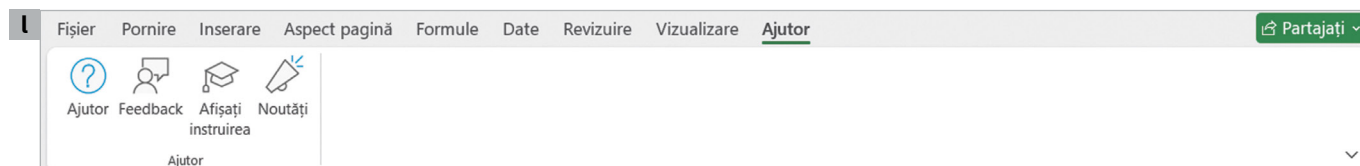


Fila **Vizualizare (View)** – permite alegerea modalității de vizualizare a foi de calcul în fereastră (**Normal**, **Aspect pagină**, **Examinare sfârșit de pagină**). Există aici posibilitatea de afișare în fereastra de lucru a unor instrumente

ajutătoare: **Rigla** (Ruler), **Linii de grilă** (Gridlines), **Bara de formule** (Formula Bar) ori **Titluri** (Headings) sau a unor instrumente care ajută la afișarea foii de calcul sau a unei zone selectate mărite sau micșorate (*imaginea k*).



Fila **Ajutor** (Help) – conține comenzi care permit căutarea de diferite informații de ajutor pentru utilizarea acestui produs software, precum și transmiterea de feedback firmei producătoare, pentru a sprijini îmbunătățirea pachetului de programe MS Office (*imaginea l*).



Fila **Partajați** (Share) – de aici puteți partaja fișierul vostru, pentru a lucra la el împreună cu alți colegi (necesită mai întâi salvarea online în cloud).

Aplicați și investigați

- Realizați următoarele acțiuni în ordinea în care sunt specificate. Observați, la fiecare pas, ce se întâmplă:
 1. Deschideți MS Excel și creați un registru de lucru nou, necompletat; observați cum arată interfața și registrul vostru.
 2. Descoperiți elementele componente ale barei de titlu, făcând analogie cu cele deja învățate pentru aplicația MS Word, în ordine de la stânga la dreapta: **Bara de acces rapid** (Quick Access Bar), numele fișierului aflat în lucru, caseta de căutare Microsoft, numele contului de utilizator, panoul de management al ferestrei (permite minimizarea, restaurarea și, respectiv, maximizarea ferestrei). Modificați comenzile din bara de acces rapid prin selectarea/deselectarea opțiunilor respective din meniul care apare la apăsarea săgeții din dreapta sa.
 3. Alegeți una câte una cele trei modalități de afișare a **Panglicii** (Mod ecran complet, Afișări numai filele, Afișări întotdeauna Panglica). Observați diferența dintre acestea.
 4. Scrieți pe coloana **A** numele tuturor colegilor de clasă. Veți observa că numele sunt mai lungi decât lungimea celulei. Pentru a rezolva această mică problemă, duceți-vă cu mouse-ul pe linia dintre capetele de coloane **A** și **B**, apăsați pe ea, țineți apăsat butonul mouse-ului și trageți la dreapta până încapă și cel mai lung nume din listă.
 5. Completați în coloana **B** timpii realizați la sport, la proba de 50 m viteză.
 6. Modificați nivelul de panoramare al foii de calcul. Analizați aspectul de fiecare dată.
 7. Alegeți, pe rând, modalitățile de vizualizare a foii de calcul: **Normal**, **Aspect pagină**, **Examinare sfârșit de pagină**. Analizați aspectul de fiecare dată.

Exersați

1. Scrieți în caiete răspunsul corect:

a. MS Excel este:

 - un procesor de texte;
 - un procesor de sunete;
 - un program de calcul tabelar.

b. Panglica aplicației MS Excel conține fila:

 - **Corespondență;**
 - **Tranziții;**
 - **Date.**
2. Enumerați cel puțin două moduri de vizualizare a unei foi de calcul.
3. Configurați nivelul de panoramare (zoom) la 55%. În câte moduri puteți modifica acest nivel?
4. Configurați panglica aplicației MS Excel astfel încât să fie vizibile doar filele.
5. Scrieți în caiete **A** (Adevărat) sau **F** (Fals), în funcție de valoarea de adevăr a propozițiilor:

a. Un fișier pregătit cu ajutorul aplicației MS Excel poate avea extensia **.xlsx**.

b. În filele aplicației MS Excel regăsim comenzi pentru afișarea statisticilor legate de foaia de calcul.

c. MS Excel poate lucra numai (și obligatoriu) cu două foi de calcul într-un registru.

d. În MS Excel, fila **Inserare** (Insert) conține grupul de comenzi **Diagramme** (Charts).

A F

A F

A F

A F
6. Enumerați trei comenzi aflate pe panglica aplicației MS Excel a căror denumire începe cu litera inițială a numelui sau a prenumelui vostru.
7. **Dezbateri.** Organizați o dezbateri cu tema „Interfața Excel – accesibilă sau prea complexă pentru utilizatorii începători?”.

3 Operații principale cu un registru de calcul



Amintiți-vă

Cu ajutorul aplicației MS Excel se pot crea **foi de calcul electronice**. Acestea pot fi salvate în format electronic pe un mediu de stocare (ca fișiere) și/sau pot fi tipărite la imprimantă.

MS Excel ne pune la dispoziție o multitudine de comenzi cu ajutorul cărora putem realiza cu ușurință tabele, de la cele mai simple până la cele mai complexe, și le putem da și o formă estetică deosebită. **Panglica** include toate comenzile aplicației MS Excel grupate în file, pe tipuri de acțiuni.

Registrul de calcul este fișierul creat cu ajutorul aplicației de calcul tabelar (în cazul nostru, MS Excel) și poate fi format din mai multe **foi de calcul**.



Descoperiți

Operații elementare de lucru cu un registru de calcul



Denumirea operației efectuate asupra fișierului registru de calcul

Detalierea acțiunilor efectuate în MS Excel

Creare

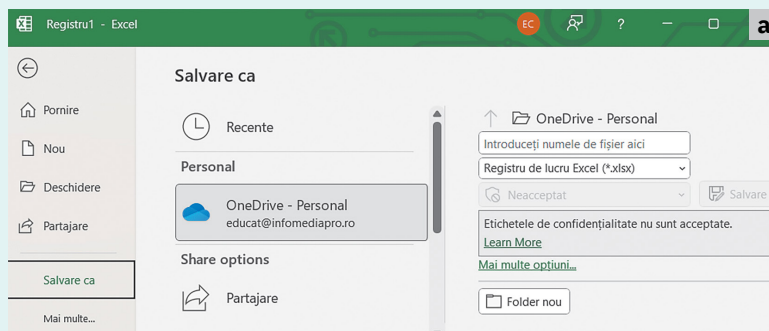
Pentru a crea un fișier registru de calcul nou, alegeți **Registrul de lucru necompletat** (*Blank workbook*) din fereastra de pornire a MS Excel. De asemenea, mai puteți apela la una dintre următoarele modalități: alegeți comanda **Nou** (*New*) din fila **Fișier** (*File*) sau apăsați combinația de taste **Ctrl+N**. Comanda **Nou** poate fi adăugată și în **Bara de acces rapid** (*Quick Access Toolbar*).

Deschidere

Pentru a deschide un registru existent, selectați comanda **Deschidere** (*Open*) din fila **Fișier** (*File*) sau apăsați **Ctrl+O**. Comanda **Deschidere** (*Open*) poate fi adăugată și în **Bara de acces rapid** (*Quick Access Toolbar*).

Salvare

- Selectarea comenzii **Salvare ca** (*Save as*) din fila **Fișier** (*File*) permite salvarea fișierului curent cu un nume tip și nume alese de voi, într-o locație aleasă de voi, pe calculatorul propriu, pe un dispozitiv extern (un mediu de stocare conectat la calculator) sau în cloud (*imaginea a*).

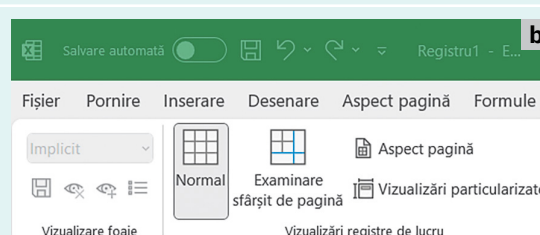


- Salvarea unui fișier, fără a se schimba numele, tipul (extensia) sau locația, se poate face selectând comanda **Salvare** (*Save*) din fila **Fișier** (*File*), apăsând butonul **Salvare** din bara de acces rapid sau combinația de taste **Ctrl+S**.



Vizualizare

Accesați fila **Vizualizare** (*View*) și alegeți opțiunea dorită: **Normal** (*Normal*), **Aspect pagină** (*Page Layout*) sau **Examinare sfârșit de pagină** (*Page Break Preview*) (*imaginea b*).



Imprimare (tipărire)

Pentru a tipări registrul de calcul, trebuie să alegeți comanda **Imprimare** (*Print*), aflată în fila **Fișier** (*File*). Documentul se poate tipări la o imprimantă instalată în prealabil sau poate fi tipărit într-un fișier de tip **.pdf**, care se salvează pe unul dintre mediile de stocare aflate la dispoziție. Acest lucru se alege din caseta **Imprimantă** (*Printer*).

Închidere

Pentru a închide un registru deschis, selectați comanda **Închidere** (*Close*) din fila **Fișier** (*File*) sau apăsați butonul de închidere din colțul din dreapta-sus al ferestrei.



Aplicații și investigații

► Pentru a vă familiariza cu interfața aplicației MS Excel și cu operațiile de bază pe care le puteți face cu aceasta, executați, în ordinea dată, următoarele acțiuni:

1. Deschideți aplicația MS Excel. Din fila **Fișier** (File), deschideți submeniul **Nou** (New) și alegeți un **Registru de lucru necompletat** (New workbook).
2. Scrieți tabla înmulțirii cu 6 în foaia de calcul a registrului creat anterior, astfel: pe prima coloană scrieți numerele de la 1 la 10, pe a doua coloană scrieți de 10 ori cifra 6 (în fiecare celulă din dreptul celulelor cu numerele de la 1 la 10), iar în a treia coloană scrieți rezultatele pentru fiecare înmulțire din tabla înmulțirii cu 6.
3. Apăsați combinația de taste **Ctrl+S**. Ce apare pe ecran?
4. Salvați registrul cu numele „Tabla înmulțirii cu 6”: din fila **Fișier** (File), alegeți opțiunea **Salvează ca** (Save as). Alegeți locația în care doriți să salvați (de preferat, un folder al vostru, de lucru) și dați-i numele precizat anterior. Apoi apăsați butonul **Salvare** (Save).
5. Din fila **Fișier** (File), alegeți opțiunea **Închide** (Close).
6. Redeschideți registrul salvat anterior folosind din fila **Fișier** (File) opțiunea **Deschide** (Open). În lista de fișiere **Recente** (Recent) veți regăsi registrul de calcul „Tabla înmulțirii cu 6”. Clic pe el și se va deschide.



Descoperiți

Operații de editare într-o foaie de calcul (selectare, copiere, mutare, ștergere)

► Pentru a putea realiza operațiile de **copiere**, **mutare**, **ștergere**, este obligatoriu să aveți selectate în prealabil zonele asupra cărora doriți să faceți acțiunile. În continuare, vom realiza aceste operații în foaia de calcul cu tabla înmulțirii creată anterior:

- **Selectați** întreaga tablă a înmulțirii: faceți clic în celula **A1**, țineți apăsat butonul mouse-ului și trageți până la celula **C10** (imaginea c).
- **Copiați zona selectată** (duplicați-o în aceeași foaie de calcul):
 - a. **Copiați în clipboard** zona selectată utilizând una dintre posibilitățile:
 1. În fila **Pornire** (Home), din grupul de comenzi **Clipboard**, apăsați butonul **Copiere** (Copy);
 2. Clic cu butonul drept al mouse-ului pe zona selectată și din meniul contextual care vi se deschide alegeți opțiunea **Copiere** (Copy);
 3. Apăsați combinația de taste **Ctrl+C**.
 - b. **Poziționați-vă la locul unde doriți să copiați și lipiți zona copiată anterior**: faceți clic în celula **E1** și apăsați butonul **Lipire** (Paste) din grupul de comenzi **Clipboard** din fila **Pornire** (Home) sau apăsați combinația de taste **Ctrl+V**. Puteți apăsa direct tasta **Enter**. Acum aveți pe foaia de calcul de două ori tabla înmulțirii cu 6.
- **Mutați zona selectată**: Dacă doriți să mutați zona selectată și nu să o copiați, în loc de acțiunea **Copiere** (Copy), alegeți **Decupare** (Cut) (sau apăsați combinația de taste **Ctrl+X**), apoi poziționați-vă în locul unde vreți să o mutați și apăsați butonul **Lipire** (Paste) sau apăsați tasta **Enter**. Faceți acest lucru cu a doua tablă a înmulțirii din foaia voastră de calcul.
- **Ștergeți**: Acum ștergeți cea de-a doua tablă a înmulțirii din foaia de calcul: selectați-o și apăsați tasta **Ștergere** (Delete).

	A	B	C	D	C
1	1	6	6		
2	2	6	12		
3	3	6	18		
4	4	6	24		
5	5	6	30		
6	6	6	36		
7	7	6	42		
8	8	6	48		
9	9	6	54		
10	10	6	60		
11					
12					
13					

Operații cu foi de calcul într-un registru: creare (inserare), accesare, redenumire, ștergere

Vom realiza operații cu foile de calcul într-un registru, crearea (inserarea), accesarea, redenumirea, ștergerea acestora, cu exemplificare în exemplul construit mai sus.

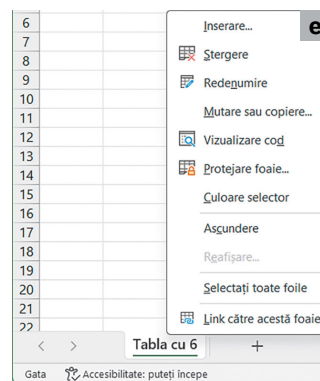
Observați în partea stângă-jos a aplicației eticheta cu numele foi de calcul create automat la crearea registrului. Implicit poartă numele „Foaie1” (imaginea d).

- **Redenumire**: Faceți dublu clic pe etichetă și scrieți noul nume: „Tabla cu 6”.
- **Creare (inserare) foaie nouă de calcul**: Apăsați butonul **+** de lângă eticheta cu numele foi de calcul inițiale și automat se creează o foaie nouă. Creați două noi foi de calcul, „Foaie2” și „Foaie3”.
- **Accesare foaie de calcul**: Pentru a accesa o foaie de calcul sau alta, doar faceți clic pe eticheta cu numele foi dorite. Faceți clic pe eticheta cu numele „Tabla cu 6” și respectiva foaie va trece în față și va putea fi accesată pentru vizualizare, modificare etc.
- **Schimbare ordine foi de calcul**: Trageți, folosind **drag & drop**, eticheta foi pe care doriți să o mutați și puneți-o la locul dorit. Trageți foaia de calcul „Foaie2” înaintea foi de calcul „Tabla cu 6”.
- **Ștergeți foaia de calcul**: Faceți clic cu butonul drept al mouse-ului pe eticheta cu numele foi și selectați opțiunea **Ștergere** (Delete). Ștergeți, astfel, foaia de calcul „Foaie2”.

15					d
16					
17					
18					
19					
20					
21					

După cum puteți observa în imaginea alăturată (*imaginea e*), acțiunile de **Inserare** (*Insert*), **Ștergere** (*Delete*), **Redenumire** (*Rename*) și **Mutare sau copiere...** (*Move or Copy...*) sunt disponibile, alături de alte câteva, în meniul contextual care apare la un clic de buton drept al mouse-ului pe eticheta cu numele unei foi de calcul.

- La apăsarea butonului **Mutare sau copiere...** (*Move or Copy...*), se deschide o fereastră din care se alege:
1. în care registru se dorește copierea/mutarea: câmpul **În registrul** (*To book*);
 2. înaintea cărei foi de calcul se introduce fila respectivă: se alege din listă **Înainte de foie** (*Before sheet*);
 3. dacă se dorește mutarea sau copierea; în acest ultim caz se bifează **Crearea unei copii** (*Create a copy*).



Aplicații și investigații

► Realizați un tabel de colectare a datelor (prin introducere directă) pentru un studiu statistic simplu.

Pentru aceasta, creați un registru Excel și salvați-l în cloud cu numele „Timp la televizor.xlsx”. Trimiteți invitația de a lucra pe acest registru cât mai multor colegi din clasa voastră, de preferat tuturor.

Faceți următorul experiment: urmăriți-vă (toți colegii implicați trebuie să facă același lucru) timp de o săptămână și notați în fiecare zi câte minute vă uitați la televizor, indiferent la ce emisiune. Notați datele în registrul „Timp la televizor.xlsx”, într-un tabel similar cu cel de mai jos. În loc de „Nume 1”, „Nume 2”, fiecare participant la studiu își va trece numele propriu.

Pentru ca tabelul să și arate frumos, selectați celulele cu zilele săptămânii și, de la butonul **Culoare de umplere** (*Fill Color*) din grupul **Font**, fila **Pornire** (*Home*), alegeți o nuanță deschisă de portocaliu, apoi selectați celulele cu numele participanților la studiu și colorați-le într-o nuanță de albastru-deschis.

Selectați întregul tabel și tot de la grupul **Font**, de la butonul **Borduri** (*Borders*), alegeți **toate bordurile** pentru a marca marginile tuturor celulelor cu o linie.

La finalul săptămânii, adunați minutele din toate cele șapte zile pentru fiecare nume în parte și gândiți-vă câte lucruri ați fi putut face în plus în acest timp petrecut la televizor.

	Luni	Marti	Miercuri	Joi	Vineri	Sambata	Duminica
Nume 1							
Nume 2							
Nume 3							
Nume 4							
Nume 5							
Nume 6							
Nume 7							
Nume 8							
Nume 9							
Nume 10							



Exersați

1. Asociați corect noțiunile referitoare la MS Excel din cele două coloane:



- | | |
|-------------------------|---|
| a. Ctrl+N | I. Afișarea mărită/micșorată a unei foi de calcul |
| b. Aspect pagină | II. Creare de nou registru de calcul |
| c. Zoom | III. Redenumire registru de calcul |
| | IV. Mod de vizualizare a unui registru de lucru |

2. Care este rezultatul accesării comenzii **Salvare ca** (*Save as*) din fila **Fișier** (*File*)?

3. Alegeți variantele corecte.

- | | |
|---|------------------|
| I. Extensia unui fișier creat în MS Excel poate fi: | |
| a. xlsx ; | b. docx ; |
| | c. mp4 . |
| II. Alegerea unui mod de vizualizare se realizează accesând: | |
| a. grupul de comenzi Afișare din fila Vizualizare ; | |
| b. grupul de comenzi Vizualizări registre de lucru din fila Vizualizare ; | |
| c. grupul de comenzi Fereastră din fila Vizualizare . | |

4. **Portofoliu.** Alcătuiți un eseu de maximum o pagină cu tema „Ce s-ar întâmpla dacă ați uita să salvați un fișier la care ați lucrat singur, într-o duminică, timp de 7 ore, termenul de predare fiind în două zile?”. Includeți în eseu propuneri de acțiuni atât pentru remedierea problemei, cât și proactive, de preîntâmpinare a unui astfel de eveniment.

4 Formatarea unui registru de calcul.

Tipuri de date. Formule



Amintiți-vă



Am văzut în lecția anterioară că putem aranja tabelele și din punct de vedere estetic. Puteți colora unele celule ale foii de calcul: doar selectați celulele dorite și, de la butonul **Culoare de umplere** (*Fill Color*) din grupul **Font**, fila **Pornire** (*Home*), alegeți o culoare. Puteți pune linii care să marcheze coloanele și liniile din foaia de calcul: selectați zona dorită sau întregul tabel și tot de la grupul **Font**, de la butonul **Borduri** (*Borders*), alegeți **Toate bordurile** (*All Borders*) pentru a marca marginile tuturor celulelor cu o linie.

Se pot face însă și alte operații de formatare a rândurilor și a coloanelor unei foi de calcul, precum și diverse operații de formatare a celulelor, în funcție de informația pe care o conțin.



Descoperiți

1. Noțiuni de formatare

a. Operații de formatare a rândurilor/coloanelor



Să reluăm registrul de calcul cu tabla înmulțirii cu 6.

În celulele completate cu tabla înmulțirii, spațiul ocupat de numere este mic, în comparație cu dimensiunea celulelor. Cum facem să micșorăm dimensiunea celulelor, astfel încât întreaga tablă a înmulțirii să ocupe un spațiu mai mic? Poziționați mouse-ul pe linia dintre literele care etichetează două coloane alăturate (de exemplu, între **A** și **B**). Cursorul mouse-ului se va transforma într-o linie verticală cu două săgețele spre stânga și spre dreapta, sugerând astfel că puteți trage de acea linie într-o direcție sau alta, micșorând, respectiv măbind lățimea întregii coloane din stânga cursorului, în cazul nostru coloana **A**. Țineți butonul stâng al mouse-ului apăsat și trageți spre stânga, micșorând lățimea coloanei cât credeți de cuviință. Procedați la fel cu coloanele **B** și **C**. Dacă veți face dublu clic în momentul în care cursorul mouse-ului este transformat după cum am văzut mai sus, lățimea coloanei se va ajusta automat la mărimea minimă, astfel încât să încapă textul cel mai lung din celulele coloanei respective (*imaginile a și b*).

Procedați similar pentru ajustarea dimensiunii verticale a unui rând. Se selectează rândurile pentru care se dorește modificarea dimensiunii, se poziționează cursorul mouse-ului pe linia dintre cifrele care constituie etichetele rândurilor și, când acesta se transformă într-o linie verticală cu două săgeți în sus și în jos, se trage în direcția necesară, cu butonul mouse-ului apăsat, până se obține dimensiunea dorită (*imagea c*).

	A	B	C	D
1	1	6	6	
2	2	6	12	
3	3	6	18	
4	4	6	24	
5	5	6	30	
6	6	6	36	
7	7	6	42	
8	8	6	48	
9	9	6	54	
10	10	6	60	
11				

Coloanele cu dimensiunile implicite

	A	B	C	D
1	1	6	6	
2	2	6	12	
3	3	6	18	
4	4	6	24	
5	5	6	30	
6	6	6	36	
7	7	6	42	
8	8	6	48	
9	9	6	54	
10	10	6	60	
11				

Coloanele cu dimensiunile micșorate

	A	B	C	D
1	1	6	6	
2	2	6	12	
3	3	6	18	
4	4	6	24	
5	5	6	30	
6	6	6	36	
7	7	6	42	
8	8	6	48	
9	9	6	54	
10	10	6	60	
11				

Înălțimea rândurilor mărită



Rețineți

Dimensiunile unei celule pot fi ajustate atât pe orizontală, cât și pe verticală.

b. Operații de formatare a celulelor

+ Pentru o formatare ușoară și de calitate, MS Excel pune la dispoziție mai multe **stiluri de celule** predefinite pe care le puteți utiliza. Un **stil** reprezintă de fapt un set de caracteristici de formatare (fonturi, dimensiuni de font, de borduri și culori de celule etc.).

Puteți alege un astfel de stil acesând fila **Pornire (Home)** > grupul **Stiluri (Styles)** > **Stiluri de celule (Cell Styles)**. Puteți să vă creați însă propria formatare, aplicând pe conținutul celulelor unei foi de calcul toate opțiunile de formatare de font și de paragraf pe care le cunoașteți de la editorul de texte. Vedeți grupurile de comenzi **Font** și **Aliniere (Alignment)** din fila **Pornire (Home)**.

Să aliniem centrat, pe mijlocul coloanei, și să apară îngroșate rezultatele înmulțirilor. Mai întâi, pentru a vedea mai bine rezultatul acțiunii, lățim puțin coloana **C**. Apoi selectăm celulele de la **C1** la **C10** (sau, mai rapid, facem clic pe litera **C**, care etichetează coloana, selectând-o astfel) și facem clic pe butonul **Aldin** din grupul **Font** și pe butonul **La centru (Center)** din grupul **Aliniere (Alignment)** (imaginea d).

	A	B	C	D
1	1	6	6	
2	2	6	12	
3	3	6	18	
4	4	6	24	
5	5	6	30	
6	6	6	36	
7	7	6	42	
8	8	6	48	
9	9	6	54	
10	10	6	60	
11				d

Coloana C – îngroșată și centrată

Rețineți

+ Asupra unei celule pot fi aplicate mai multe **opțiuni de formatare**. Pentru a vedea toate posibilitățile de formatare disponibile, faceți clic cu butonul drept al mouse-ului pe o celulă, iar din meniul contextual care se deschide alegeți **Formatare celule...** (*Format Cells...*). Va apărea o fereastră cu mai multe file, dintre care, pentru moment, ne interesează filele **Aliniere (Alignment)**, **Font**, **Bordură (Border)** și **Umplere (Fill)**.

La filele **Aliniere (Alignment)** și **Font** puteți ajunge și făcând clic pe săgețelele din dreapta-jos, în grupurile cu aceleași nume din panglica programului MS Excel, fila **Pornire (Home)**.

Selectați coloana **A** din foaia de calcul cu tabla înmulțirii și aplicați, pe rând, toate opțiunile disponibile în filele **Aliniere (Alignment)** (imaginea e) și **Font** (imaginea f). Observați ce efect au modificările asupra textelor din celule?

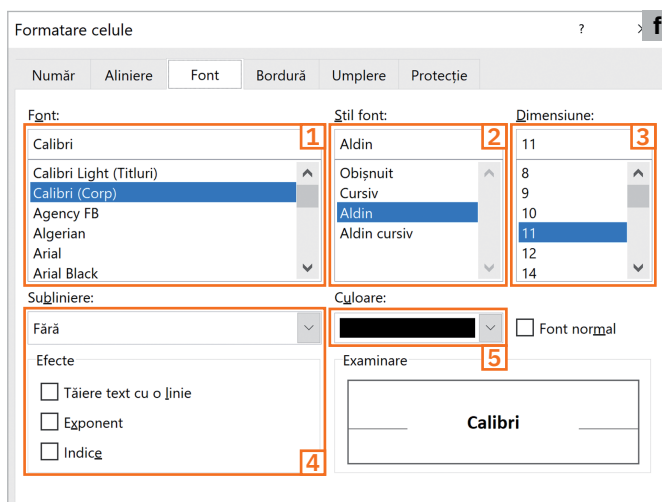
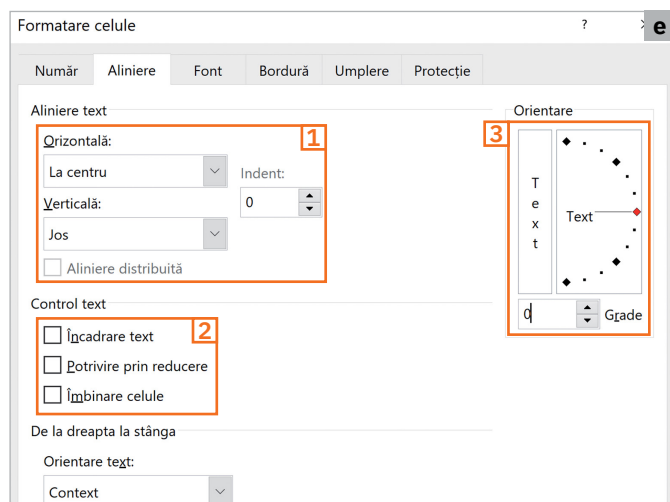
Din fila **Aliniere (1)**, puteți alinia textul pe orizontală sau pe verticală alegând opțiunea dorită din lista derulantă pusă la dispoziție (la stânga, la centru, la dreapta etc.).

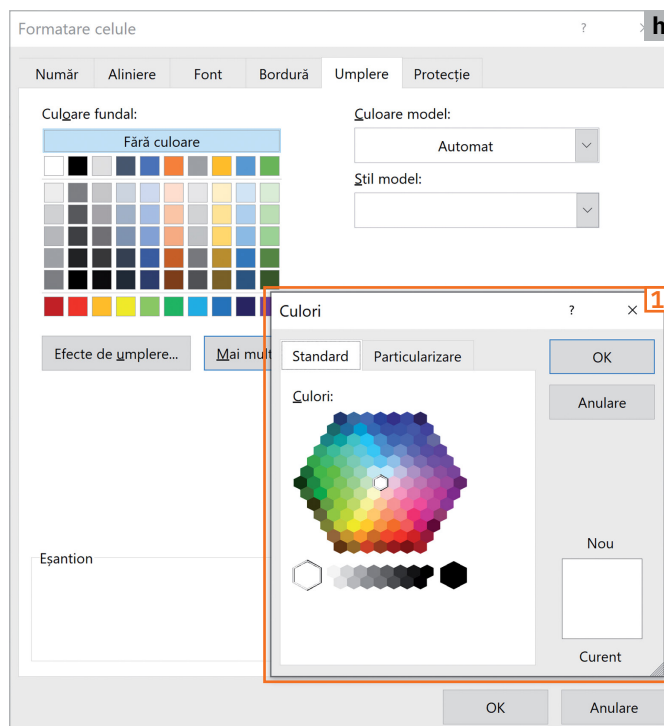
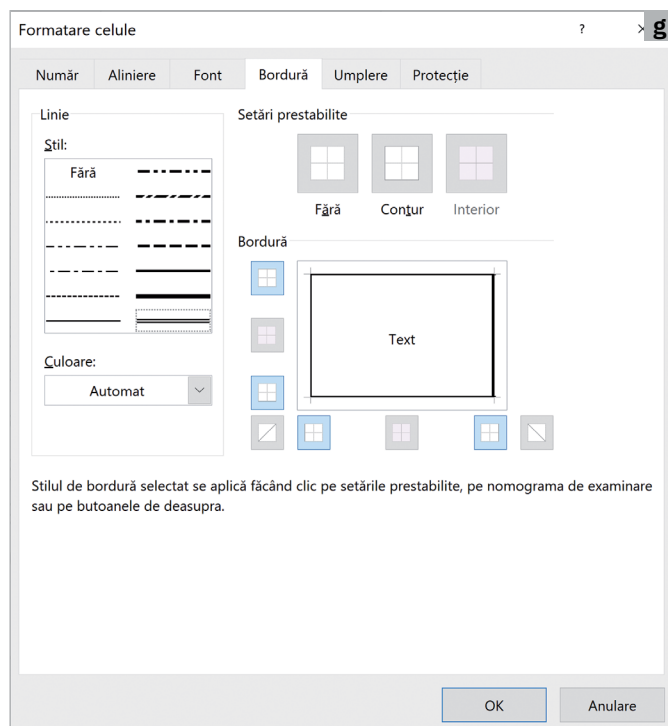
Puteți controla cum să apară textul dintr-o celulă (2):

- dacă dimensiunea sa e mai mare decât cea a celulei, să fie împărțit pe mai multe linii: **Încadrare text (Wrap text)**;
- dacă dimensiunea sa e mai mare decât cea a celulei, să fie redusă dimensiunea cât să se încadreze în celulă: **Potrivire prin reducere (Shrink to fit)**;
- să se îmbine mai multe celule alăturate selectate, astfel încât textul să fie comun pentru toate: **Îmbinare celule (Merge cells)**.

O opțiune interesantă este aceea de **Orientare** a textului în celule (3). De aici puteți configura textul să fie orizontal, vertical sau oblic (cu unghiuri specificate în grade) în celula foi de calcul.

Din fila **Font**, se pot schimba fontul (1), stilul acestuia (obișnuit, cursiv, aldin, aldin cursiv) (2), dimensiunea (3), culoarea (5). De asemenea, poate fi subliniat sau poate avea alte efecte (tăiat cu o linie, micșorat ca exponent sau ca indice) (4).





Selectați toate celulele cu tabla înmulțirii și aplicați, pe rând, toate opțiunile disponibile în filele **Bordură** (Border) (imaginea g) și **Umplere** (Fill) (imaginea h). Experimentați diverse stiluri de linie pentru contururile celulelor, diferite culori de umplere pentru fundalul celulelor și diferite culori pentru scrisul din interiorul celulelor.



Observați că, deși aveți la dispoziție o multitudine de culori și nuanțe pentru umplerea fundalurilor celulelor, în fila **Umplere** (Fill) există și un buton **Mai multe culori...** (More colors...) de la care puteți alege dintre mai multe nuanțe disponibile (1).



Aplicați și investigați

- ▶ Selectați toată tabla înmulțirii cu 6 și, de la opțiunile din fila **Borduri** (Borders), găsiți soluția pentru a avea conturul exterior al tabelului cu tabla înmulțirii marcat cu o dungă dublă, iar liniile interioare să fie marcate cu o linie punctată. De asemenea, umpleți fiecare coloană cu o altă culoare, iar scrisul faceți-l să fie într-o culoare diferită de culoarea neagră, dar care să se vadă bine pe fundalul ales de voi pentru celule.

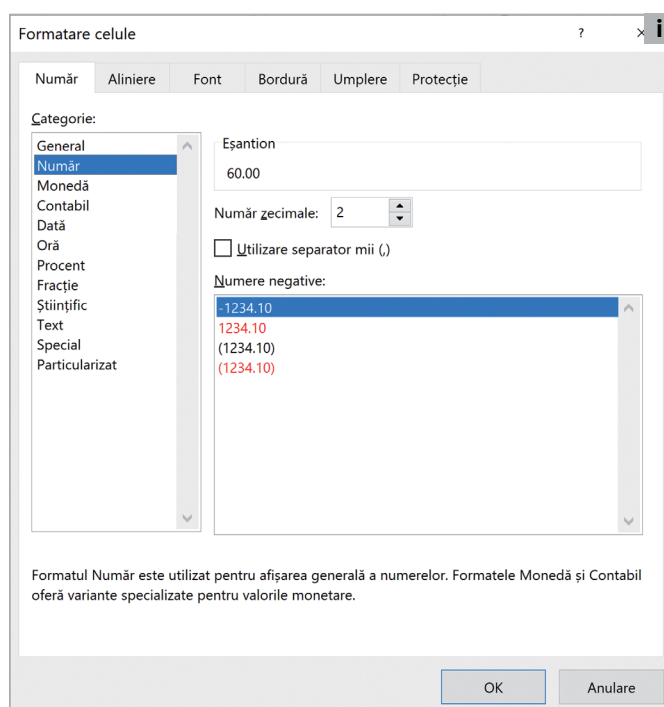


Descoperiți

2. Tipuri de date: numeric, text, dată calendaristică și altele

Poate că cea mai importantă opțiune de formatare a unei celule (din punct de vedere practic) este faptul că o putem configura pentru a avea conținutul de un anumit tip (imaginea i). De exemplu:

- număr (puteți configura utilizarea separatorului de mii, precum și numărul de zecimale care să se afișeze);
- monedă (puteți configura diferite monede, inclusiv RON);
- dată sau oră (puteți face operații cu date, respectiv ore);
- procent;
- fracție (dacă selectați „fracție” și introduceți în acea celulă valoarea „5.5”, va apărea „5 1/2”);
- text etc.



3. Utilizarea formulelor de calcul într-o foaie de calcul

Menirea principală a unei foi de calcul, dincolo de a ne oferi un tabel predefinit în care putem introduce date, este aceea de a ne oferi posibilitatea de a obține rezultate calculate automat, în baza unor formule.

Să luăm exemplul foi de calcul cu tabla înmulțirii cu 6. În celula **C1**, în loc să scrieți rezultatul înmulțirii, calculat de voi, scrieți formula care calculează acel rezultat și îl afișează în celula tabelului.

O formulă într-o celulă de foaie de calcul începe întotdeauna cu semnul = . Așadar, formula noastră este: **=1*6**.



Rețineți

Puteți folosi toți operatorii pe care îi cunoașteți: + (adunare), - (scădere), * (înmulțire), / (împărțire), ^ (ridicare la putere) etc. Bineînțeles că se pot folosi și parantezele, cu același rol și aceeași prioritate pe care le știți de la matematică.

O facilitare și mai interesantă oferită de programele de calcul tabelar este să înmulțim nu două numere, ci valorile conținute în două celule, oricare ar fi acele valori. Astfel: pentru a obține în celula **C1** rezultatul înmulțirii lui 1 cu 6, adică ceea ce este în celula **A1** înmulțit cu ceea ce este în celula **B1**, trebuie să scriem în celula **C1** formula:

$$= A1*B1.$$

Acum, dacă vă veți poziționa în celula **C1** și o veți copia în clipboard (**Ctrl+C**), iar apoi vă veți duce în celula **C2** și veți „lipi” (**Ctrl+V**) acolo formula copiată, veți avea surpriza ca după ce apăsați **Ctrl+V** să vă apară „12”, adică rezultatul înmulțirii lui 2 cu 6. Aceasta pentru că programul a știut să copieze și să adapteze formula de calcul astfel: dacă în **C1** s-a introdus formula **A1*B1**, atunci în **C2** se va introduce formula **A2*B2**, adică **2*6**.

Programul mai oferă o facilitare de lucru: după ce ați introdus în celula **C1** formula de calcul cu adresare la celule (**=A1*B1**), este suficient să vă poziționați cu mouse-ul pe celula **C1** și să trageți de pătrățică din colțul din dreapta-jos al celulei până la **C10** (adică pentru toată tabla înmulțirii). Vor apărea instant rezultatele tuturor înmulțirilor cu 6, de la 1 la 10. Aceasta pentru că s-a replicat automat formula de calcul din celula **C1**, cu referință la rândul pe care apare (în **C2**, se calculează **A2*B2**, în **C3** se calculează **A3*B3** și așa mai departe).



Știați că...?



Dacă introduceți un număr într-o celulă (de exemplu, numărul 8 în celula **B1**), iar apoi trageți cu mouse-ul de colțul din dreapta-jos al celulei în jos pe coloană, veți replica numărul 8 (adică ce e conținut în celula **B1**) în toate celulele peste care trageți mouse-ul. Veți obține automat o **serie de date** cu aceeași valoare.



Aplicați și investigați

- ▶ În foaia de calcul cu tabla înmulțirii cu 6 înlocuiți, în coloana **B**, cifra 6 cu cifra 8, din celula **B1** până în celula **B10**. Ce s-a întâmplat? Rezultatele din coloana **C** au fost automat modificate și ați obținut tabla înmulțirii cu 8.



Descoperiți

4. Sortarea crescătoare/descrescătoare a datelor dintr-un tabel după unul sau mai multe criterii

În lecția anterioară ați creat registrul „Timp la televizor.xlsx”. Să presupunem că îl avem completat și arată ca în imaginea de mai jos. Acum sarcina este să **ordonați crescător**, după timpul petrecut la TV în ziua de luni, toate numele din tabel. Pentru aceasta, poziționați-vă pe o celulă din tabel și urmați pașii:

- Mergeți în fila **Date** (*Data*);
- Din grupul **Sortare și filtrare** (*Sort&Filter*) apăsați butonul **Sortare** (*Sort*);
- În fereastra care se deschide, în câmpul **Sortare după** (*Sort by*) alegeți coloana după care se realizează sortarea, **Luni** (sarcina era să se ordoneze după timpul petrecut la TV în ziua de luni), iar la **Ordine** alegeți tipul de ordonare dorit, **De la cel mai mic la cel mai mare** (în ordine crescătoare) (*Lowest to highest*);
- Apăsați butonul **OK**.

	A	B	C	D	E	F	G	H
1		Luni	Marti	Miercuri	Joi	Vineri	Sambata	Duminica
2	Popescu Dan	18	233	41	33	43	0	0
3	Vrânceanu Vlad	23	43	22	51	11	0	0
4	Mihai Daniel	43	55	12	15	19	45	0
5	Iscrulescu Flavia	44	22	56	10	10	110	0
6	Popescu Ana	18	43	82	19	14	90	0
7	Lazăr Andrei	45	12	112	12	15	0	0
8	Popescu Virginia	18	21	23	18	9	0	0
9	Vasilescu Marian	12	14	33	15	11	0	0
10	Dănescu Andreea	5	16	22	13	23	0	0
11	Avram Sofia	33	19	11	12	21	0	0

În acest moment, veți vedea că tabelul vostru s-a reordonat după criteriile selectate, dar există trei colegi cu același număr de minute petrecute la TV în ziua de luni (Popescu Dan, Popescu Ana și Popescu Virginia). Pe ei cum îi ordonăm? Răspunsul firesc este: alfabetic. Adică, în caz de egalitate, **vom introduce un al doilea criteriu de ordonare**, care să îi ordoneze pe cei trei (în cazul nostru) care au timpi egali, după criteriul „ordonare alfabetică”.

Procedați astfel, fiind poziționați în tabel:

- Apăsați din nou butonul **Sortare (Sort)**;
- Din fereastra **Sortare** apăsați butonul **Adăugare nivel (Add Level)**;
- În câmpul **Apoi după (Then by)** alegeți **Coloană A (Column A)**;
- Apăsați butonul **OK**.

Acum, tabelul, sortat după cele două criterii, arată ca în imaginea alăturată.

Observați că tot tabelul este ordonat crescător după timpii petrecuți la TV în ziua de luni, iar cei trei colegi care au timpi egali au fost ordonați alfabetic (și cum aveau inclusiv numele de familie identic, ordonarea s-a făcut după prenume: Ana, Dan și Virginia).

	A	B	C	D	E	F	G	H
1		Luni	Marti	Miercuri	Joi	Vineri	Sambata	Duminica
2	Dănescu Andreea	5	16	22	13	23	0	0
3	Vasilescu Marian	12	14	33	15	11	0	0
4	Popescu Ana	18	43	82	19	14	90	0
5	Popescu Dan	18	233	41	33	43	0	0
6	Popescu Virginia	18	21	23	18	9	0	0
7	Vrânceanu Vlad	23	43	22	51	11	0	0
8	Avram Sofia	33	19	11	12	21	0	0
9	Mihai Daniel	43	55	12	15	19	45	0
10	Iscrulescu Flavia	44	22	56	10	10	110	0
11	Lazăr Andrei	45	12	112	12	15	0	0

Rețineți

Puteți sorta (ordona) tabelul vostru în funcție de mai multe criterii de sortare odată, fiecare criteriu fiind specificat pe un nivel de sortare. Ordonarea se va realiza ținându-se cont de ordinea în care apar nivelurile de sortare, de sus în jos.

Aplicați și investigați

- Refaceți ordonarea tabelului tot după timpii petrecuți la TV în ziua de luni, dar de această dată în ordine descrescătoare.

Știați că...?

Dacă introduceți un număr într-o celulă (de exemplu, numărul 1 în celula **A1**), iar apoi trageți cu mouse-ul de colțul din dreapta-jos al celei, în jos pe coloană, **ținând apăsată în acest timp și tasta Ctrl**, veți obține în fiecare celulă peste care trageți mouse-ul o valoare cu 1 mai mare decât în celula anterioară. Adică în celula **A2** veți obține automat valoarea 2, în celula **A3**, valoarea 3 și așa mai departe. Se va realiza astfel completarea automată a unei **serii de date** cu valori crescătoare.



Exersați

1. Creați o foaie de calcul, completați un tabel cu 4 coloane și 6 rânduri cu numere diferite, atât întregi, cât și cu zecimale. Formatați tabelul, astfel încât să afișeze toate numerele cu separatorul de mii și cu trei zecimale.
2. De ce tip sunt datele din celulele unui registru, dacă în două dintre celule avem valorile 14:55 și 13:30, iar în a treia celulă rezultatul scăderii lor este 1:25?
 - a. Număr;
 - b. Oră;
 - c. Contabil.
3. Realizați în maximum 3 minute o foaie de calcul cu tablele înmulțirii cu 6, 7, 8, 9 și 10, cu înmulțiri de la 1 la 50 pentru fiecare. Folosiți formule de calcul și multiplicarea conținutului unei celule, cu și fără tasta **Ctrl**, în funcție de necesități.
4. Scrieți, în caiete, **A** (Adevărat) sau **F** (Fals) pentru fiecare propoziție de mai jos:
 - a. Textul dintr-o celulă poate fi formatat să apară chiar și pe verticală. **A F**
 - b. Nu se pot formata caracterele dintr-o celulă cu fonturi și mărimi diferite. **A F**
 - c. Scrierea unei formule de calcul începe întotdeauna cu semnul =. **A F**
5. **Dezbateri.** Inițiați o dezbateri în clasă pe tema „Adăugarea de formule într-un registru de calcul înseamnă automatizarea calculelor și eficiență sporită sau doar o complicație în plus?”.



Portofoliu



Realizați în MS Excel un carnet de note electronic, cu toate materiile pe care le faceți la școală. Puneți coloane pentru note, coloane separate pentru medii și calcularea mediei generale.

5 Modalități de utilizare a registrelor de calcul. Calcule tabelare, funcții



Descoperiți

Am văzut că celulele unei foi de calcul pot conține date de diferite tipuri (număr, text, monedă, dată, oră, procent, fracție etc.) și că putem folosi formule într-o foaie de calcul pentru ușurarea lucrului. Vom vedea împreună câteva modalități practice de utilizare a foilor de calcul și câteva funcții specifice aplicației de calcul tabelar.

▶ Formule de calcul care utilizează operatori aritmetici (+, -, *, /)

Să pornim de la tabelul alăturat.

1. Pentru a calcula suma în celula **C2**, folosim operatorul aritmetic **+**, iar formula este: **=87+33**.

Dacă vrem (ceea ce este de preferat) să folosim referințe de celule, formula este: **=A2+B2**.

2. Pentru a calcula diferența în celula **C5**, folosim operatorul aritmetic **-**, iar formula este: **=100-53**.

Dacă folosim referințe de celule, formula este: **=A5-B5**.

3. Pentru a calcula produsul în celula **C8**, folosim operatorul aritmetic *****, iar formula este: **=9*8**.

Dacă folosim referințe de celule, formula este: **=A8*B8**.

4. Pentru a calcula câtul în celula **C11**, folosim operatorul aritmetic **/**, iar formula este: **=99/9**.

Dacă folosim referințe de celule, formula este: **=A11/B11**.

	A	B	C
1	Primul nr	Al 2-lea nr	SUMA
2	87	33	
3			
4	Primul nr	Al 2-lea nr	DIFERENȚA
5	100	53	
6			
7	Primul nr	Al 2-lea nr	PRODUSUL
8	9	8	
9			
10	Primul nr	Al 2-lea nr	CÂTUL
11	99	9	



Rețineți

Puteți folosi operatorii aritmetici cunoscuți pentru a obține expresii oricât de complexe legate de conținutul celulelor. De exemplu: **=((A1+A2)*(A4-B4))/((A8*B8)+(A11/B11))**.

⊕ Funcții specifice aplicației de calcul tabelar pentru sumă, maxim, minim, medie aritmetică și decizie



1. Funcția SUM – SUMĂ

Descriere: adună valori. Puteți aduna valori individuale, referințe de celule, zone sau o combinație a acestora.

Sintaxa este: **=SUM(număr1, [număr2], ...)**.

De exemplu: **=SUM(A2:A10)** adună valorile celulelor de la **A2** la **A10**.

=SUM(A2:A10, C2:C10) adună valorile din celulele de la **A2** la **A10**, precum și valorile din celulele de la **C2** la **C10**.

=SUM(4,10) întoarce valoarea 14.

=SUM(A2,B3) întoarce suma valorilor celulelor **A2** și **B3**.

2. Funcția MAX – MAXIM

Descriere: întoarce valoarea cea mai mare dintr-un set de valori.

Sintaxa este: **=MAX(număr1, [număr2], ...)**.

De exemplu: **=MAX(4, 99, 43)** întoarce valoarea 99.

=MAX(A2:A10) afișează maximul celulelor de la **A2** la **A10**.

3. Funcția MIN – MINIM

Descriere: întoarce valoarea cea mai mică dintr-un set de valori.

Sintaxa este: **=MIN(număr1, [număr2], ...)**.

De exemplu: **=MIN(4, 99, 43)** întoarce valoarea 4.

=MIN(A2:A10) afișează minimul celulelor de la **A2** la **A10**.

4. Funcția AVERAGE – MEDIE ARITMETICĂ

Descriere: calculează media aritmetică a unui grup de numere.

Sintaxa este: **=AVERAGE(număr1, [număr2], ...)**.

De exemplu: **=AVERAGE(10, 2, 18)** întoarce valoarea 10.

=AVERAGE(A2:A10) afișează media aritmetică a numerelor corespunzătoare celulelor de la **A2** la **A10**.

5. Funcția IF – DACĂ

Descriere: este funcția de decizie. Aceasta vă permite să efectuați comparații logice între o valoare și valoarea la care vă așteptați.

Așadar, o funcție **IF** poate avea două rezultate. Primul rezultat apare când comparația este TRUE (ADEVĂRAT), al doilea când este FALSE (FALS).

Sintaxa este:

=IF(test_logic, [valoare_pt_ADEVĂRAT], [valoare_pt_FALS])

De exemplu: în celula **D4** din tabelul alăturat avem calculat un punctaj la un examen. Condiția pentru a promova examenul este ca punctajul să fie cel puțin 80.

=IF(D4>=80, "PROMOVAT", "RESPINS") întoarce rezultatul „PROMOVAT”, dacă punctajul calculat în celula **D4** este mai mare sau egal cu 80, dar întoarce „RESPINS”, dacă punctajul este mai mic de 80.

	A	B	C	D	E
		Punctaj 1	Punctaj 2	TOTAL	
1	NUME, PRENUME				PROMOVAT/RESPINS
2	Popescu Dan	45	33	78	RESPINS
3	Vrânceanu Vlad	45	46	91	PROMOVAT
4	Mihai Daniel	42	41	83	PROMOVAT
5	Isculescu Flavia	39	40	79	RESPINS
6	Popescu Ana	38	22	60	RESPINS
7	Lazăr Andrei	45	55	100	PROMOVAT
8	Popescu Virginia	44	44	88	PROMOVAT
9	Vasilescu Marian	39	47	86	PROMOVAT
10	Dănescu Andreea	38	37	75	RESPINS
11	Avram Sofia	48	49	97	PROMOVAT



Rețineți

Puteți utiliza toate aceste funcții împreună cu operatorii aritmetici învățați pentru a crea expresii mai complexe. De exemplu: **=Sum(B1,B2)*MIN(C1,C2,C3)/MAX(C4,C5)/IF(C11=49,2,3)**.



Aplicați și investigați

1. Faceți un tabel cu o listă de cumpărături: numele produselor (în coloana **A**), cantitatea (în coloana **B**) și prețurile lor în lei, cu TVA inclus (în coloana **C**). TVA-ul reprezintă „Taxa pe Valoarea Adăugată” și este de 19%. Calculați:

- pe coloana **D**: prețul fără TVA al fiecărui produs în parte, după următoarea formulă:
=preț cu TVA/1.19 (concret, în tabelul vostru introduceți în celula **D2** formula **=C2/1.19** și replicați această formulă pe toate celulele din coloană).
- pe coloana **E**: valoarea TVA pentru fiecare produs în parte (preț cu TVA/preț fără TVA);
- pe coloana **F**: care este valoarea totală cheltuită pentru fiecare produs în parte (cantitate*preț).

Pe coloanele **G, H, I, J** și **K** îmbinați toate celulele astfel încât valoarea celulei îmbinate să fie comună pentru toate produsele. Afișați următoarele date în aceste coloane:

- în coloana **G**: cât s-a cheltuit în total (suma);
- în coloana **H**: care este prețul cel mai mic al produselor (minimul);
- în coloana **I**: care este prețul cel mai mare al produselor (maximul);
- în coloana **J**: care este prețul mediu al produselor (media aritmetică);
- în coloana **K**: afișați mesajul „V-AȚI ÎNCADRAT ÎN BUGET”, dacă suma totală în lei nu depășește valoarea de 10 000 de lei, altfel afișați mesajul „BUGET DEPĂȘIT”.

2. Putem utiliza foile de calcul cu funcții și formule pentru rezolvarea unor probleme la fizică, matematică, chimie, geografie etc.? Răspunsul este DA. Să vedem câteva exemple.

2.1. Rezolvarea unei probleme de fizică

Să luăm următoarea problemă dintr-o culegere de fizică: *Într-un experiment de monitorizare a balenelor, biologii au constatat că o balenă a coborât la adâncimea $h = 3$ km, stabilind un record de scufundare. Se știe că durata scufundării a fost $t = 137$ min. Calculați viteza medie de scufundare a balenei.*

Modalitate de rezolvare:

- faceți un tabel precum cel din figura alăturată;
- în celula **B2**, calculați distanța în metri (**=B1*1000**);
- în celula **B5**, calculați timpul (durata scufundării) în secunde (**=B4*60**);
- știind că formula vitezei este $viteza = \frac{distanța}{timp}$ (în cazul nostru, distanța este adâncimea până la care s-a scufundat balena), în celula **B8** calculați viteza medie de scufundare a balenei (**=B2/B5**) și veți afla rezultatul final: 0,36 m/s.

	A	B	C
1	Adâncimea h:	3	km
2		3000	m
3			
4	Timpul t:	137	min
5		8220	s
6			
7	V=h/t	=B2/B5	
8		0.36	m/s

2.2. Rezolvarea unei probleme de matematică

Se dă următoarea problemă de matematică: *Stabiliți care dintre numerele următoare sunt divizibile cu 3: 21, 100, 51, 14, 55, 88, 65, 77, 13, 79.*

Modalitate de rezolvare:

- faceți un tabel precum cel din figura alăturată;
- în coloana **A** introduceți toate numerele din problemă pe care trebuie să le testați;
- în coloana **B** calculați rezultatul împărțirii fiecărui număr la trei ($=A2/3$) și replicați aceeași formulă până la ultimul număr (79, din rândul 11);
- în acest moment, puteți spune care numere sunt divizibile cu trei (cele care au dat un număr întreg la împărțire) și care nu (cele care nu au dat număr întreg în urma operației).

	A	B	C
1	Numarul	Nr/3	DIVIZIBIL/NEDIVIZIBIL
2	21	7	DIVIZIBIL
3	100	33.33333	NEDIVIZIBIL
4	51	17	DIVIZIBIL
5	14	4.666667	NEDIVIZIBIL
6	55	18.33333	NEDIVIZIBIL
7	88	29.33333	NEDIVIZIBIL
8	65	21.66667	NEDIVIZIBIL
9	77	25.66667	NEDIVIZIBIL
10	13	4.333333	NEDIVIZIBIL
11	79	26.33333	NEDIVIZIBIL



Descoperiți

Pentru a înțelegi problema de matematică 2.2, descoperiți în cele ce urmează o altă funcție interesantă:

Funcția **INT** – ÎNTREG

Descriere: rotunjește prin lipsă la cel mai apropiat număr întreg.

Sintaxa este: **=INT(număr)**

De exemplu: **=INT(3.44)** întoarce valoarea 3.



Aplicați și investigați

- La problema 2.2. de mai sus, automatizați partea de decizie. În dreptul numerelor divizibile cu 3 scrieți **DIVIZIBIL**, în celelalte cazuri, **NEDIVIZIBIL**.

Puteți obține direct în coloana **C** calitatea unui număr de a fi sau nu divizibil cu trei, astfel:

- introduceți în celula **C2** funcția **=IF(INT(B2)=B2, "DIVIZIBIL", "NEDIVIZIBIL")**;
- replicați această funcție până în celula **C11** și veți obține automat **DIVIZIBIL** sau **NEDIVIZIBIL** pentru fiecare număr cerut de problemă.



Exersați

1. Utilizând o foaie de calcul în MS Excel, calculați suma numerelor de la 1 la 10, apoi suma numerelor de la 1 la 100.

2. Faceți corespondența între funcțiile MS Excel din prima coloană și definițiile lor din a doua coloană:



- | | |
|-------------------|--|
| a. SUM | I. funcție de decizie |
| b. AVERAGE | II. funcție care întoarce valoarea cea mai mică dintr-un set de valori |
| c. MAX | III. funcție care calculează media aritmetică a unui grup de numere |
| d. MIN | IV. funcție care înmulțește valori |
| e. IF | V. funcție care întoarce valoarea cea mai mare dintr-un set de valori |
| | VI. funcție care adună valori |

3. Faceți un tabel în MS Excel cu toți colegii din clasa voastră, adăugând înălțimea fiecăruia în centimetri, și găsiți care este cel mai înalt, folosind funcția corespunzătoare.

4. Care este funcția pentru calculul mediei în MS Excel? Dar sintaxa ei?

5. Realizați un tabel în MS Excel cu situația școlară la matematică a tuturor elevilor din clasa voastră, calculați mediile lor și determinați media minimă, respectiv pe cea maximă.

6. Faceți un tabel în MS Excel cu un traseu de făcut prin țară cu bicicleta într-o excursie de 10 zile. Introduceți orașele de vizitat, distanțele de parcurs până la fiecare oraș și timpii estimați pentru parcurgerea fiecărei etape. Calculați și afișați: distanța totală, timpul total de mers, etapa cu distanța minimă și etapa cu distanța maximă.



6 Modalități grafice de utilizare a registrelor de calcul



Descoperiți

Am văzut că, folosind un program de calcul tabelar, putem face simplu și rapid o multitudine de calcule. Putem chiar rezolva și probleme la diferite materii (matematică, fizică etc.).

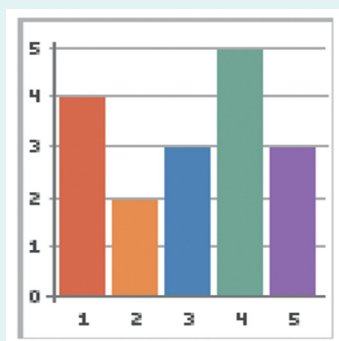


Rețineți

O facilitate foarte importantă a programelor de calcul tabelar este aceea de a crea reprezentări grafice (numite și „diagrame”) care să ne ajute să ne facem o imagine clară și rapidă despre diferite date din tabele.

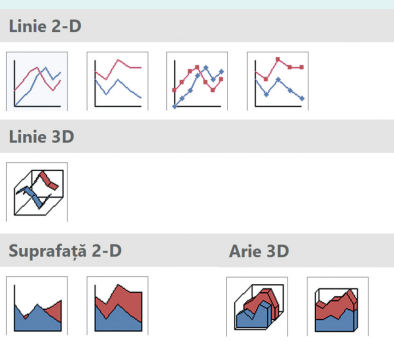


Programul MS Excel ne pune la dispoziție mai multe modele de grafice predefinite:



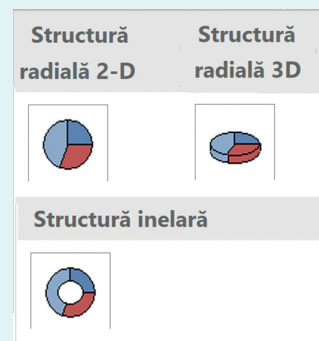
Diagramă coloană sau bară

Utilizați acest tip de diagramă pentru a compara vizual datele din câteva categorii.



Diagramă cu linii sau arie

Utilizați acest tip de diagramă pentru a arăta tendințele în timp sau pe categorii.



Diagramă radială sau inelară

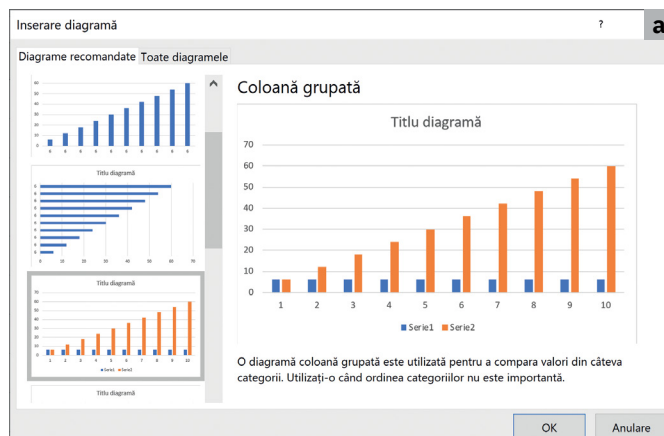
Utilizați acest tip de diagramă pentru a reprezenta grafic proporțiile unui întreg.



Sunt și alte tipuri de diagrame disponibile; toate se regăsesc în fila **Inserare** (*Insert*), grupul **Diagrame** (*Charts*). Faceți un clic pe săgeata din dreapta-jos a grupului și descoperiți atât lista tuturor diagramelor disponibile, cât și diagramele recomandate pentru tabelul vostru în particular (*imaginea a*). O diagramă se actualizează automat dacă se modifică datele din tabelul din care a fost generată.

Cum pot fi create grafice (diagrame) în Excel, în doar patru pași simpli:

1. Faceți un tabel în care puneți datele care vor fi reprezentate grafic.
2. Selectați tot tabelul.
3. Accesați fila **Inserare** (*Insert*), grupul **Diagrame** (*Charts*).
4. Alegeți graficul care se potrivește cel mai bine situației și care arată așa cum doriți din fereastra **Inserare diagramă** (*Insert Chart*) care se deschide.



Aplicați și investigați

1. Să luăm următorul scenariu: avem o școală cu 1 300 de elevi, facem un tabel în care trecem câte medii avem în școală din fiecare (de la 1 la 10), dorind să reprezentăm grafic distribuția mediilor.

	A	B	C	D	E	F	G	H	I	J	K
1	Media	1	2	3	4	5	6	7	8	9	10
2	Numărul mediilor	0	0	0	36	132	234	243	292	267	96

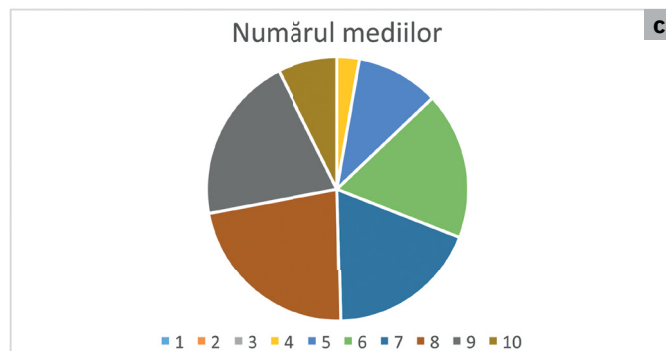
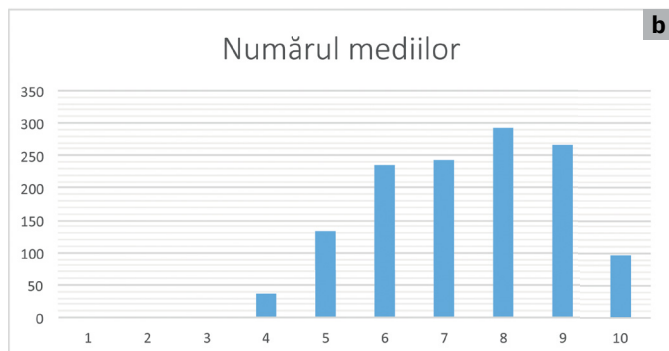
Modalitate de rezolvare:

Selectați întregul tabel, apoi mergeți în fila **Inserare (Insert)** și, din grupul **Diagramme (Charts)**, alegeți diagrama cu bare de la **Diagramme recomandate (Recommended Charts)**. Vi se va afișa o diagramă precum cea de mai jos (imaginea b).

Veți obține în acest fel o diagramă din care se poate vedea cu ușurință repartizarea mediilor din școală. Puteți observa imediat că mediile de 8 sunt cele mai numeroase și mediile de 4 sunt mai rare, mult mai puține comparativ cu celelalte.

Dacă veți dori să evidențiați ponderea mediilor de 4 din totalul mediilor din școală, de exemplu, este potrivită o diagramă radială, pe care o puteți alege tot de la **Diagramme recomandate (Recommended Charts)** sau direct de la **Diagramme cu structură radială (Pie)**.

Observați că ponderea mediilor de 4 (reprezentate cu galben) din întreg (din totalul mediilor) este mică (imaginea c).



2. Aveți următoarea problemă de fizică: *Dan a înregistrat într-un tabel poziția automobilului în care se afla față de casa lui, la diferite momente. Datele obținute de el se regăsesc în tabelul de mai jos.*

t(s)	0	1	2	3	4	5	6	7	8
x(m)	10	11.5	13	14.5	16	17.5	19	20.5	22

Reprezentați graficul mișcării și calculați viteza automobilului în intervalul de timp considerat.

Modalitate de rezolvare:

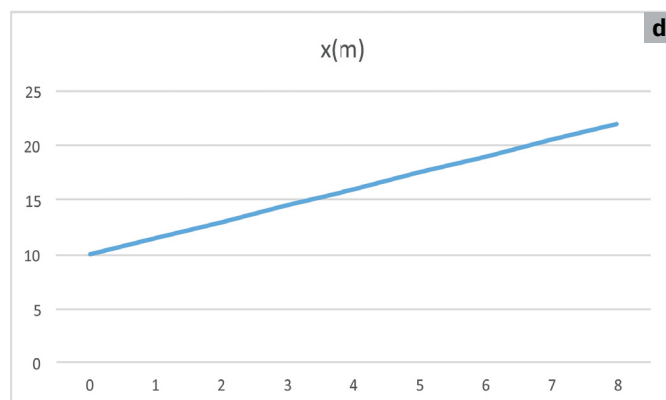
Tot ce aveți de făcut pentru reprezentarea graficului mișcării este să introduceți datele într-o foaie de calcul în MS Excel, să selectați întregul tabel și să creați graficul corespunzător tabelului (imaginea d).

Pentru calcularea vitezei, trebuie să cunoașteți formula vitezei, $v = d/t$. Puteți introduce într-o celulă a foii de calcul formula:

$$=(22-10)/8.$$

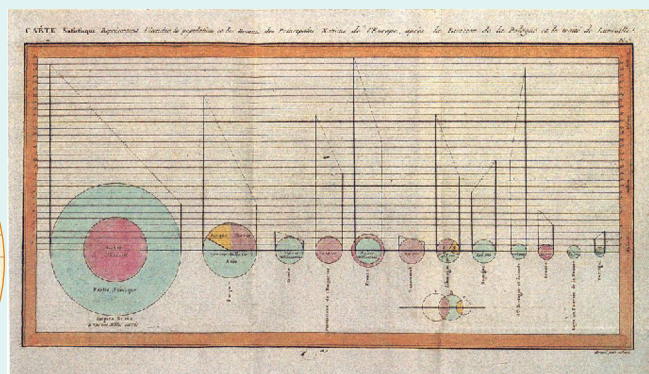
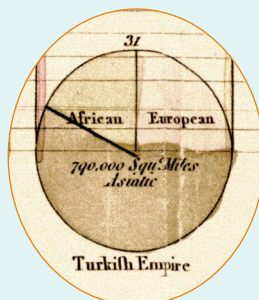
Va rezulta $v = 1,5$ m/s.

Puteți, de asemenea, să folosiți adresarea celulelor pentru calculul vitezei.

**Știați că...?**

O diagramă radială mai poartă numele sugestiv de diagramă **plăcintă (pie chart)**, datorită formei sale circulare și a împărțirii în „felii” pentru a ilustra proporțiile numerice.

Primele grafice de acest fel datează din anul 1801 și au fost prezentate de către William Playfair, inginer și economist politic scoțian, în lucrarea sa *Statistical Breviary*. Una dintre diagramele sale arată proporțiile Imperiului Otoman localizat în Asia, Europa și Africa, înainte de 1789.



Primele grafice plăcintă (desen întreg și detaliu)

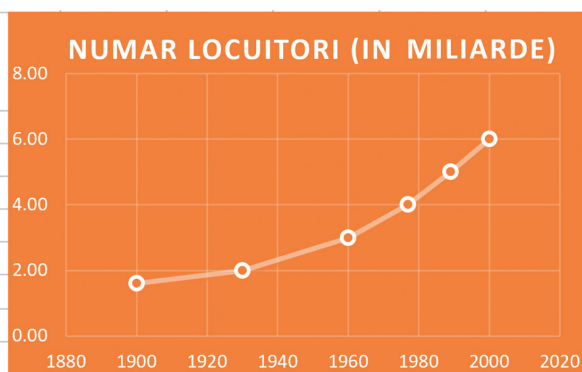
3. La geografie, trebuie să reprezentați grafic evoluția numărului de locuitori ai planetei. La o căutare rapidă pe Internet, veți găsi astfel de informații: *În ceea ce privește creșterea populației, a fost nevoie de întreaga istorie ca populația lumii să atingă 1,6 miliarde în anul 1900, totalul acesteia ajungând în 1930 la două miliarde. Al treilea miliard s-a adăugat până în anul 1960, iar al patrulea miliard până în anul 1977. În doar 12 ani, populația a ajuns la 5 miliarde, respectiv până în anul 1989, depășind 6 miliarde în anul 2000.*

Modalitate de rezolvare:

Pentru a reprezenta grafic aceste informații, faceți un tabel în care să treceți, pentru fiecare an menționat în text, numărul de locuitori ai planetei noastre.

Selectați, apoi, întregul tabel și inserați diagrama prin puncte, dintre cele recomandate de MS Excel. Veți obține o reprezentare grafică sugestivă pentru creșterea populației planetei, așa cum vedeți în imaginea alăturată.

Populația lumii	Număr locuitori (în miliarde)
1900	1.60
1930	2.00
1960	3.00
1977	4.00
1989	5.00
2000	6.00



Exersați

1. Cum vă găsiți lucrurile în casă? Analizați-vă cum procedați atunci când căutați ceva.



- Căutați până găsiți singuri;
- Renunțați;
- MAMAAAAAAA!

Treceți într-un tabel realizat în Excel aceste situații dintr-o săptămână, apoi reprezentați grafic rezultatul. Cum vi se pare?

Indicație: Realizați în Excel un tabel cu două coloane și patru rânduri, precum cel alăturat, completați numerele pentru fiecare caz, apoi mergeți în fila **Inserare (Insert)** și, de la grupul **Diagrame (Charts)**, alegeți o diagramă **Structură radială**.

Soluția aleasă	Nr de cazuri din fiecare soluție
Caut și găsesc singur	
Renunț	
MAMAAAAAAA!	

2. Într-un tabel cu date creat în MS Excel, pentru care ați realizat și o diagramă, modificați datele dintr-o celulă. Ce se întâmplă cu diagrama? Marcați răspunsul corect:

- diagrama a fost ștearsă;
- diagrama a fost modificată în funcție de noul conținut al celulei;
- diagrama a fost dublată, cea nouă fiind actualizată după noul conținut al celulei modificate.

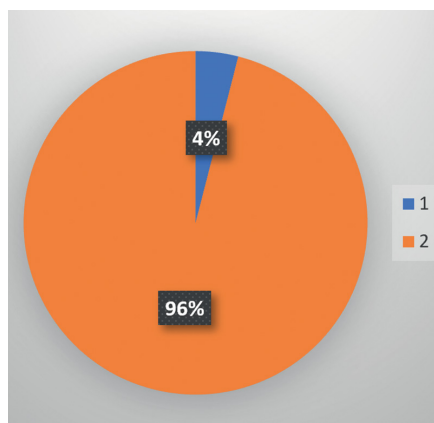
3. Realizați un tabel cu două rânduri și zece coloane în MS Excel, completați-l cu date numerice oarecare, apoi creați două diagrame (una cu bare și una radială) pentru reprezentarea grafică a datelor din tabel.

4. Dați două exemple de diagrame cu structură radială din MS Excel.

5. Priviți diagrama alăturată („Diagramă de test”). Cum o interpretați?

6. Răspundeți la următoarele întrebări legate de realizarea modelelor grafice (diagramelor) în MS Excel:

- Unde se găsește grupul **Diagrame (Charts)**?
- Când este potrivit să utilizați diagrama radială?
- Când este potrivit să utilizați diagrama cu linii?



Portofoliu



Evoluția populației României

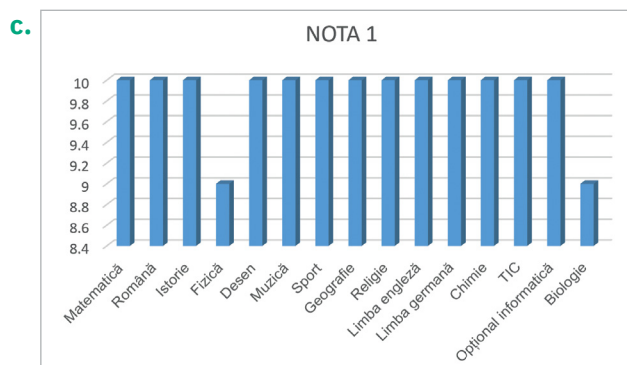
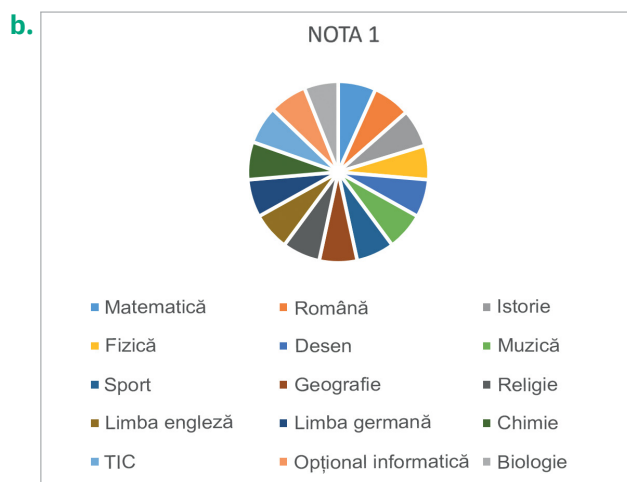
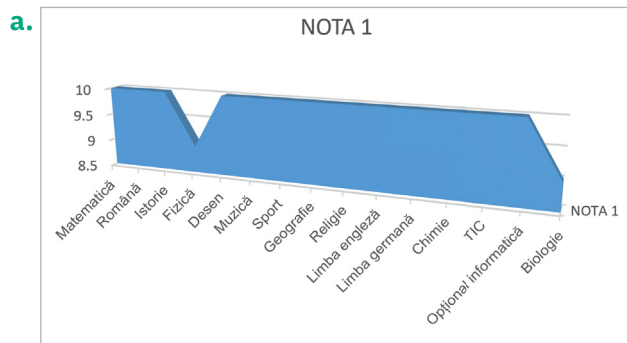
Știm că în 1980 populația României era de aproximativ 21 000 000 de locuitori. Realizați un tabel cu două coloane („Anul” și „Populația României”). În prima coloană completați o serie de ani, începând de la 1980, din 3 în 3, până la 2037. În a doua coloană completați numărul de locuitori ai României, pornind de la populația existentă în 1980, cu un procent de scădere de 10% la fiecare 3 ani (folosiți formula de calcul potrivită). Realizați apoi diagrama care reprezintă grafic evoluția populației României în această perioadă. Cum ar arăta diagrama dacă procentul de scădere ar fi de 3%?

Recapitulare

1. Ce este MS Excel?
 - a. un limbaj de programare;
 - b. un program de calcul tabelar;
 - c. un program de editat texte.
2. Numiți și descrieți trei dintre filele aplicației MS Excel.
3. În MS Excel, adresa **C3** se referă la:
 - a. primele 3 celule din coloana **C**;
 - b. celula aflată pe linia a treia, coloana **C**;
 - c. pagina **C3** din registrul de calcul.
4. Coloanele în MS Excel se numerotează cu:
 - a. cifre romane;
 - b. litere mari;
 - c. cifre arabe.
5. Pentru introducerea formulelor în celule în MS Excel, trebuie să:
 - a. începem formula cu semnul „=”;
 - b. scriem formula, urmată de semnul „=”;
 - c. scriem direct formula de calcul.
6. O celulă dintr-o foaie de calcul a MS Excel poate conține:
 - a. texte;
 - b. formule;
 - c. numere.
7. Funcția **SUM** din MS Excel se folosește la:
 - a. realizarea sumei tuturor celulelor din registrul de calcul;
 - b. realizarea sumei celulelor furnizate ca argument al funcției;
 - c. realizarea sumei celulelor care conțin valori exprimate în Euro.
8. Extensia unui fișier creat cu MS Excel este:
 - a. .exe;
 - b. .xlsx;
 - c. .doc.
9. Care dintre următoarele funcții calculează media aritmetică a celulelor **C1**, **C2** și **C3** în MS Excel?
 - a. **=AVERAGE(C1:C3)**;
 - b. **=AVERAGE(C1, C4)**;
 - c. **=SUM(C1, B3)/3**.
10. Realizați un tabel în MS Excel cu numele a 10 fluvii din lume și cu lungimile lor, apoi ordonați-le descrescător, în funcție de lungime.

11. Realizați în MS Excel un tabel cu inventarul sertarului vostru cu rechizite (creioane, pixuri, carioci, caiete etc.) și calculați totalul obiectelor, folosind funcția corespunzătoare.

12. Faceți corespondența corectă între diagrame și numele lor, după modelul *d-4*:



1. **Diagramă coloană 3D** (utilizată pentru a compara valori din mai multe categorii)
2. **Diagramă arie 3D** (utilizată pentru a reprezenta tendințele pe categorii)
3. **Diagramă cu structură radială** (utilizată pentru a reprezenta proporțiile unui întreg)
4. **Diagramă cu linii** (utilizată pentru a arăta tendințele în timp)

1. Ce este un program de calcul tabelar?
2. În care dintre următoarele tipuri de date poate fi formatată o celulă dintr-o foaie de calcul în MS Excel?
 - a. text;
 - b. număr;
 - c. monedă;
 - d. dată;
 - e. procent;
 - f. fracție.

3. Realizați, în caiete, corespondența dintre noțiunile date și descrierile lor, din tabelul de mai jos:

a. Bara de titlu	1. include toate comenzile aplicației MS Excel grupate pe file, pe tipuri de acțiuni
b. Panglica de butoane	2. include eticheta foii de calcul creată în registrul aflat în lucru
c. Caseta de nume	3. afișează numele celulei în care suntem poziționați
d. Zona de lucru	4. foaia de calcul, zona în care introducem date, formule, realizăm diagrame
	5. conține butoanele de minimizare, restaurare, închidere, salvare, anulare acțiune, refacere acțiune și titlul registrului MS Excel

4. Alegeți, din lista următoare, elementele principale ale interfeței utilizator a MS Excel:
 - a. Bara de defilare verticală;
 - b. Panglica;
 - c. Zona de lucru;
 - d. Bara de desenare;
 - e. Instrument de panoramare;
 - f. Eticheta foii de calcul.
5. Dintre aplicațiile următoare, transcrieți-le în caiete pe cele care sunt programe de calcul tabelar:
 - a. MS Excel;
 - b. MS Outlook;
 - c. Libre Office Calc;
 - d. Excel Online;
 - e. Ashampoo PlanMaker;
 - f. MS Word.

6. Numiți două avantaje practice ale stocării unui registru de calcul în cloud.

7. Scrieți **DA** sau **NU** pentru fiecare dintre afirmațiile de mai jos:

- a. Textul dintr-o celulă poate fi: drept (simplu), aplecat, îngroșat, subliniat. DA / NU
- b. Butonul **Sortare** din fila **Aspect pagină** sortează datele din tabel după criteriile selectate. DA / NU
- c. Funcția **SUM** adună valori. Se pot aduna valori individuale sau referințe de celule. DA / NU

8. Completați, în caiete, enunțurile următoare, folosind cuvintele potrivite:

- a. Fila **Inserare** (*Insert*) conține necesare pentru adăugarea de alte elemente de conținut într-o foaie de, spre exemplu,, ilustrații,, hărți, video, legături către diferite pagini web,, simboluri etc.
- b. Pentru a tipări registrul de, trebuie să alegeți comanda, aflată în fila **Fișier** (*File*). Documentul se poate tipări la o instalată în prealabil.
- c. Funcția **IF** este funcția de Este una dintre cele mai populare funcții din și vă permite să efectuați comparații între o valoare și valoarea la care vă așteptați. Așadar, o funcție **IF** poate avea rezultate.

9. Dați trei exemple de operații elementare din MS Excel pentru lucrul cu un registru de calcul.

10. Ce sunt și la ce folosesc diagramele din MS Excel?

Punctaje:

1 – 10 puncte;	6 – 10 puncte;	Insuficient: 0-50 de puncte
2 – 10 puncte;	7 – 10 puncte;	Suficient: 51-60 de puncte
3 – 10 puncte;	8 – 10 puncte;	Bine: 61-80 de puncte
4 – 10 puncte;	9 – 10 puncte;	Foarte bine: 81-90 de puncte
5 – 10 puncte;	10 – 10 puncte.	Excelent: 91-100 de puncte

Punctaj maxim: 100 de puncte. ● Timp de lucru: 50 de minute.
Răspunsurile se găsesc la finalul manualului.

CUM VĂ APRECIĂȚI?

1. Evaluați-vă singuri rezultatele utilizând grila de punctaje.
2. Scrieți cum vă simțiți după ce ați rezolvat acest test, alegând unul dintre cuvintele de mai jos:
 - nemulțumit(ă);
 - mulțumit(ă);
 - încântat(ă).



Concurs: Planificarea unui eveniment

Descrierea concursului

În cadrul concursului **Planificarea unui eveniment**, participanții vor lucra în echipe de 2-3 elevi și vor avea de realizat un proiect în MS Excel, în care să planifice un eveniment la alegere (excursie, petrecere, tabără etc.), gestionând costurile și evidența participanților.

Timpul de lucru

Alocat de cadrul didactic.

Cerințe pentru realizarea proiectului

1. Fiecare echipă trebuie să creeze o listă de minimum 6 tipuri de costuri pentru fiecare participant în parte (ex.: transport, mese, cazare, bilete la atracții turistice etc.). Lista trebuie să conțină coloane pentru: *Descriere cost/participant, Sumă*.
2. Fiecare echipă trebuie să creeze o listă de cel puțin 10 participanți. Lista va conține coloane pentru: *Nume, Prenume, Vârstă, Cost participare, Sumă achitată, Status plată* (achitat integral/achitat parțial/neachitat).
3. Se va calcula costul total de participare care trebuie achitat de fiecare persoană în parte.
4. Utilizând **Inserare -> Diagrame recomandate**, se va genera un grafic ce va afișa câți participanți au plătit integral, câți participanți au plătit parțial și câți nu au achitat deloc.



Cum se face evaluarea

Fiecare echipă își va prezenta proiectul prin intermediul unei prezentări MS PowerPoint, iar acesta va fi evaluat și va primi un punctaj ținându-se cont de următorul barem:

Criteriu	Punctaj (maxim)
Structura corectă a tabelelor	30 de puncte
Utilizarea funcțiilor SUM, IF	15 puncte
Aspect vizual și organizare a fișierului	15 puncte
Prezentare clară și completă a informațiilor	10 puncte
Creativitate și originalitate	10 puncte
Respectarea cerințelor și predarea la timp	10 puncte
Puncte din oficiu	10 puncte
Total:	100 de puncte

Premii și recunoaștere

- Echipa cu cel mai bun proiect va primi **Diploma de excelență în planificare digitală**, iar toate celelalte echipe participante vor primi diplome de participare.
- Cele mai bune trei proiecte vor fi prezentate în cadrul unui eveniment special.

Autoevaluarea proiectului

1. Am respectat cerințele proiectului:
 - a. În întregime
 - b. Parțial
2. În timpul de lucru alocat am terminat proiectul:
 - a. În întregime
 - b. Parțial
3. Nivelul de documentare în vederea elaborării proiectului este:
 - a. Foarte bun
 - b. Bun
 - c. Slab
4. Comunicarea cu colegii a fost:
 - a. Foarte bună
 - b. Bună
 - c. Slabă
5. Nivelul de creativitate la realizarea proiectului a fost:
 - a. Foarte bun
 - b. Bun
 - c. Slab
6. Proiectul le-a plăcut colegilor:
 - a. În mare măsură
 - b. În măsură normală
 - c. În mică măsură

U2

Pagini web



Lecția 1	36-39	Pagini web – exemple, analiză, structură
Lecția 2	40-44	Aplicații specializate pentru crearea paginilor web – elemente de interfață și facilități
Lecția 3	45-50	Crearea unei pagini web. Publicarea unui site de o pagină
Lecția 4	51-53	Construirea unui site Internet cu mai multe pagini. Lucrul colaborativ
Recapitulare	54	
Autoevaluare	55	
Proiect	56	Crearea unui site al clasei

1 Pagini web – exemple, analiză, structură



Amintiți-vă

Internetul este sistemul global de rețele de computere interconectate, care utilizează un anumit protocol de comunicare (TCP/IP) pentru a conecta dispozitivele din întreaga lume. Este o rețea de rețele.

O **pagină web** este o resursă din spațiul web (WWW) din Internet, de obicei în format HTML, și având hiperlinkuri (hiperlegături) pentru navigarea simplă (cu un singur clic de mouse) de la o pagină sau secțiune de pagină la alta. Pagina web se numește astfel („pagină”), deoarece se aseamănă într-o oarecare măsură cu o pagină de ziar.

Un **site web** este o grupare pe o anumită temă a mai multor pagini web care sunt conectate între ele prin hiperlinkuri. Există site-uri web care conțin o singură pagină web, acestea fiind cazuri particulare.



Descoperiți

Cu toții „navigăm” pe Internet, sărim cu ușurință de la o pagină web la alta, de la un site la altul, în căutarea informațiilor dorite. Site-uri cu informații, enciclopedii, site-uri de sport, site-uri de știri, bloguri, site-uri cu jocuri, cu filme, site-uri de modă, magazine online sau orice altceva, toate ne sunt foarte la îndemână.

Dați cinci exemple de site-uri pe care le vizitați de obicei. Ce tip de informații găsiți aici?

Se nasc, însă, mai multe întrebări. Cum și cu ce programe anume se face un site Internet? Care sunt elementele specifice unei pagini web? La toate aceste întrebări și la multe altele legate de subiectul „site-uri web” vs. „pagini web” vom găsi răspunsurile în paginile următoare.

Să vedem împreună care sunt elementele de structură ale unei pagini web. Cel mai bine ar fi să analizăm câteva exemple, să vizităm pagini web aparținând unor site-uri diferite.

Deschideți și analizați cu atenție următoarele site-uri (puteți, de asemenea, deschide orice alte pagini pe care voi le utilizați de obicei sau unele noi, pe care le deschideți acum pentru prima dată). Alegeți două dintre acestea și notați ce elemente sunt comune în alcătuirea lor (ce se află în partea de sus, ce se află în partea de jos):

- **www.manuale.edu.ro** – site-ul unde găsim toate manualele noastre digitale
- **www.aboutyou.ro** – un site de modă, atât pentru fete, cât și pentru băieți
- **www.spotify.com** – una dintre sursele noastre de muzică, unde putem găsi și asculta muzica favorită
- **www.scratch.mit.edu** – site-ul mediului de programare Scratch
- **www.brainly.ro** – site-ul de ajutor la teme, în care o comunitate mare de elevi, părinți și profesori își împărtășesc cunoștințele
- **www.pbInfo.ro** – un site care conține o colecție de lecții și probleme de informatică, grupate pe ani de studii; site-ul permite postarea rezolvărilor de către utilizatorii înregistrați și realizează automat corectarea acestora.
- **www.brainpop.com** – un site pe care putem învăța despre evenimente istorice, știință și chiar economie, cu ajutorul unui roboțel simpatic și al prietenilor săi, care ne prezintă totul sub formă de animații și dialoguri captivante (*imaginea a*)

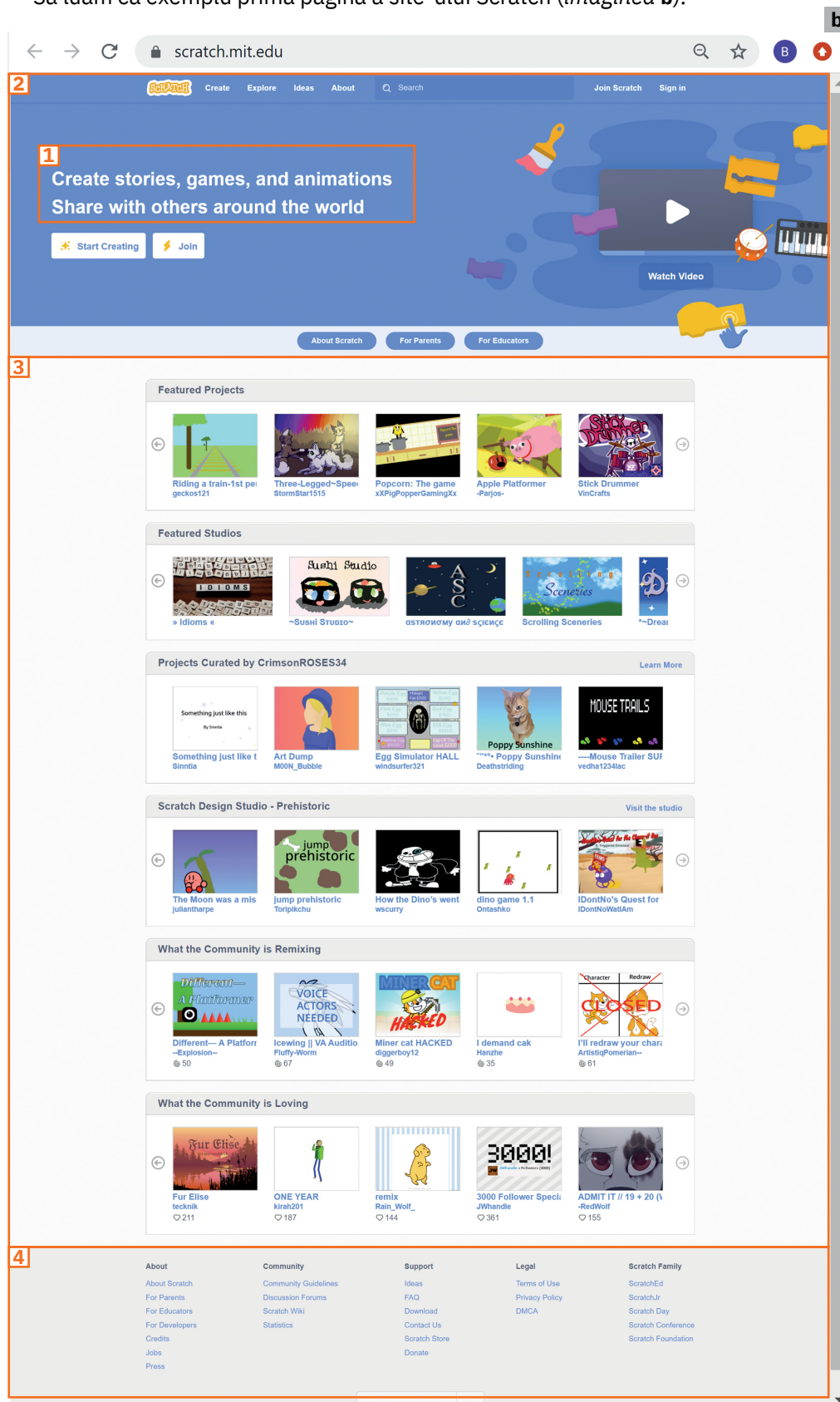


The screenshot shows the BrainPOP website with a dark teal background. At the top, there's a navigation bar with links like 'Apps', 'Gmail', 'YouTube', 'Maps', 'Imported', 'Optimizare site...', 'Google Play Console', 'MBA Jr', and 'Online Courses - U...'. Below this is a search bar with the text 'Search BrainPOP'. The main content area features a large promotional banner for 'Save for the Holidays' with '50% OFF' in large white letters. The banner also includes the text 'Don't Miss Out!', 'Take advantage of our special pricing through December 31. You'll get access to 850+ standards-aligned topics for all subjects and grades.', and buttons for 'Buy Now' and 'Request a Demo'. A cartoon orange mug character is visible in the bottom right corner of the banner.

▶ Vizitând paginile web ale site-urilor de mai sus, observăm că, în general, o pagină web este formată din câteva **elemente de structură** principale:

1. titlul paginii;
2. antetul paginii;
3. corpul paginii;
4. subsolul paginii.

Să luăm ca exemplu prima pagină a site-ului Scratch (*imaginea b*).



Știați că...?



- Viziunea lui Tim Berners-Lee (inventatorul World Wide Web) pentru ceea ce urma să devină web, înaintată spre acceptare șefului său direct, Mike Sendall, în martie 1989, într-un document numit „Managementul informației: Propunere”, nu a fost acceptată imediat. Acesta nota pe prima pagină a documentului: „Vag, dar interesant.”



Tim Berners-Lee

- Primul site web din lume a fost realizat în cadrul CERN (European Organization for Nuclear Research/Organizația Europeană de Cercetare Nucleară) de către Tim Berners-Lee și a fost lansat la 6 august 1991. El era dedicat proiectului WWW (World Wide Web) și rula pe calculatorul lui Berners-Lee, NeXT. Acest site poate fi accesat și astăzi la adresa: <http://info.cern.ch/hypertext/WWW/TheProject.html>.



► **Lucrați în perechi.** Observați zonele mai sus menționate, clar delimitate pe această pagină. Comparați antetul, corpul și subsolul paginii web din punctul de vedere al dimensiunii acestora. Ce elemente sunt prezente în antet și, respectiv, în subsolul paginii exemplificate?

Observați titlul acestei pagini și poziționarea sa. În jurul lui este lăsată o zonă liberă tocmai pentru a ieși în evidență ușor.



Rețineți

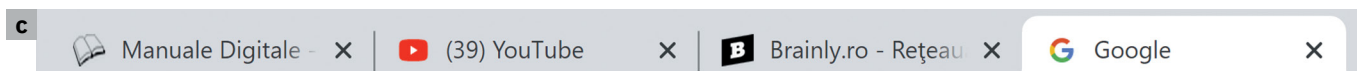
În general, în **antet** găsim **titlul paginii** și elemente de identitate vizuală a paginii.

În **corpul paginii** regăsim informația principală. Această zonă este cea mai importantă parte a paginii web, ea ocupând cel mai mult spațiu din pagină, oferind informație consistentă.

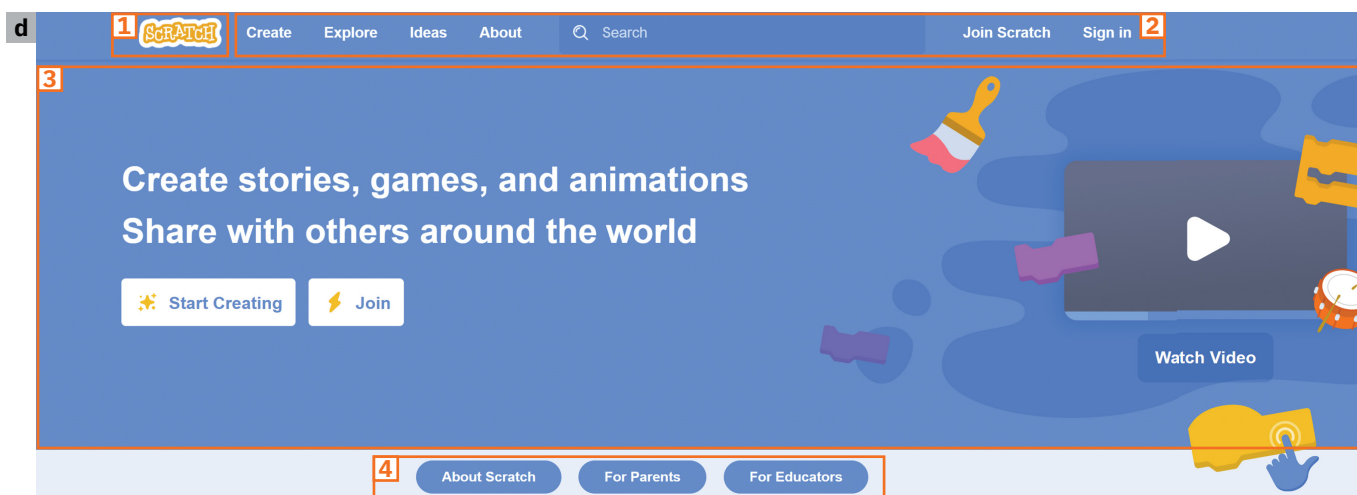
Subsolul este ultima parte a paginii. El poate să conțină câteva informații sumare: un meniu secundar, date de contact și linkuri către platforme sociale.

O pagină web poate exista și poate funcționa fără oricare dintre aceste trei elemente (antet, corp, subsol), dar va fi ca o compunere căreia îi lipsește fie introducerea, fie cuprinsul, fie încheierea.

Se mai poate identifica un element interesant, numit **tag**, care anunță conținutul unei pagini web deschise și apare alături de o iconiță sugestivă, în fila browserului corespunzătoare paginii deschise a site-ului web (*imaginea c*).



Dacă analizăm în amănunt, în zona antetului mai observăm câteva elemente care se regăsesc la majoritatea site-urilor, respectându-se un fel de șablon (*imaginea d*).



1. logo;

2. meniuri;

3. zona imaginii sau a imaginilor de antet (după cum observați, în această zonă putem regăsi și unele butoane cu diferite funcții);

4. al doilea rând de meniuri (poate lipsi la unele pagini web).

La unele pagini web, în antet veți mai găsi și un instrument de căutare. Introduceți aici textul de căutat pentru a identifica mai ușor informația dorită în textul paginii.

Tot în antet, unele site-uri alocă un spațiu mai mare unui element grafic folosit pentru prezentare, numit **slider** sau **carusel de imagini** (*imaginea e*). În terminologia utilizată în web design, termenul „slider” este folosit pentru o prezentare de imagini sugestive introdusă în pagina web. Este vorba de un set de imagini care se schimbă automat sau nu, cu diferite efecte de trecere între ele, menite să atragă atenția, să încante și să convingă utilizatorul să parcurgă conținutul paginii. Există o mulțime de unelte (software) ajutătoare cu care puteți crea astfel de slidere pe care să le introduceți în paginile voastre web.

Unele site-uri păstrează fixă zona de antet pe toate paginile lor, altele o au doar pe prima pagină (pagina **Acasă**), iar pe restul paginilor o au într-o formă simplificată, care ocupă mai puțin loc, în general fără zona de imagine sau de **carusel de imagini** din antet.



Un avantaj clar al utilizării unui slider este faptul că puteți plasa într-o manieră vizuală informațiile pe care doriți să le scoateți în evidență pentru a fi recepționate simplu și rapid de către vizitatorii paginii.



Un slider poate rula imagini în mod automat fără vreo acțiune din partea utilizatorului, schimbându-le la un interval de timp predefinit, cu anumite efecte de trecere între ele. Sliderele pot și să răspundă la interacțiunea utilizatorului, cum ar fi clic sau glisare, pentru a vizualiza imaginile următoare sau anterioare. În plus, ele pot avea și butoane sau miniaturi pe care utilizatorii pot face clic pentru a vizualiza o anumită imagine din slider.

Acum aveți mult mai multe cunoștințe despre elementele paginilor web, cunoștințe care vă vor face să priviți paginile web cu alți ochi, să fiți analiști fini ai paginilor pe care le veți vizita.

IMPORTANT! Trebuie să aveți grijă cum vă protejați împotriva fraudelor informatice atunci când deschideți diferite pagini web. Securitatea cibernetică este vitală.

Mare atenție: trebuie să acordați o importanță deosebită informațiilor voastre personale! Nu vă lăsați păcăliți de acele pagini web care încearcă, prin diferite formulări destul de convingătoare (la o primă vedere), să obțină de la voi astfel de informații. Sub termenul general de **inginerie socială** înțelegem, în legătură cu securitatea informațiilor, manipularea psihologică a oamenilor pentru a-i face să întreprindă acțiuni prin care să își divulge, către persoane rău intenționate, diferite informații confidențiale (începând cu adrese de e-mail sau numere de telefon, adresa de domiciliu sau codul numeric personal și mergând până la parole sau informații legate de conturi bancare). În concluzie, este o formă de escrocherie prin care se urmărește colectarea de informații care ulterior pot fi folosite de către cei care le-au colectat, cauzând diferite și însemnate pagube celor care au căzut în capcană.



Rețineți

Reguli pentru o navigare mai sigură printre paginile de net:

- Nu accesați pagini web care par nesigure încă de la prima vedere.
- Verificați secțiunea de contact a site-urilor web (adresă, număr de telefon, e-mail).
- Verificați destinația reală a linkurilor prin trecerea cursorului mouse-ului peste acestea și vizualizarea adresei reale în partea stângă-jos a browserului.
- Nu apăsați pe linkurile din cadrul ferestrelor de tip *pop-up* (fereastră nouă de browser care afișează în general reclame).
- Verificați existența „https://” în partea de început a adresei web, înainte de a introduce informații personale.
- Nu instalați programe decât din proprie inițiativă, nu la propunerea altcuiva! Multe amenințări online vin sub forma cererilor de a da clic pe un anumit link sau de a deschide atașamentul unui mesaj e-mail. Sunt pagini web ce deschid ferestre *pop-up* care vă cer să rulați un scanner de securitate sau să instalați un player pentru a putea vizualiza diverse conținuturi care pot părea atrăgătoare (escrocii știu să vă stârnească interesul). Evitați să dați curs unor asemenea cereri.
- Dacă este necesar să instalați un program, încercați să îl descărcați direct de la sursă (de pe site-ul web al producătorului) și nu de pe site-uri web terțe.



Aplicați și investigați

1. Deschideți pagina **www.pbinfo.ro**. Identificați antetul paginii. Antetul este cel pe care îl aveți colorat cu albastru în partea de sus. Identificați în subsolul paginii butonul de contact. Subsola acestei pagini este colorată în gri. Identificați corpul paginii. Corpul paginii este reprezentat de tot ce este cuprins între antet și subsol. Găsiți în corpul paginii cele două câmpuri care se completează pentru autentificare. Care sunt acestea?
2. Alegeți o pagină web preferată și identificați-i elementele componente.



Exersați



1. Scrieți, în caiete, **A** (Adevărat) sau **F** (Fals), în funcție de valoarea de adevăr a propozițiilor:
 - a. O pagină web fără subsol nu poate funcționa. **A F**
 - b. Este obligatoriu ca în antetul unei pagini web să existe instrumentul de căutare de text. **A F**
 - c. Titlul unei pagini web anunță conținutul respectivei pagini. **A F**
2. Dați trei exemple de bune practici pentru o navigare mai sigură printre paginile web din Internet.

Portofoliu



Realizați o prezentare PowerPoint referitoare la momente din istoria paginilor web. Evidențiați schimbările de structură suferite în timp. Prezentarea trebuie să conțină imagini și sunete, eventual înregistrate de voi.

2 Aplicații specializate pentru crearea paginilor web – elemente de interfață și facilități



Descoperiți

Am văzut în lecția anterioară elementele de structură ale unei pagini web. Dar cum și cine poate crea o pagină web? Oare noi putem? Răspunsul este DA! Să vedem cum.



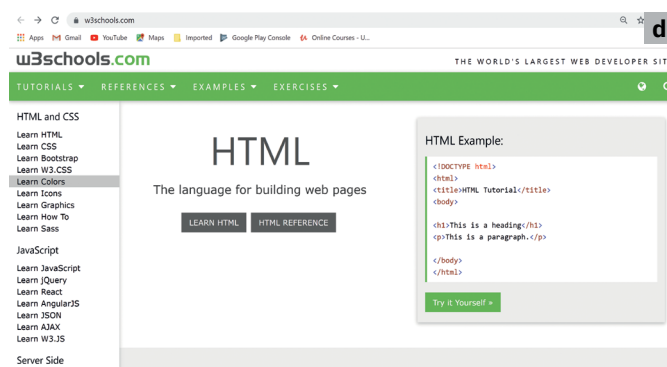
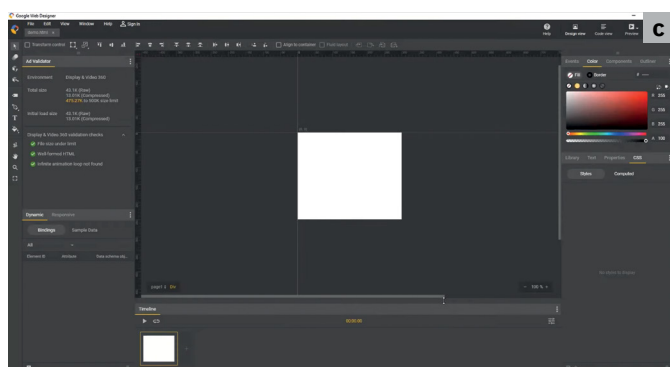
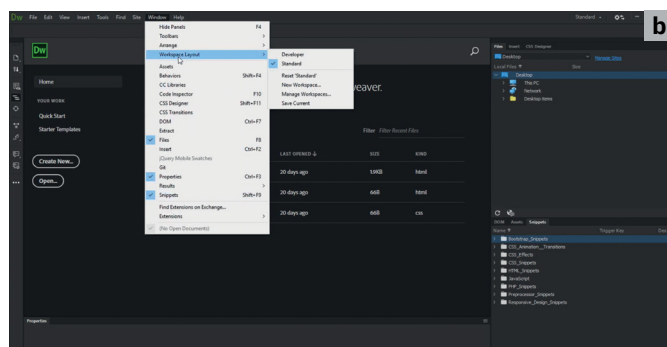
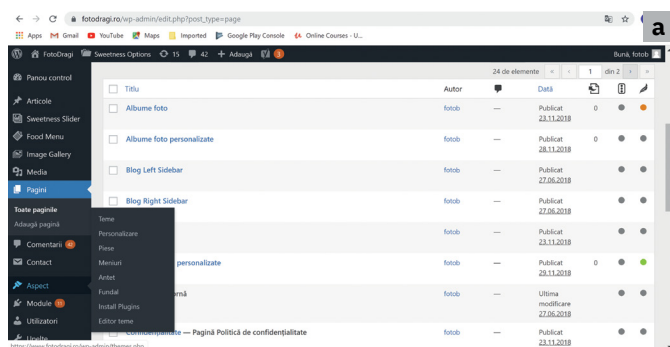
Rețineți

Există două posibilități total diferite și clar definite pentru crearea unei pagini web, dar și combinații ale acestor două posibilități, diferite variante intermediare:

1. prima posibilitate este prin programare, prin scriere de linii de cod (de exemplu, în limbaj HTML) pentru a obține pagina web dorită;
2. a doua metodă are în vedere folosirea de aplicații (programe) specializate în crearea și editarea de pagini web. În cele ce urmează, vom utiliza această a doua metodă.



Există o multitudine de programe pentru creare și editare de pagini web. Unele se instalează pe calculatoarele voastre, altele se accesează online, în browser. Dintre cele mai utilizate și apreciate în acest moment sunt: WordPress (*imaginea a*), Wix, Kompozer, Google Sites, Dreamweaver (*imaginea b*), Blogger, Google Web Designer (*imaginea c*), Site123, Squarespace, SiteBuilder, Jimdo și multe, multe altele. W3Schools (*imaginea d*) este un site foarte important pentru învățarea realizării paginilor web.

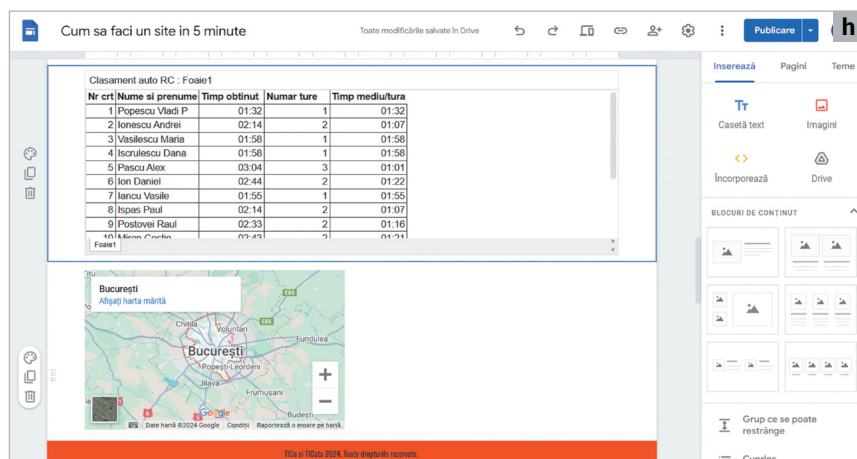
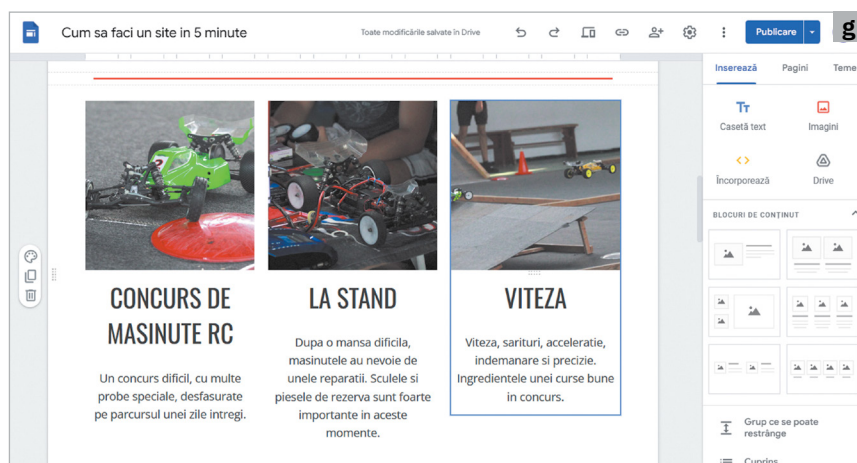
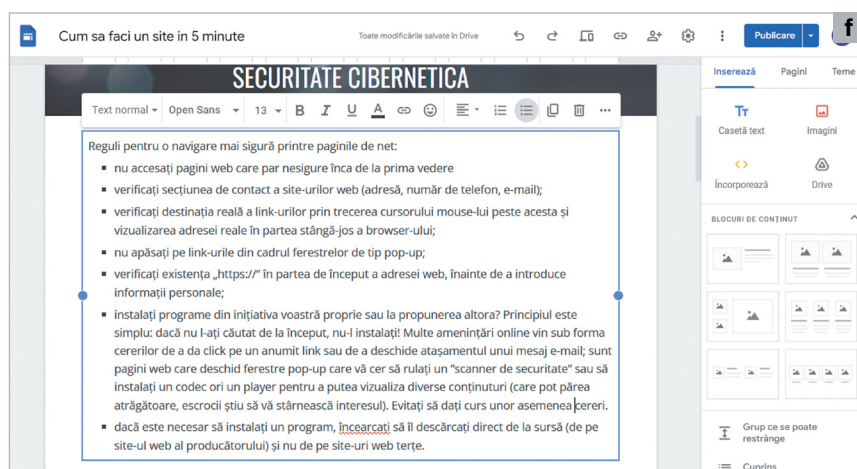
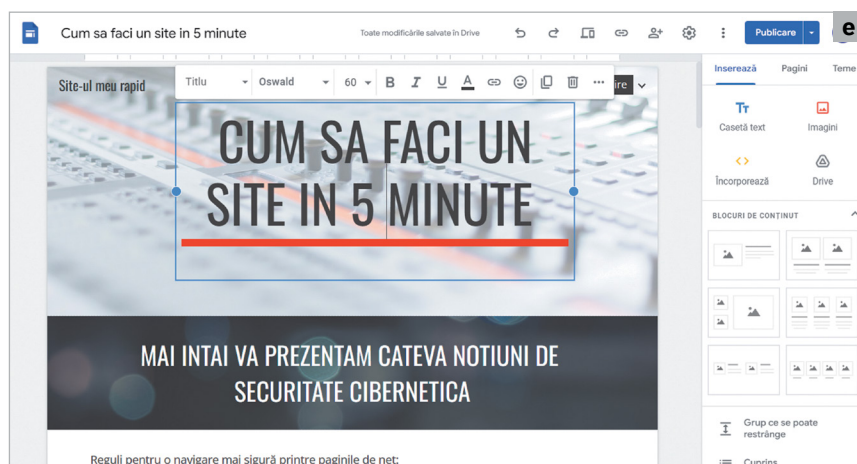


Analiza unei pagini web



Chiar și din Microsoft Word puteți crea o pagină web simplă: scrieți conținutul-text pe care îl doriți într-un document nou, inserați imaginile potrivite, aranjați totul după bunul-plac, iar la final, la salvare, alegeți „web Page” la tipul de fișier în care vreți să salvați și veți obține un fișier **html**, adică o pagină web. Doar că pentru a arăta ca o pagină web modernă, actuală și pentru a avea și funcționalități similare, e nevoie de mai multă muncă. Unele funcționalități vor rămâne însă imposibil de implementat din Word.

Dintre toate aceste aplicații, noi vom analiza mai amănunțit și vom folosi în lecțiile următoare programul **Google Sites**. Îl puteți deschide accesând linkul următor: <https://sites.google.com>. Atenție, conform specificațiilor producătorului, Google Sites funcționează bine cu unul dintre următoarele browsere de Internet: Chrome, Firefox sau Safari.



Deschideți mai întâi, în editorul Google Sites, o pagină web existentă și identificați elementele din care este formată. Accesați, de exemplu, pagina <https://sites.google.com/view/site-ulmeurapid>.

Observăm *numele site-ului* („Site-ul meu rapid”) și *antetul*. În antet putem identifica titlul paginii („CUM SA FACI UN SITE IN 5 MINUTE”) și imaginea de antet (*imaginea e*).

După antet începe *corpul paginii*, conținutul paginii, partea consistentă, plină de informații. Observăm o zonă de text, cu câteva paragrafe și cu titlul aferent (*imaginea f*).

Urmează un element grafic, o linie roșie. Este numit „element separator” și are rolul (după cum îi spune și numele) de a separa vizual două zone din corpul paginii.

După separator, observăm un element al corpului paginii care seamănă cu un tabel cu două rânduri și trei coloane. Pe primul rând vedem trei imagini (sunt fotografii de la un concurs de mașinuțe RC pilotate de colegi de-ai voștri), iar pe al doilea rând, câte un text pentru fiecare imagine, împreună cu titlul aferent (*imaginea g*).

Această parte a paginii web este o secțiune a sa. O pagină web poate avea mai multe secțiuni.

Urmează un element de altă natură: un tabel. Este un tabel făcut cu un alt program, fiind inserat ulterior aici, în pagina web. Avem posibilitatea (oferită de editorul Google Sites) să inserăm diferite elemente, cum ar fi: documente (realizate într-un editor de text), filmulețe de pe YouTube, prezentări, formulare, diagrame etc. În acest caz, a fost inserat în pagina web un tabel făcut într-un program specializat în realizarea de foi de calcul. Am învățat și noi să facem astfel de tabele la capitolul dedicat foilor de calcul tabelar.

După tabel, observăm un alt element care vouă vă este familiar din diferite site-uri vizitate sau aplicații utilizate: „harta”. Se poate insera un astfel de element pentru a indica o locație importantă pentru informațiile din corpul paginii (*imaginea h*).

La final regăsim *subsolul* paginii, pe fond roșu – în cazul paginii noastre, în armonie cu dunga roșie de sub numele din antet și cu elementul separator.

Elemente de interfață Google Sites

Să discutăm acum despre editorul pe care îl vom folosi pentru realizarea paginii noastre web și **elementele de interfață** ale acestuia.

Ați făcut deja cunoștință cu **Google Sites**, una dintre cele mai populare aplicații de editare a paginilor web. Să analizăm împreună principalele elemente de interfață ale editorului de pagini web Google Sites.



Odată ce accesăm linkul **https://sites.google.com**, ne facem un cont (dacă nu avem deja unul) și ne autentificăm; ni se va deschide un ecran pentru gestionarea proiectelor de site-uri similar cu cel din *imaginea i*.



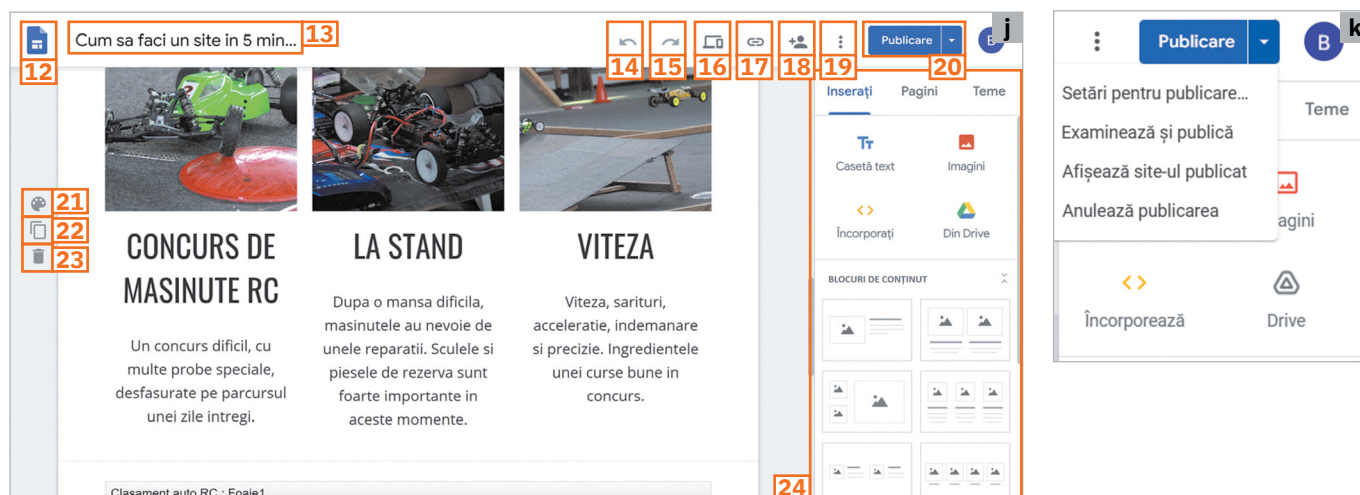
Iată ce conține:

1. sub denumirea **Site-uri recente** vom regăsi ultimele proiecte la care am lucrat (primul este cel mai recent);
2. **iconițele corespunzătoare proiectelor** (un clic pe iconiță deschide proiectul corespunzător);
3. **bara de căutare** – ne va fi utilă atunci când vom avea multe site-uri create; doar introducem câteva cuvinte din numele proiectului căutat și ne apar toate proiectele care au în nume ceea ce s-a introdus;
4. de la acest meniu putem alege site-urile **Deținute de mine** (adică site-urile proprii), **Nedeținute de mine** (adică site-urile create de alții, dar la care ni s-a dat acces, mai precis cele ce au fost create de altcineva și au fost partajate și cu noi pentru lucrul colaborativ) sau **Deținute de oricine** (adică ambele categorii de site-uri);
5. butonul pentru **Schimbarea modului de afișare a iconițelor** proiectelor (**Vizualizare grilă**, cum sunt afișate în ecranul nostru, sau **Vizualizare listă**);
6. butonul **Opțiuni de sortare**, de la care se pot ordona proiectele existente după diferite criterii (faceți un clic pe buton și veți descoperi care sunt opțiunile de sortare disponibile);
7. butonul **Deschideți selectorul de fișiere**, un instrument care ne ajută să navigăm printre folderurile și fișierele pe care le avem la dispoziție și să selectăm fișierul dorit;
8. butonul **Aplicații Google** este unul general al interfeței Google, de la care putem accesa orice aplicație Google (YouTube, Hărți, Gmail, Drive, Calendar etc.);
9. butonul **Contul Google**, de la care putem gestiona contul Google;
10. butonul de unde se accesează **Meniul principal** al drive-ului Google, de la care gestionăm inclusiv versiunile anterioare ale site-urilor lucrate;
11. butonul **+** este cel de la care putem crea un nou proiect, un nou site.

Observați că avem două proiecte (două site-uri): „Cum sa faci un site in 5 minute” și „Primul meu site”. Să deschidem proiectul „Cum sa faci un site in 5 minute”.

Elementele de interfață, care ne sunt folositoare în realizarea unui proiect, sunt cele din *imaginea j*.

12. **Pagina de pornire site-uri** este butonul de întoarcere în ecranul anterior, de unde ne putem alege un proiect sau putem crea unul nou;
13. titlul proiectului;
14. butonul **Anulează ultima acțiune**, clasicul buton **Undo** din orice editor;
15. butonul **Refă ultima acțiune** este clasicul buton **Redo** din orice editor;
16. **Vizualizare** – buton care ne afișează site-ul în mod *fullscreen* (pe tot ecranul), așa cum arată el pe Internet în acel moment al editării;
17. **Copiază linkul către site-ul publicat** – copiază (în memorie) linkul către site, iar cu ajutorul opțiunii **Lipește** putem să transmitem linkul site-ului către oricine dorim;
18. **Trimite altor persoane** este un buton prin care putem invita prieteni sau colegi să lucreze împreună cu noi la acest proiect – le dăm acces la proiectul nostru (lucru colaborativ);
19. **Mai multe** – diverse alte meniuri pe care le avem la dispoziție, de la **Istoricul versiunilor** sau **Fă o copie** până la **Ajutor**.



20. butonul **Publicare**, necesar pentru publicarea proiectului, cu trei opțiuni pe care le putem vedea în detaliu în *imaginea k*;

21. **Fundalul secțiunii** – buton de la care se alege fundalul secțiunii de site selectate în acel moment;

22. **Duplică secțiunea** – buton pentru duplicarea secțiunii selectate, util când vrem să punem într-un site mai multe secțiuni asemănătoare;

23. **Șterge secțiunea** – buton pentru ștergerea secțiunii de site selectate;

24. în partea dreaptă a ecranului avem **Bara de unelte** (un meniu de editare cu trei file/taburi), în care se regăsește tot ce este necesar pentru realizarea site-ului;

25. zona de selectare a unuia dintre cele trei taburi: **Inserați** (*imaginile l, m*), **Pagini** (*imaginea n*) și **Teme** (*imaginea o*);

26. zona de unelte de inserat casete de text, imagini, elemente de pe Internet sau elemente din Drive-ul Google precum documente text, filmulețe etc. (toate acestea se pot insera rapid și ușor în site);

27. zona **Blocuri de conținut**, pentru diferite șabloane de așezare a imaginilor și a textelor.

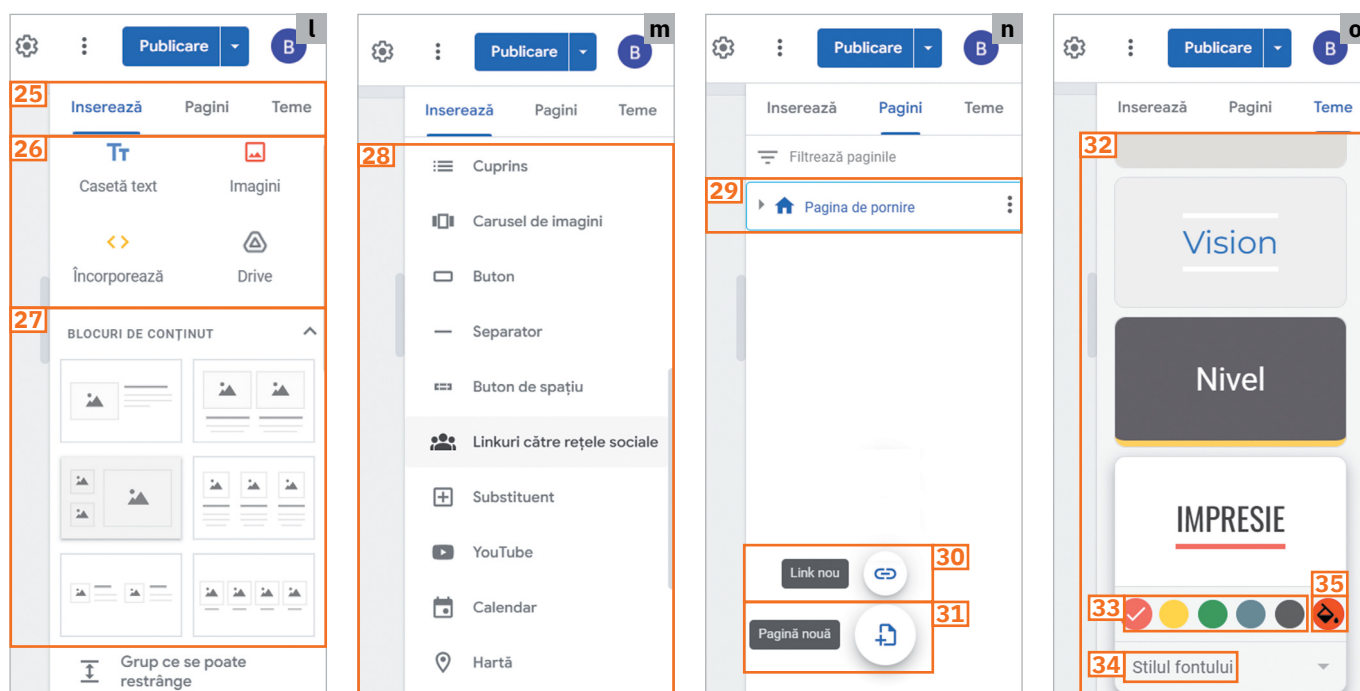
➕ În tabul **Inserează** regăsim mai multe opțiuni de adăugare a elementelor de conținut:

28. zona din bara de unelte din care putem insera diferite elemente în pagina noastră: carusel de imagini (mai multe imagini care se succed), butoane, separatoare, filme de pe YouTube, calendare, hărți, documente, prezentări, **foi de calcul**, formulare, diagrame etc.

Tabul **Pagini** conține:

29. zona în care putem vedea și gestiona toate paginile site-ului nostru (momentan avem doar o pagină, **Pagina de pornire** sau **Acasă**);

30. buton de **adăugare a unei noi legături (Link nou)**; aceasta poate fi editată în orice moment pentru a realiza o legătură către o altă pagină;



31. buton de adăugare **Pagină nouă** în site-ul nostru (într-un site construit cu Google Sites se pot adăuga un număr nelimitat de pagini).

Tabul **Teme** conține:

32. mai multe teme predefinite dintre care se poate alege. O **temă** este un șablon pentru cum va arăta site-ul din punct de vedere coloristic, al stilului fonturilor sau stilul în ansamblu. Cu un singur clic putem schimba radical felul în care arată o pagină de site deja creată;

33. culori pentru a putea personaliza fiecare temă;

34. stiluri de fonturi din care putem alege ceea ce ni se pare mai potrivit pentru textul nostru;

35. paletă de culori integrată, necesară atunci când dorim o altă culoare decât cele oferite implicit în tema aleasă. Putem chiar specifica o anumită culoare prin codul ei Hexa, o succesiune de trei perechi de numere în sistem hexazecimal precedate de # (de exemplu, portocaliu are codul #FFA500).



Aplicați și investigați

1. Căutați pe Internet informații despre trei aplicații utilizate pentru editarea paginilor web. Analizați comparativ cele trei aplicații, evidențiind câte două puncte tari și câte două puncte slabe ale fiecăreia.
2. Luați o pagină web existentă, creată pentru exercițiu de profesorul vostru, și modificați-i conținutul. Încercați să modificați și să adăugați texte, să înlocuiți imagini, să adăugați diferite elemente etc. Pentru fiecare modificare, vizualizați rezultatul în programul de navigare pe Internet, pentru a-l vedea în forma în care îl vor vizualiza vizitatorii paginii respective.



Exersați

1. Dați trei exemple de aplicații specializate în crearea paginilor web.
2. Scrieți, în caiete, **A** (Adevărat) sau **F** (Fals), în funcție de valoarea de adevăr a propozițiilor:

a. Se poate crea o pagină web folosind programul Microsoft Word.	A F
b. Se poate încărca o imagine ca fundal al unei pagini web.	A F
c. Dreamweaver este un program de interpretat vise și de creat horoscopul zilnic.	A F
3. Răspundeți la următoarele întrebări:
 - a. Ce este o temă în Google Sites?
 - b. Ce se poate face dacă, în timpul editării unei pagini web în Google Sites, am șters din greșală un element din pagină care nu trebuia șters?
 - c. Cum putem vedea, la un moment dat, cum arată site-ul la care lucrăm în Google Sites? Cum îl putem vedea la final, înainte de a-l publica pe Internet?
 - d. Care sunt cele două moduri de afișare a iconițelor proiectelor în Google Sites?
 - e. Ce este pagina **Acasă**?
4. Scrieți în caiete răspunsul corect pentru următorul enunț. O temă Google Sites permite:
 - a. aplicarea unei scheme de culori predefinite site-ului vostru;
 - b. adăugarea unor modele 3D site-ului vostru;
 - c. crearea unui site web pornind de la un site web existent.



Portofoliu



Unul dintre avantajele principale ale lui Google Sites este că oferă suport pentru lucrul colaborativ. Scrieți un mic eseu de maximum 200 de cuvinte în care să expuneți părerea voastră legată de cum anume poate profita o instituție școlară de această caracteristică a lui Google Sites și cum anume se poate transforma aceasta într-un veritabil instrument pentru eLearning, în scopul susținerii procesului educațional.



3 Crearea unei pagini web. Publicarea unui site de o pagină



Descoperiți

A. Acțiuni pregătitoare

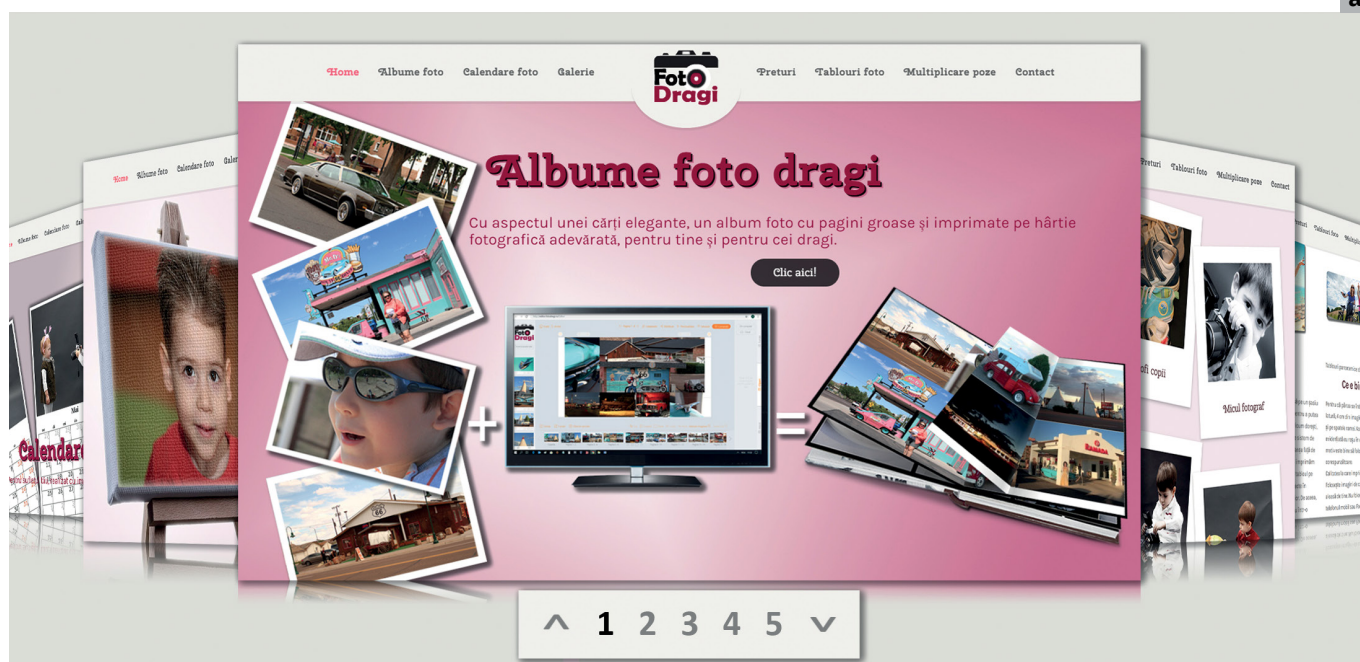
Acum știți deja multe lucruri despre paginile web. Este timpul să facem prima noastră pagină.

Mai întâi va trebui să ne gândim la câteva aspecte, pregătind „terenul”:

- care va fi tema (subiectul) paginii noastre web;
- ce nume îi vom da;
- care va fi structura paginii (ne gândim ce elemente vrem să punem în pagină și cum vrem să le aranjăm);
- ce conținut va avea fiecare element din pagină (căutăm sau realizăm imaginile, scriem textele, pregătim tabelele etc.).

Ne propunem să facem o pagină web personală pentru un oracol al prietenilor și al colegilor noștri. Știți ce este un oracol? Conform definiției din *Dicționarul de argou al limbii române* (George Volceanov, 2007), acesta reprezintă o pagină cu un chestionar privind gusturile în materie de muzică, film și altele, dar și cu o predicție legată de viitor, la care răspund colegii de clasă, prietenii etc.

Să creăm, așadar, o pagină dedicată unui astfel de oracol. Îi vom da numele „Oracol + numele propriu”. Pentru exemplificare, să considerăm „Oracol Maria P”.



Drept conținut al paginii ne propunem să avem:

- Carusel de imagini (în antet) (*imaginea a*) – 3-4 imagini reprezentative pentru noi;
- Cuvânt de introducere (poate fi chiar și numai un singur cuvânt);
- Numele;
- Data nașterii/„Când ai aterizat pe Terra?”;
- O imagine cu ceva drag (obiect, loc, ființă);
- Muzica preferată;
- Filmele preferate;
- Florile preferate;
- Sportul preferat;
- Materiile preferate;
- Secțiunea întrebărilor serioase (inserare paragraf):
 - Ți-ar fi plăcut să trăiești la curtea regală a Franței pe vremea lui Ludovic al XIV-lea? De ce?
 - Ce ai face dacă ai fi multimilionar?

Desigur, putem completa această structură cu orice alte texte, imagini, tabele sau grafice dorim. Putem da frâu liber imaginației.

Pentru variația elementelor pe care le putem introduce într-o pagină web, ne propunem să inserăm și un tabel făcut în Microsoft Excel, de exemplu, un tabel cu punctajele, mediile și clasamentul de la un concurs școlar.

În continuare, pregătim și răspunsurile la toate cele de mai sus. Apoi trecem la pasul următor.

Dacă avem un cont Gmail sau Google, avem deja un cont inclusiv pentru Google Sites. Ne conectăm la contul nostru, alegem un nume de site și o structură de conținut și, în felul acesta, terminăm prima parte.

B. Crearea unui site web cu o pagină

Cum putem crea un site în Google Sites? Parcurgeți pașii de mai jos și executați și voi, la rândul vostru, pe calculator:

1. Accesați <https://sites.google.com> (imaginea b).

2. Creați un site nou.

În **Pagina de pornire Site-uri** apăsați butonul + din dreapta-jos (**Crează un site nou**).

3. Alegeți o temă pentru pagina voastră și configurați antetul.

Fiecare temă vine cu un fundal presetat, cu o schemă de culori și cu un font selectat. Puteți să modificați oricare dintre aceste elemente, sau chiar tema în sine, în orice moment,

chiar și după ce au fost introduse informații pe pagina web. Dacă aveți nevoie să anulați sau să refaceți unele schimbări, puteți folosi butoanele **Anulează ultima acțiune (Undo)** și, respectiv, **Refă ultima acțiune (Redo)**.

1. Modificarea imaginii de fundal a antetului (imaginea c):

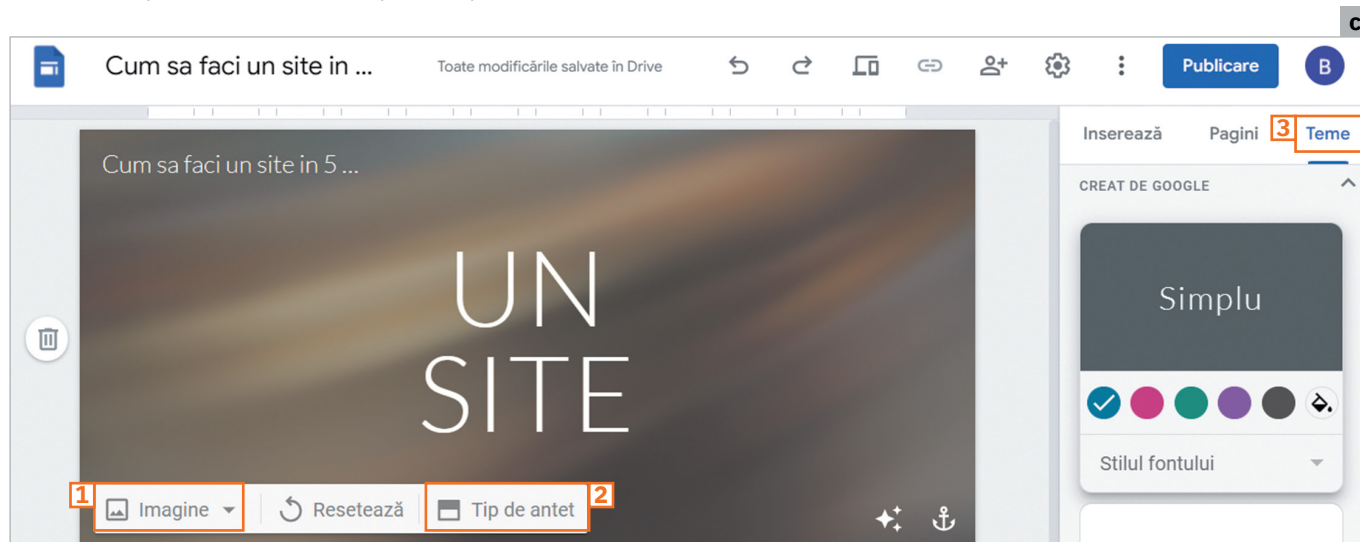
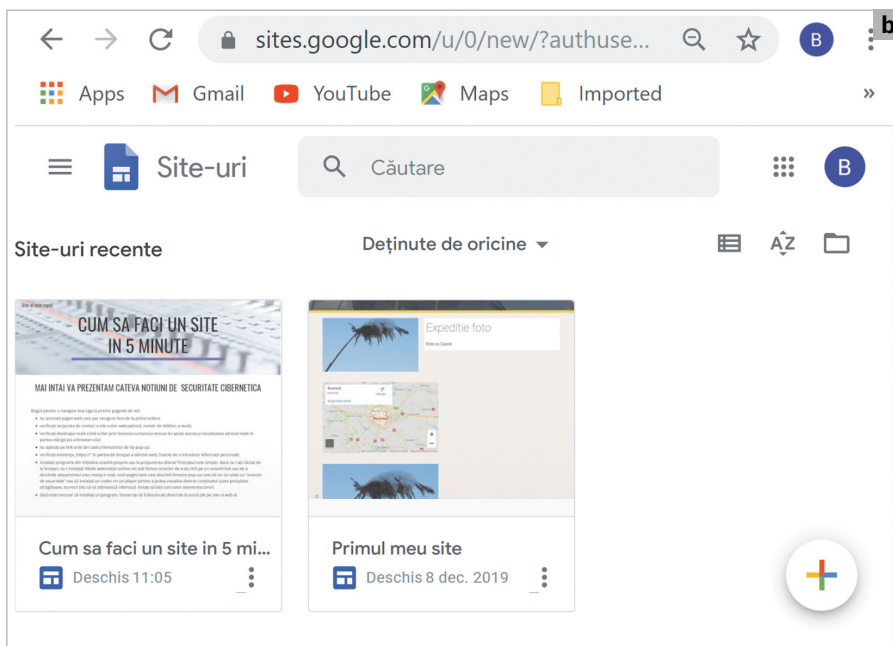
- Mergeți pe zona de antet și selectați **Imagine**.
- Alegeți una dintre opțiunile:
 - **Încarcă** – pentru a încărca o imagine din calculatorul propriu;
 - **Selectare** – pentru a alege o imagine din galeria de imagini;
 apoi apăsați pe butonul **Inserează**.
- Pentru a reveni la imaginea de fundal originală, apăsați butonul **Resetează** (opțional).

2. Modificarea tipului de antet:

- Mergeți pe zona de antet și apăsați butonul **Tip de antet**.
- Alegeți o opțiune:
 - Copertă;
 - Banner;
 - Banner mare;
 - Numai titlu.

3. Schimbarea temei site-ului și a fontului (stilului de scris) folosit:

- Din colțul din dreapta-sus, alegeți **Teme**.
- Selectați o temă care vi se pare mai potrivită și alegeți o culoare (dacă doriți să o modificați pe cea implicită a temei).
- Apăsați pe **Stilul fontului** și alegeți un stil.



4. Dați un nume site-ului și puneți un titlu paginii.

Atenție, aveți trei părți diferite ale site-ului care trebuie să primească un nume (*imaginea d*):

- 1. Numele de fișier al site-ului**, acel nume cu care îl veți regăsi între fișierele voastre din drive-ul Google. Acest nume este vizibil numai pentru voi, nu și pe Internet.
- 2. Numele site-ului**. Acest nume apare în antet și în bara de titlu a browserului de Internet după ce faceți site-ul public. Trebuie să aveți două sau mai multe pagini în site-ul vostru web pentru ca acest nume să fie afișat.
- 3. Titlul paginii**. Fiecare pagină din site-ul vostru trebuie să aibă un nume care apare în partea de sus a paginii. Titlul paginii apare și în meniul de navigare al site-ului.



Numele care apare în bara de titlu a browserului (tagul paginii web) (*imaginea e*) este de ajutor atât pentru utilizator, cât și pentru motoarele de căutare care vor citi și prelucra această informație. Este important ca acesta să fie sugestiv ales și concis formulat pentru a fi clar ce informații se găsesc în pagina respectivă.

Imaginați-vă acum că ați deschis în browserul vostru de Internet mai multe pagini, de exemplu, zece. Cum puteți trece de la o pagină la alta fără să le luați la rând sau pe încercate? Aceasta este o altă situație în care se dovedește utilitatea tagului, pentru că numele site-ului este afișat în eticheta corespunzătoare tabului paginii respective. Când doriți să reveniți la o anumită pagină, o puteți regăsi și accesa cu ușurință datorită numelor site-urilor, care sunt afișate de browserul de Internet, după cum puteți vedea în imaginea de mai jos („Manuale Digitale”, „YouTube”, „Brainly.ro – Rețeaua...”, „Google”).



Rețineți

Câteva recomandări („cele mai bune practici”) pentru a crea taguri bune, sugestive și relevante, atât pentru utilizatori, cât și pentru motoarele de căutare:

- să fie cât mai scurte (ideal ar fi să aibă maximum 55 de caractere);
- nu folosiți numai litere mari;
- nu repetați cuvinte în cadrul unui tag;
- să fie cu adevărat relevante pentru conținutul paginii;
- scrieți nume convingătoare, care v-ar face și pe voi să doriți să faceți clic pe el.

5. Creați paginile și subpaginile dorite (meniul rezultat este disponibil în dreapta-sus); momentan, creați o singură pagină, numită **Pagina de pornire** sau, cum este cunoscută în lumea programatorilor, pagina **Acasă**.

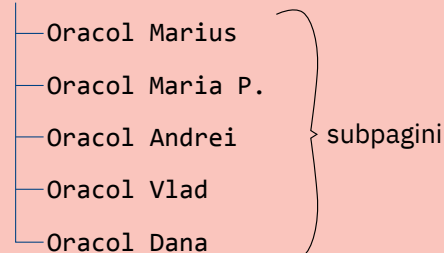
Rețineți

O **subpagină web** este o „pagină copil”, o pagină web care este legată de o altă pagină web în structura de pagini a site-ului (*imaginea f*). Putem avea și subpagini ale unei subpagini.

Odată creată o pagină, se pot alege aspecte pentru secțiunile sale mergând la tabul **Inserează > Aspecte**. Selectarea unui aspect va duce la inserarea unei noi secțiuni în pagina voastră, care are aranjarea imaginilor și textelor corespunzătoare aspectului selectat.

Puteți vedea în imaginea alăturată un exemplu de structură a unei pagini web cu cinci subpagini.

Pagina principală (Acasă)





6. Încărcați conținut în pagină: carusel de imagini, texte, poze, hartă (dacă este cazul), casete text etc.

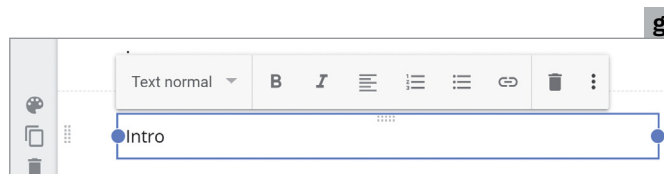
Din colțul din dreapta-sus alegeți **Inserează**. Din bara de unelte care se desfășoară pe verticală puteți insera diferite elemente în pagina web: text (**Casetă text**), imagini (**Imagini**), pagini web existente (**Încorporează**), carusel de imagini (**Carusel de imagini**), butoane (**Buton**), separatoare (**Separator**), filme de pe YouTube (**YouTube**), calendare (**Calendar**), hărți (**Hartă**), documente (**Documente**), prezentări (**Prezentări**), foi de calcul (**Foi de calcul**), formulare (**Formulare**), diagrame (**Diagrame**).

Editarea textului

Pentru a edita un text, faceți clic pe căsuța de text și introduceți de la tastatură textul dorit. Puteți, de asemenea, să selectați dintre opțiunile de editare disponibile din caseta de unelte de deasupra (*imaginea g*).

În funcție de stilul ales pentru text (text normal/titlu/antet de rubrică – în engleză, *heading* etc.), veți avea la dispoziție opțiuni diferite de editare/formatare. De exemplu, în *imaginea g* puteți vedea care sunt opțiunile pentru un text normal.

Descrierea opțiunilor de editare și formatare în funcție de stilul de text este detaliată în tabelul de mai jos.



Pentru titluri și antete de rubrici	
Titlu ▼	Selectați Titlu sau un antet de rubrică presetat. Dacă selectați Text normal , veți avea la dispoziție opțiuni suplimentare de formatare, prezentate în partea a doua a tabelului.
	Schimbați alinierea textului.
	Inserați un hiperlink în textul selectat.
	Ștergeți o căsuță de text.
Pentru text normal	
B <i>I</i>	Selectați textul dorit și aplicați-i efectul de îngroșat sau <i>aplecat</i> .
	Adăugați liste numerotate sau cu buline.
	Mai multe opțiuni...
	Selectați textul și tăiați-l cu o linie.
	Selectați un text și eliminați-i toate opțiunile de formatare aplicate.

Editarea unei secțiuni

Alegeți secțiunea pe care doriți să o editați.

- Schimbați fundalul pentru fiecare secțiune – clic pe **Fundalul secțiunii** și alegeți un stil (Normal, Subliniere 1, Subliniere 2). Încercați, pe rând, toate opțiunile disponibile și alegeți varianta care se potrivește cel mai bine cu proiectul vostru. Puteți să selectați și **Imagine** pentru a adăuga pe fundalul secțiunii o imagine.
- Mutați o secțiune – când introduceți conținut nou, acesta este adăugat la o secțiune nouă sau la una existentă. Pentru a muta o secțiune în sus sau în jos pe pagină, faceți clic pe **Mutare secțiune** și trageți secțiunea unde doriți.
- Duplicarea unei secțiuni – faceți clic pe buton și secțiunea respectivă se va duplica, împreună cu tot conținutul.
- Dacă doriți să ștergeți o secțiune (valabil pentru orice tip de secțiune), e suficient să mergeți la acea secțiune și să faceți clic pe butonul **Șterge secțiunea** (*imaginea h*).

O secțiune specială este subsolul paginii. Pentru a adăuga o secțiune de subsol (în engleză, *footer*), mergeți în josul paginii și faceți clic pe **Adaugă subsol**. Secțiunea de subsol va apărea pe toate paginile site-ului vostru.

Mergeți la ea și faceți clic pe **Editează subsolul** pentru a o modifica. Pentru a ascunde secțiunea de subsol pe o anumită pagină, deplasați-vă la ea și faceți clic pe **Ascunde subsolul pe această pagină** .



▶ Editarea imaginilor

• Redimensionarea unei imagini:

Selectați imaginea, apoi trageți de laturile sau de colțurile imaginii. Țineți apăsată tasta **Shift** în timp ce trageți, pentru a păstra raportul laturilor imaginii (pentru a nu o deforma).

• Mutarea unei imagini:

Trageți imaginea într-o altă parte a secțiunii sau într-o nouă secțiune. Puteți plasa imaginea oriunde vedeți o zonă evidențiată.

• Decuparea unei imagini:

- Clic pe imagine, apoi selectați **Decupează** ;
- Trageți cursorul de pe linia orizontală apărută (punctul) pentru a o mări sau a o micșora;
- Puteți mișca imaginea pentru a schimba zona vizibilă;
- Apăsați pe **Editarea s-a terminat** ;
- Pentru a anula decuparea, selectați imaginea și faceți clic pe **Anulează decuparea**  (opțional).

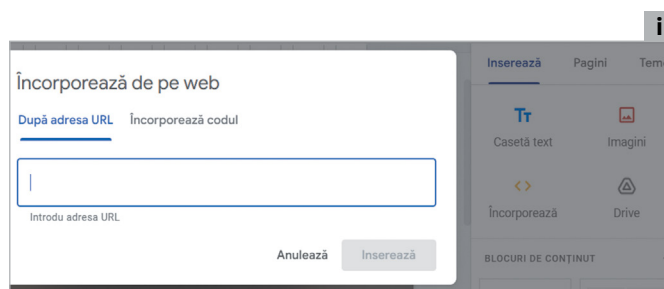
• Înlocuirea unei imagini:

- Selectați imaginea;
- Clic pe **Mai multe opțiuni de editare**  > **Înlocuiește imaginea** > **Încarcă** sau **Selectare** și alegeți un nou fișier imagine.

Adăugarea unui conținut dintr-un alt site web

Puteți încorpora conținut dintr-un alt site web (*imaginea i*). În funcție de site-ul pe care îl adăugați, puteți personaliza modul în care apare site-ul respectiv pe site-ul vostru. Unele site-uri conțin opțiunea de previzualizare.

- Deschideți proiectul site-ului vostru și din partea dreaptă alegeți **Inserează** > **Încorporează**.
- În caseta care se deschide, introduceți adresa web a site-ului sau a paginii dorite, apoi apăsați **Inserează**.



7. Verificați ce a rezultat (butonul Vizualizare).

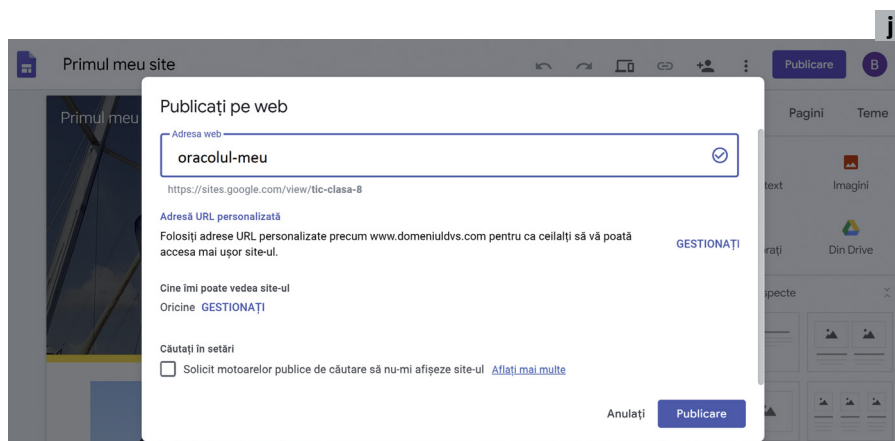
Dacă vizualizați un site pe un dispozitiv mobil, conținutul se adaptează automat pentru a se potrivi ecranului. Puteți previzualiza cum apare site-ul vostru pe diferite dispozitive (telefon, dispozitiv de tip tabletă sau ecran mare de calculator).

- Clic pe **Vizualizare** .
- Pentru a vedea cum arată site-ul vostru pe diferite dispozitive, alegeți una dintre opțiunile:
 - Telefon;
 - Dispozitiv de tip tabletă;
 - Ecran mare.
- Clic pe **Ieși din modul previzualizare**  pentru a închide acest ecran.

8. Să îl facem public, să îl vadă toată lumea (butonul Publicare). La acest pas trebuie să introducem numele de domeniu dorit. Google ne oferă posibilitatea de a ne găzdui site-ul chiar pe serverele Google. Dacă vom pune numele site-ului „Oracolul-meu”, adresa site-ului nostru va fi **<https://sites.google.com/view/oracolul-meu>**. Puteți comunica această adresă tuturor prietenilor, rudelor, cunoscuților, pe scurt, faceți site-ul public (*imaginea j*)!

Notă: Vizitați adresa URL a site-ului vostru pentru a verifica dacă site-ul a fost publicat corect. După ce publicați un site, puteți promova site-ul publicat făcând clic pe **Copiază linkul către site-ul publicat**  > **Copiază linkul** și apoi lipiți linkul în mesajele de text sau în e-mailurile pe care le transmiteți.

Și, cu această ultimă acțiune, am terminat. Primul nostru site web care conține o pagină este gata și poate fi vizitat online!





Rețineți

Când creați un site nou, acesta este adăugat în Google Drive, la fel ca orice alt fișier. Programul salvează automat orice modificare pe care voi o faceți, dar nu este vizibilă pe Internet până nu apăsați butonul **Publicare**.



Știați că...?

Puteți vedea cum sunt realizate diverse pagini web făcând clic dreapta pe pagina dorită și apăsând **View source page**. Veți vedea codul sursă, adică succesiunea de instrucțiuni HTML pe care calculatorul le interpretează pentru a afișa pagina web așa cum o vedem noi pe ecran.



Aplicați și investigați

1. Operații cu un tabel

Inserați în pagina voastră web un tabel creat în Excel și salvat în Google Drive cu filmele preferate și duratele lor, în minute, numit „Filme.xlsx”. Pentru aceasta:

- deschideți pagina voastră și poziționați-vă în locul în care doriți să inserați tabelul;
- mergeți în dreapta, la bara de unelte, și alegeți tabul **Inserează**;
- faceți clic pe **Foi de calcul**;
- din lista care se deschide, faceți clic pe fișierul „Filme.xlsx”;
- apoi clic pe butonul **Inserare**, din partea de jos.

Tabelul creat în Excel este acum inserat în pagina voastră web. Dacă doriți să îi schimbați poziția, doar îl trageți cu mouse-ul în poziția dorită din pagină. Dacă doriți să modificați numele unui film (să editați tabelul), faceți clic pe tabel, apoi clic pe butonul **Deschide într-o filă nouă** dintre cele trei butoane apărute în stânga-sus. Se deschide, pentru editare, tabelul vostru și puteți face modificarea dorită. După ce ați realizat modificarea și ați închis tabelul, doar apăsați butonul **Reîncarcă această pagină** din browserul în care aveți deschis Google Sites, și modificările făcute în tabel apar imediat.

2. Operații cu o legătură către o altă pagină web

Să presupunem că pe pagina voastră aveți, la rubrica „Muzica preferată”, muzică reggae. Pentru a-i ajuta pe cei care nu cunosc acest gen muzical, vreți să plasați o hiperlegătură către pagina destinată acestui subiect de pe Wikipedia. Astfel:

- în textul vostru, selectați cuvântul „reggae”;
- din meniul care apare deasupra secțiunii de text, faceți clic pe butonul **Inserează un link** ();
- în fereastra care se deschide, introduceți adresa paginii web din Wikipedia (<https://ro.wikipedia.org/wiki/Reggae>);
- apăsați butonul **Aplică**.

În acest moment, vizitatorii paginii voastre, când cîtesc acea rubrică, vor putea face clic pe cuvântul marcat (*reggae*) și li se va deschide pagina din Wikipedia destinată muzicii reggae.

Pentru a edita sau pentru a șterge o legătură, doar faceți clic pe textul pe care este aplicată legătura și apăsați butonul **Editează linkul** (pentru modificare) sau **Eliminare** (pentru ștergere).



Exersați

1. Un „sticker” este, prin definiție, un „abțibild” autocolant. Prin analogie, un sticker digital este o imagine de dimensiuni mici care poate fi „lipită” peste alte conținuturi digitale (imagini, texte etc.). Faceți cinci stickere și încărcați-le în pagina voastră web. Pentru aceasta puteți utiliza un editor grafic învățat. Există și aplicații online sau aplicații pentru mobile în care le puteți crea; doar căutați „sticker maker”.
2. Creați o pagină web (într-un proiect nou) în care să aveți: trei paragrafe de text, patru imagini, un document făcut în Word, un tabel făcut în Excel. Paragrafele de text să fie formate astfel: font Arial de 14, scris normal, unul cu textul aliniat la stânga, unul aliniat la dreapta și unul centrat, iar titlurile să fie îngroșate. Dintre cele trei paragrafe de text, unul trebuie să se repete de două ori. Cum faceți acest lucru fără a munci să îl scrieți și să îl formatați la fel de două ori?
3. În pagina web creată la exercițiul 2, formatați textul din secțiunile de text astfel încât acesta să fie centrat și îngroșat.
4. Cum se poate face ca fundalul unei secțiuni de text să fie un desen făcut în Paint?
5. Scrieți, în caiete, **A** (Adevărat) sau **F** (Fals), în funcție de valoarea de adevăr a propozițiilor:
 - a. Într-o pagină web făcută de noi putem încorpora conținut dintr-o altă pagină web. **A F**
 - b. O subpagină nu poate avea și ea o subpagină. **A F**
 - c. O secțiune de text nu poate avea un fundal colorat într-o singură culoare. **A F**

Portofoliu



Realizați o pagină web având ca temă sportul vostru preferat și echipa îndrăgită sau sportivul favorit (după caz). Setati culorile din tema site-ului în concordanță cu culorile echipei îndrăgite. La una dintre secțiunile de text, fundalul trebuie să fie cu o imagine reprezentativă sportului respectiv.

4 Construirea unui site Internet cu mai multe pagini. Lucrul colaborativ



Descoperiți

Mai multe pagini – un site. Site-ul clasei

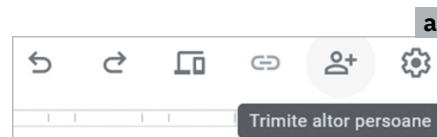
Scenariul 1: Profesoara/profesorul de la clasă creează un nou proiect de site, cu nume, pagină de pornire și subpaginile aferente: câte o subpagină pentru fiecare elev. Apoi, fiecare elev va crea conținutul pentru pagina web alcătuită, după propria dorință, dar în conformitate cu șablonul stabilit.

Scenariul 2: Profesoara/profesorul de la clasă creează un nou proiect de site doar cu nume și pagină de pornire. Fiecare elev se va conecta la acest proiect, își va crea propria pagină (după un șablon de nume anterior stabilit în comun, de tipul „Oracol_num_prenume”), apoi va încărca pagina cu informațiile dorite, respectând însă șablonul stabilit.

Pe oricare dintre aceste scenarii se lucrează, profesorul trebuie să dea drepturi de editare fiecărui elev din clasă (trimite „invitații”), iar voi vă conectați la conturile voastre Google, accesați **sites.google.com**, deschideți proiectul nou creat, pe care îl regăsiți în lista de proiecte, ca și cum ar fi fost creat de voi. Mergeți la butonul **Pagini** (din colțul din dreapta-sus), deschideți pagina voastră și începeți să o încărcați cu conținut (respectând șablonul stabilit).

Cum se realizează acest lucru? Cum putem face astfel încât să lucrăm împreună, ca o echipă, la același proiect de site web? Iată pașii necesari:

- cineva (profesorul/profesoara, dar poate fi și oricare dintre voi – îl vom numi „lider de grup”) creează proiectul; automat se creează și prima pagină, pagina de pornire; apoi liderul de grup face acțiunile de mai jos;
- clic pe **Trimite altor persoane** (imaginea a);
- se introduce adresa de e-mail a persoanei invitate și se apasă butonul **Trimite**;
- se trimite în același mod invitații de acces tuturor persoanelor implicate în proiect;
- la final, se apasă butonul **Terminat**



Fiecare persoană care a fost invitată să contribuie la proiect va primi pe e-mail invitația și un link de acces. Clic pe link și lucrul poate să înceapă.

Odată ce avem un site cu mai multe pagini, avem și o serie de noi operații de făcut și opțiuni disponibile. Să le trecem în revistă pe cele mai importante dintre ele.

Adăugați, reordonați și ascundeți paginile

Adăugați pagini pentru mai mult conținut, pentru un site mai bogat. Păstrați informațiile corelate laolaltă prin utilizarea subpaginilor. Subpaginile apar ca un „subtopic” al unei alte pagini, cea principală (imaginea b).

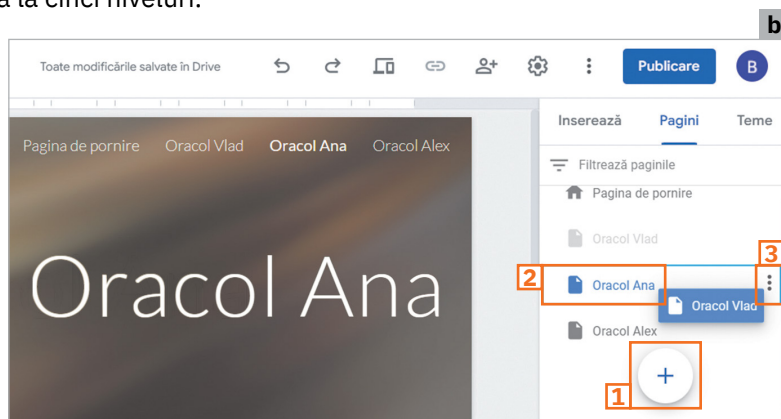
1. Adăugarea de pagini:
 - În colțul din dreapta-sus, faceți clic pe **Pagini** > **Adaugă** .
 - Pentru a adăuga o pagină, faceți clic pe **Pagină nouă** . Dați un nume paginii și apoi apăsați **Terminat**.
2. Reordonarea sau subordonarea paginilor:
 - Clic pe **Pagini**;
 - Trageți o pagină mai sus sau mai jos în listă pentru a-i schimba ordinea între celelalte;
 - Trageți o pagină peste o altă pagină pentru a o subordona. Pagina trasă deasupra va deveni subpagină a celei peste care a fost trasă;
 - Pentru a anula subordonarea unei pagini, trageți-o în partea de jos a listei (opțional).

Notă: Putem avea subpagini la subpagini până la cinci niveluri.

3. Diferite opțiuni pentru pagini:

De sub **Pagini**, selectați o pagină, în dreptul ei faceți clic pe **Mai multe** și alegeți una dintre opțiunile:

- **Setează ca pagină principală**;
- **Duplică pagina**;
- **Adaugă o subpagină**;
- **Ascunde din bara de navigare**. Nu veți putea ascunde o pagină care este setată ca pagină principală;
- **Șterge**. Nu veți putea șterge pagina setată ca pagină principală a site-ului vostru.



Configurați navigarea pe site

Dacă site-ul vostru are mai multe pagini, vizitatorii site-ului vor putea utiliza meniul de navigare pentru a trece de la o pagină la alta. În mod implicit, meniul de navigare se află în partea de sus a site-ului (în colțul din dreapta-sus).

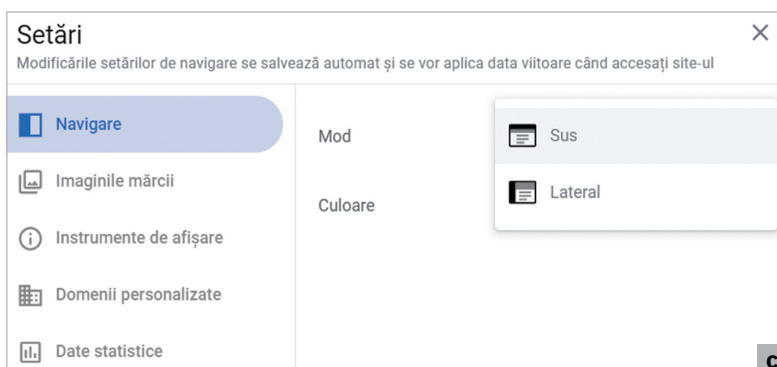


Puteți muta meniul de navigare pe partea stângă (pe verticală) dacă doriți, dar trebuie să aveți mai mult de o pagină pe site-ul vostru pentru a schimba locul unde apare meniul de navigare.



Cum se poate schimba locul în care apare meniul de navigare (imaginea c):

Mergeți în partea de sus, pe zona cu numele site-ului, și faceți clic pe **Setări** , de unde puteți alege **Sus** (în partea de sus a paginii) sau **Lateral** (meniul va fi disponibil pe verticală, în partea stângă a site-ului).

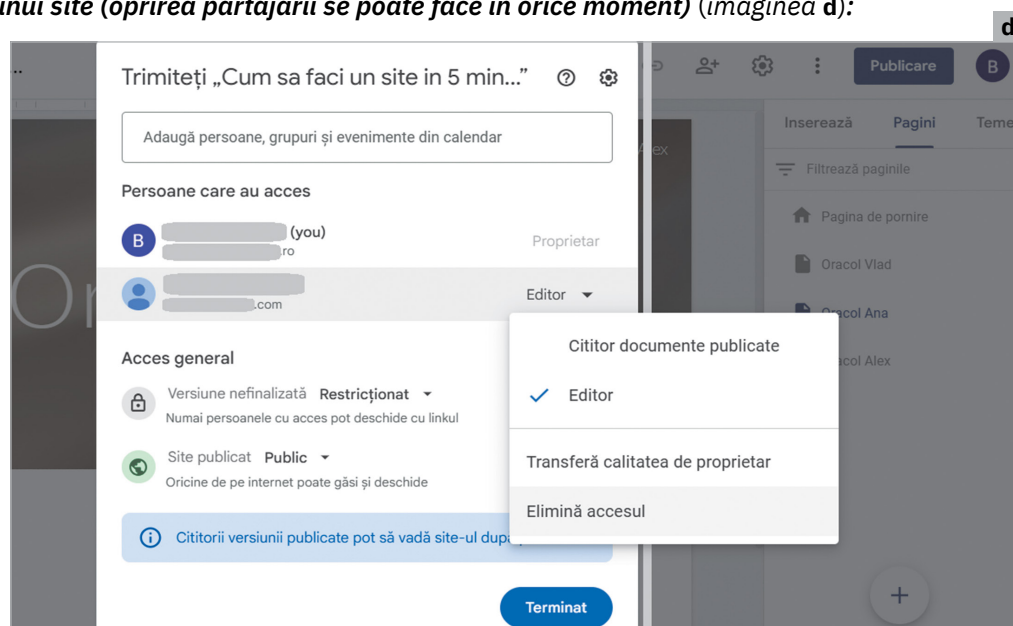


Partajarea și colaborarea pe Google Sites (cum să lucrăm împreună la un proiect)

În orice moment al lucrului la site v-ați afla, puteți să îl partajați cu colegii din grupul vostru de lucru pentru a colabora la definitivarea site-ului (și la conținut, și la structură). Am văzut la începutul lecției cum se realizează acest lucru.

Pentru a opri partajarea unui site (oprirea partajării se poate face în orice moment) (imaginea d):

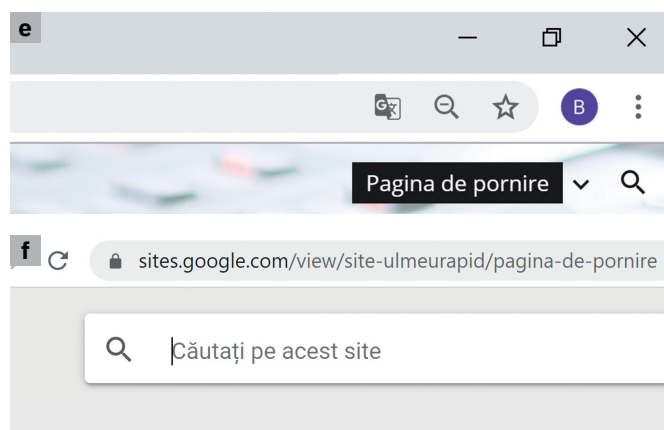
1. Deschideți site-ul și, în partea de sus, faceți clic pe **Trimite altor persoane**;
2. Din lista de sub **Persoane care au acces**, găsiți persoana pe care vreți să o înălțurați;
3. În dreptul numelui respectivei persoane, faceți clic pe săgeata în jos pentru a se afișa meniul și se alege **Elimină accesul**, apoi se apasă pe **Salvează**.



Căutarea într-un site

Când un site realizat în Google Sites are mai mult de o pagină, aveți la dispoziție o unealtă pentru a căuta un anumit conținut (text). Acest instrument nu este disponibil decât în **varianta publicată** a site-ului.

1. Mergeți în colțul din dreapta-sus al site-ului și apăsați **Deschide bara de căutare**  (imaginea e);
2. Introduceți textul căutat și apăsați tasta **Enter** sau butonul **Căutare**  din stânga barei (imaginea f).





Aplicații și investigații

► Realizarea unui site cu cinci pagini, împreună cu un coleg sau un prieten

1. Faceți un proiect nou pentru un nou site (accesați <https://sites.google.com> și faceți clic pe butonul + din dreapta-jos).
2. Scrieți numele proiectului.
3. Mergeți la bara de unelte, la tabul **Pagini**.
4. Creați patru pagini (+, **Pagină nouă** și introduceți numele pentru fiecare dintre cele patru pagini).
5. Faceți una dintre pagini să fie subpagină a paginii principale (pagina **Acasă**): trageți respectiva pagină peste pagina căreia i se va subordona.
6. Acordați unui coleg drept de editare pe site-ul vostru: faceți clic pe butonul **Trimite altor persoane**, introduceți adresa de e-mail a colegului cu care veți lucra la site și apăsați butonul **Trimite**. Persoana respectivă va primi pe e-mail invitația și un link de acces. Clic pe acel link, iar el va putea să înceapă lucrul.
7. Încărcați (împreună) conținut pe toate cele patru pagini ale site-ului (text, imagini, o foaie de calcul creată în Excel, o listă etc.).
8. Apăsați butonul **Vizualizare** pentru a vedea cum va apărea site-ul pentru vizitatori și, când considerați că totul este pus la punct, apăsați butonul **Publicare**. De acum, site-ul vostru este online.



Exersați

1. Inițiați un proiect pentru crearea unui site web în Google Sites. Oferiți-i acces unui prieten sau unui coleg pentru a putea lucra și el la acest proiect.
2. Dacă aveți create patru pagini în site-ul vostru, cum procedați pentru a face o pagină să devină subpagină a uneia dintre celelalte trei?
3. Completați, în caiete, cuvintele lipsă din textul următor:



Dacă avem un site cu mai multe . . . , vizitatorii site-ului vor putea utiliza meniul de . . . pentru a trece de la o . . . la alta. În mod implicit, meniul de . . . se află în partea de sus a site-ului (în colțul din . . . -sus). Putem muta . . . de navigare pe partea stângă (pe . . .), dar trebuie să avem . . . de o pagină pe site pentru a schimba locul unde apare . . . de

4. Care dintre opțiunile următoare nu există între opțiunile disponibile pentru o pagină web în Google Sites?
 - a. Setări ca pagină principală;
 - b. Duplicați pagina;
 - c. Adăugați o subpagină;
 - d. Mutați pagina;
 - e. Ascundeți din bara de navigare;
 - f. Ștergeți.
5. În Google Sites, odată create paginile unui site web într-o anumită ordine, se mai poate schimba ordinea lor? Dacă da, cum?



Portofoliu



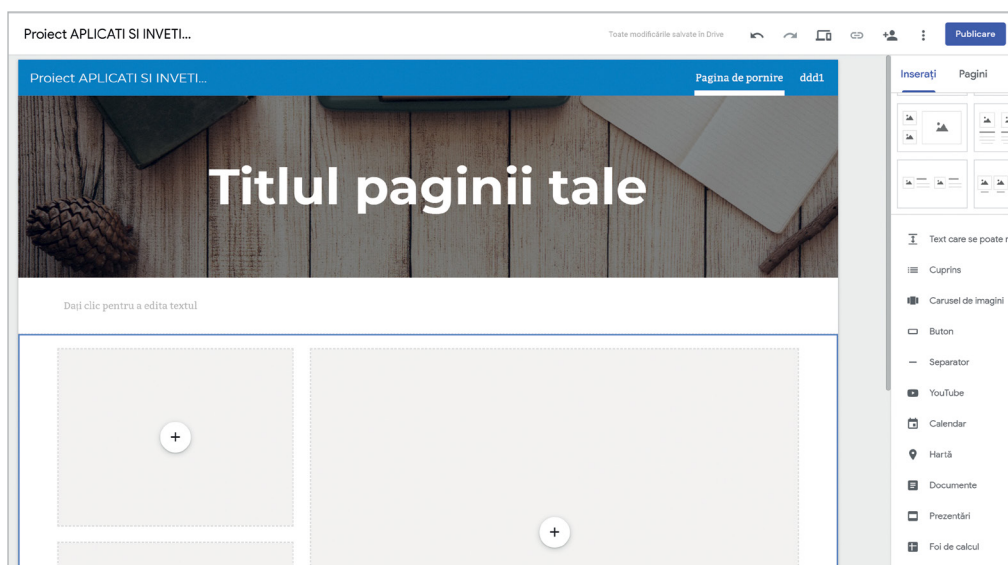
Împărțiți colectivul clasei în cinci grupuri. Realizați un site web în care să prezentați cinci locuri deosebite din Munții Carpați. Fiecare grup trebuie să pregătească o pagină cu una dintre atracțiile munților noștri. Unificați, apoi, totul într-un singur site intitulat „Carpații – munții noștri”. Setări culorile folosite în aspectul paginii astfel încât să fie în concordanță cu subiectul ales.

Recapitulare

1. Alegeți varianta corectă. Ce este Google Sites?
 - a. un limbaj de programare;
 - b. un program de creat site-uri web;
 - c. un program de editat texte.
2. Care sunt cele patru elemente de structură ale unei pagini web?
3. O pagină web poate exista și poate funcționa fără unul dintre aceste trei elemente: antet, corp, subsol?
4. Ce reprezintă termenul **slider** în terminologia utilizată în web design?
5. Realizați, în caiete, legătura dintre denumirile următoarelor trei butoane din interfața Google Sites și descrierile corespunzătoare:

a. Copiați linkul către site-ul publicat	1. Buton prin care putem invita prieteni sau colegi să lucreze împreună cu noi la același proiect – le dăm acces la proiectul nostru (lucru colaborativ).
b. Trimiteți altor persoane	2. Afîșează diverse alte meniuri pe care le avem la dispoziție, de la Statistici privind site-ul sau Duplicați site-ul până la Ajutor .
c. Mai multe	3. Copiază (în memorie) linkul către site, iar cu ajutorul opțiunii Lipește putem transmite linkul site-ului către oricine dorim.

6. Care dintre următoarele acțiuni sunt posibile pentru editarea imaginilor dintr-un site realizat cu Google Sites?
 - a. Redimensionarea unei imagini;
 - b. Mutarea unei imagini;
 - c. Decuparea unei imagini;
 - d. Înlocuirea unei imagini.
7. Alegeți varianta corectă. Locul în care apare meniul de navigare al unui site realizat în Google Sites poate fi setat să apară:
 - a. în partea de sus a site-ului;
 - b. pe verticală, în partea dreaptă a site-ului;
 - c. pe verticală, în partea stângă a site-ului.
8. Care dintre următoarele opțiuni nu sunt valabile pentru pagina principală a unui site realizat în Google Sites?
 - a. Setați ca pagină principală;
 - b. Duplicați pagina;
 - c. Adăugați o subpagină;
 - d. Ascundeți din bara de navigare;
 - e. Ștergeți.
9. Cărui program de realizare pagini web îi aparține interfața din imaginea de mai jos?



10. Realizați în Google Sites o pagină web despre rasele de cai românești, folosind tema (șablonul) „Aristotel”.

1. Ce este o pagină web?

2. Care dintre următoarele elemente pot fi introduse într-un site web?

- | | | |
|------------|----------------|----------------|
| a. tabele; | c. pictograme; | e. fotografii; |
| b. desene; | d. texte; | f. modele 3D. |

3. Realizați, în caiete, corespondența dintre noțiunile date și descrierile lor din tabelul de mai jos:

a. Corpul paginii web	1. acel element care anunță conținutul unei pagini web; îl regăsim, alături de o iconiță sugestivă, în fila browserului corespunzătoare paginii web
b. Antetul paginii web	2. este ultima parte a paginii; poate să conțină câteva informații sumare: un meniu secundar, date de contact și linkuri către platforme sociale
c. Titlul paginii web	3. zona în care găsim informația principală; aceasta este pe primul loc în ceea ce privește spațiul ocupat din pagină
	4. zona în care găsim numele paginii și elemente de identitate vizuală

4. Alegeți, din lista următoare, aplicațiile dedicate creării de pagini web:

- | | | |
|-----------------|------------------|-------------------------|
| a. Dreamweaver; | c. MS Word; | e. Google Web Designer; |
| b. WordPress; | d. Google Sites; | f. Wix. |

5. Explicați pe scurt ce trebuie făcut pentru a putea lucra mai mulți colegi la un site web în Google Sites.

6. Numiți două dintre opțiunile pe care le avem la dispoziție pentru modificarea tipului de antet în Google Sites.

7. Scrieți, în caiete, **DA** sau **NU** pentru fiecare dintre afirmațiile de mai jos referitoare la Google Sites:

- | | |
|--|----------------|
| a. Textul editat poate fi: drept (simplu), aplecat, îngroșat, subliniat. | DA / NU |
| b. Textul editat poate să aibă diferite feluri de font, precum și diferite culori. | DA / NU |
| c. Într-un text din pagina web putem insera și o hiperlegătură (hyperlink). | DA / NU |

8. Completați, în caiete, enunțurile următoare, folosind cuvintele potrivite:

- a. Titlul paginii este de ajutor atât pentru . . . , cât și pentru . . . de căutare care vor citi și prelucra această informație. Este important ca titlul paginii să fie . . . ales și concis formulat pentru a fi clar ce . . . se găsesc în pagina respectivă.
- b. O secțiune specială este . . . paginii. Pentru a adăuga o secțiune de . . . (în engleză, *footer*), mergeți în josul paginii și faceți clic pe **Adăugați** Secțiunea de . . . va apărea pe toate paginile site-ului vostru. Mergeți la ea și faceți clic pe **Editați** . . . pentru a o modifica.

9. Ce număr maxim de pagini poate conține un site construit cu Google Sites?

- | | | | |
|-------|--------|---------|---------------|
| a. 1; | b. 50; | c. 100; | d. nelimitat. |
|-------|--------|---------|---------------|

10. Ce sunt și la ce se folosesc cele trei taburi din bara de unelte Google Sites?

CUM VĂ APRECIĂȚI?

1. Evaluați-vă singuri rezultatele utilizând grila de punctaje.
2. Scrieți cum vă simțiți după ce ați rezolvat acest test, alegând unul dintre cuvintele de mai jos:
 - nemulțumit(ă);
 - mulțumit(ă);
 - încântat(ă).



Punctaje:

1 – 10 puncte;	6 – 10 puncte;	Insuficient: 0-50 de puncte
2 – 10 puncte;	7 – 10 puncte;	Suficient: 51-60 de puncte
3 – 10 puncte;	8 – 10 puncte;	Bine: 61-80 de puncte
4 – 10 puncte;	9 – 10 puncte;	Foarte bine: 81-90 de puncte
5 – 10 puncte;	10 – 10 puncte.	Excelent: 91-100 de puncte

Punctaj maxim: 100 de puncte. ● Timp de lucru: 50 de minute.

Răspunsurile se găsesc la finalul manualului.

Proiect: Crearea unui site al clasei

Descrierea proiectului

În cadrul proiectului **Site-ul clasei**, toți elevii din clasă vor lucra colaborativ pentru a crea un site virtual al clasei, utilizând Google Sites.

Timpul de lucru

Alocat de cadrul didactic.

Cerințe pentru realizarea proiectului

1. Structura site-ului

- Pagina principală va conține o introducere și o prezentare generală a clasei.
- Fiecare elev își va crea o pagină personală cu numele său, o fotografie și o scurtă descriere (pasiuni, aspirații, hobby-uri).
- Secțiunea **Realizările Clasei** va conține rezultatele obținute la examene, participări la concursuri, activități extracurriculare, acțiuni sociale.
- Secțiunea **Galerie Foto** va conține imagini de la diferite evenimente și activități la care au participat elevii clasei.

2. Colaborarea și organizarea pentru realizarea site-ului

- Un grup de maximum 3 elevi coordonatori va organiza structura generală a site-ului.
- Fiecare elev este responsabil de propria pagină.
- Alocarea elevilor pentru realizarea celorlalte pagini ale site-ului va fi stabilită de coordonatori.

3. Designul și aspectul vizual al site-ului

- Se va utiliza un stil unitar de culori și fonturi.
- Se vor adăuga imagini și elemente grafice relevante, pentru care există drepturi de autor sau consimțământul celor care apar în imagini pentru ca acestea să fie postate online.
- Paginile trebuie să fie clare, atractive și ușor de navigat.

Cum se face evaluarea

Site-ul final va fi prezentat profesorului și colegilor din alte clase. Profesorul va evalua site-ul și îi va acorda un punctaj, utilizând următoarele criterii de punctare:

Criteriu	Punctaj (maxim)
Organizarea și structura site-ului	15 puncte
Calitatea paginilor individuale	25 de puncte
Prezentarea realizărilor clasei	15 puncte
Design atractiv și coerent	10 puncte
Colaborare eficientă între colegi	10 puncte
Corectitudinea textelor și claritatea informațiilor	10 puncte
Finalizare la timp	5 puncte
Puncte din oficiu	10 puncte
Total:	100 de puncte

Premii și recunoaștere

- Echipa cu cel mai bun proiect va primi **Diploma de excelență în planificare digitală**, iar toate celelalte echipe participante vor primi diplome de participare.
- Cele mai bune trei proiecte vor fi prezentate în cadrul unui eveniment special.



Autoevaluarea proiectului

- Am respectat cerințele proiectului:
 - În întregime
 - Parțial
- În timpul de lucru alocat am terminat proiectul:
 - În întregime
 - Parțial
- Nivelul de documentare în vederea elaborării proiectului este:
 - Foarte bun
 - Bun
 - Slab
- Comunicarea cu colegii a fost:
 - Foarte bună
 - Bună
 - Slabă
- Nivelul de creativitate la realizarea proiectului a fost:
 - Foarte bun
 - Bun
 - Slab
- Proiectul le-a plăcut colegilor:
 - În mare măsură
 - În măsură normală
 - În mică măsură

U3

Algoritmi



Lecția 1

Lecția 2

Lecția 3

Lecția 4

Lecția 5

Lecția 6

Lecția 7

Lecția 8

Lecția 9

Lecția 10

Lecția 11

Lecția 12

Recapitulare

Autoevaluare

Proiect

58-61 Structura decizională (alternativă) – recapitulare

62-65 Structura repetitivă – recapitulare

66-67 Utilizarea algoritmilor în robotică

68-69 Șiruri de valori. Generarea șirurilor de numere cu proprietăți simple

70-71 Generarea șirurilor de numere pe baza unor reguli

72-73 Operații cu șiruri de valori: citire, parcurgere, afișare

74-75 Verificarea unor proprietăți ale elementelor șirului

76-77 Numărarea elementelor care îndeplinesc o condiție

78-80 Verificarea îndeplinirii unor proprietăți de către toate elementele unui șir

81-82 Căutarea unui element în șir de valori. Verificarea existenței unui anume element

83-85 Minimul și maximum dintr-un șir de valori

86-87 Suma și produsul elementelor unui șir de valori

88

89

90 Liceul preferat

1 Structura decizională (alternativă) – recapitulare



Amintiți-vă



Structura decizională este structura utilizată în cazul în care avem de executat acțiuni diferite în funcție de rezultatul unei decizii (al evaluării unei condiții).

	Pseudocod	Sintaxa C++
Structura decizională (alternativă) cu două ramuri (sunt precizate acțiuni pe două fire de execuție alternative)	<pre> Dacă condiție C Atunci secvență de instrucțiuni A Altfel secvență de instrucțiuni B </pre>	<pre> if (condiție C) { secvență de instrucțiuni A; } else { secvență de instrucțiuni B; } </pre>
Structura decizională (alternativă) cu o singură ramură (sunt precizate acțiuni pe un singur fir de execuție, lipsește alternativa)	<pre> Dacă condiție C Atunci secvență de instrucțiuni A </pre>	<pre> if (condiție C) { secvență de instrucțiuni A; } </pre>
Structura decizională multiplă, cu mai multe ramuri (ramura „Altfel” poate să lipsească)	<pre> În cazul că expresie selector E Cazul expr_constantă_1: secvență de instrucțiuni 1 [ieșire] Cazul expr_constantă_2: secvență de instrucțiuni 2 [ieșire] Cazul expr_constantă_n-1: secvență de instrucțiuni n-1 [ieșire] [Altfel secvență de instrucțiuni n] </pre>	<pre> switch (expresie selector E) { case expr_constantă_1: secvență de instrucțiuni 1; [break;] case expr_constantă_2: secvență de instrucțiuni 2; [break;] case expr_constantă_n-1: secvență de instrucțiuni n-1; [break;] [default: secvență de instrucțiuni n;] } </pre>

Observații legate de sintaxa C++:

1. Pentru structurile decizionale cu una sau două ramuri, dacă secvența de instrucțiuni este formată dintr-o singură instrucțiune, atunci acoladele pot lipsi.
2. În structura decizională multiplă, valorile expresiilor constante (*expr_constantă_X*) trebuie să fie întregi și distincte. Instrucțiunea **break** este opțională, aceasta cauzează ieșirea imediată din **switch**.



Descoperiți

Scrieți în Code::Blocks programele propuse la punctele **A** și **B** și urmăriți pas cu pas funcționarea lor. Redescoperiți logica de execuție a unui program care include structuri decizionale.

A. Analiza informațiilor din CNP (cod numeric personal)

În cadrul unui proces de votare, este analizat CNP-ul fiecărui alegător. Realizați un program care citește de la tastatură un număr de 13 cifre reprezentând CNP-ul unei persoane și afișează dacă aceasta este de sex masculin sau feminin și data sa de naștere sub forma AA.LL.ZZ, unde AA sunt ultimele două cifre din an, LL este luna și ZZ este ziua.

Exemplu: **Intrare:** n = un număr de forma 2780712XXXXXX, unde X sunt alte cifre
Ieșire: Sex feminin 78 07 12

Indicații: Pentru a înțelege cum se calculează, priviți forma unui CNP din imaginea alăturată.



Cifrele importante pentru programul de față sunt următoarele:

S	sexul masculin/feminin al persoanei; sexul feminin este reprezentat de numerele pare
AA	ultimele două cifre din anul nașterii persoanei
LL	luna nașterii persoanei, în format de două cifre (exemplu: dacă luna nașterii este 5, atunci LL este 05)
ZZ	ziua nașterii persoanei, în format de două cifre (exemplu: dacă ziua nașterii este 7, atunci ZZ este 07)

Pentru rezolvarea problemei, trebuie să extragem prima cifră (pentru a afișa sexul persoanei) și, respectiv, următoarele șase cifre (pentru a afișa data nașterii).

Date de intrare: n – variabilă de tip long long int, reprezintă CNP (număr de 13 cifre)

Date de ieșire: s – variabilă de tip întreg, reprezintă cifra care indică sexul persoanei

z – variabilă de tip întreg, reprezintă ziua nașterii persoanei

l – variabilă de tip întreg, reprezintă luna nașterii persoanei

a – variabilă de tip întreg, reprezintă anul nașterii persoanei

Programul C++ este următorul:

```
#include <iostream>
using namespace std;

int main()
{
    long long int cnp;
    int s, z, l, a;
    cout<<"Introduceti CNP:";
    cin>>cnp;

    s=cnp/1000000000000;
    if (s%2==0) cout<<"Sex feminin ";
    else cout<<"Sex masculin ";

    a=cnp/10000000000%100;
    if (a<10) cout<<"0"<<a<<" ";
    else cout<<a<<" ";

    l=cnp/100000000%100;
    if (l<10) cout<<"0"<<l<<" ";
    else cout<<l<<" ";
}
```

```
z=cnp/1000000%100;
if (z<10) cout<<"0"<<z<<" ";
else cout<<z<<" ";

return 0;
}
```



Specimen carte de identitate din România



Aplicați și investigați

- În programul „Analiza informațiilor din CNP” ștergeți liniile de cod: `if (l<10) cout<<"0"<<l<<" "; else cout<<l<<" ";`. Scrieți în locul lor `cout<<l<<" "`. La rularea programului pentru un CNP de forma 2780712XXXXXX, veți obține ca rezultat: Sex feminin 78 7 12. Care este explicația?



Descoperiți

B. Oracolul din calculatorul tău

Calculatorul poate deveni un adevărat oracol. Vă puteți uimi prietenii și le veți arăta ce poate face un calculator! Scrieți un program C++, în care un prieten al vostru va putea introduce anul nașterii sale, și i se va afișa imediat un mesaj, o „previziune” amuzantă, în funcție de acesta.

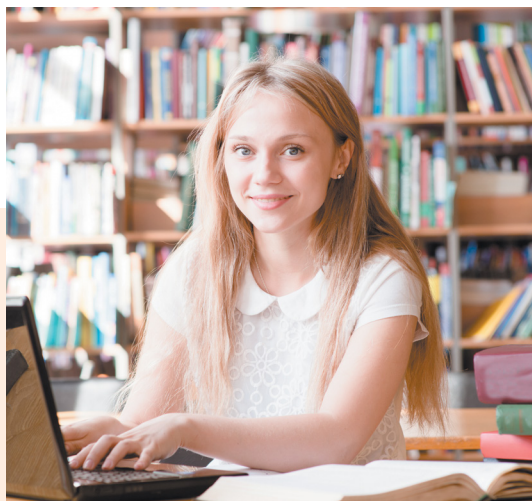
Indicații: Pentru a rezolva acest program, vom nota cu **a** variabila care va stoca anul introdus și vom calcula ultima cifră a acestuia, într-o variabilă **uc**. Ultima cifră a anului poate fi un număr cuprins între 0 și 9; pentru fiecare dintre aceste numere vom afișa un alt mesaj. Pentru că avem 10 cazuri, câte unul pentru fiecare cifră de la 0 la 9, vom utiliza o instrucțiune decizională multiplă **switch**. Puteți pune orice mesaje doriți, cât mai hazlii cu putință! Lăsați-vă imaginația să zboare!

Date de intrare: a – variabilă de tip întreg, reprezintă anul introdus

Date de manevră: uc – variabilă de tip întreg, reprezintă ultima cifră a anului introdus

Programul în C++ este următorul:

```
#include <iostream>
using namespace std;
int main()
{
    int a, uc;
    cout<<"Introdu anul nasterii tale: ";
    cin>>a;
    uc=a%10;
    cout<<"Oracolul spune: ";
    switch (uc)
    {
        case 0:  cout << "Vei descoperi un talent pe care ";
                  cout << "nici nu stiai ca il ai!";
                  break;
        case 1:  cout << "Esti mai pregatit decat crezi. ";
                  cout << "Doar incearca si vei vedea!";
                  break;
        case 2:  cout << "Un prieten adevarat te va surprinde ";
                  cout << "cand ai cea mai mare nevoie.";
                  break;
        case 3:  cout << "Un mesaj neasteptat iti va aduce un ";
                  cout << "zambet sincer pe fata.";
                  break;
        case 4:  cout << "O prietenie veche se va intari, ";
                  cout << "iar una noua va incepe curand!";
                  break;
        case 5:  cout << "Vei realiza ceva care iti va dovedi ";
                  cout << "ca esti pe drumul cel bun!";
                  break;
        case 6:  cout << "Vei straluci la examenul de Evaluare Nationala!";
                  break;
        case 7:  cout << "Vei avea parte de o surpriza frumoasa ";
                  cout << "de la un coleg neasteptat!";
                  break;
        case 8:  cout << "Un profesor te va lauda in fata clasei ";
                  cout << "pentru ambitia ta!";
                  break;
        case 9:  cout << "Un vis important al tau va prinde ";
                  cout << "contur mai repede decat crezi!";
    }
    return 0;
}
```



Știați că...?

Oracolele jucau un rol foarte important în Grecia antică. Cel mai cunoscut este Templul lui Apollo din Delphi, cunoscut și sub numele *Oracolul din Delphi*. Sub templu exista, conform miturilor, o peșteră sacră sau o crăpătură în pământ, despre care se spunea că este o cale către lumea de apoi. Preoteasa Pythia intra în stare de transă și spunea diverse profeții în numele zeilor.





Aplicații și investigații

- În programul „Oracolul din calculatorul tău” eliminați din cadrul structurii decizionale **switch** toate instrucțiunile **break**; apoi rulați din nou programul. Observați ce se afișează de fiecare dată. Explicați de ce.
- Lucrați în perechi.** Modificați în programul „Oracolul din calculatorul tău” instrucțiunea `uc=a%10`; în `uc=a%100%20`. Care este rezultatul? Completați, apoi, pe rând, mesajele „previziune” până la 19 și rulați programul pentru fiecare membru al echipei în parte. Așa puteți mări diversitatea mesajelor afișate, amuzându-vă în același timp.



Exersați

- Dați exemplu de un program C++ pentru rezolvarea căruia este recomandat să utilizați instrucțiunea decizională multiplă.
- Realizați un program care calculează suma a două numere întregi mai mici decât 1 000 introduse de la tastatură – dacă numerele sunt divizibile cu 7, respectiv produsul lor – dacă acestea sunt pare. În celelalte cazuri afișați „Alte valori”.

Exemplu: **Intrare:** 4 8 **Ieșire:** 32 **Intrare:** 7 14 **Ieșire:** 21
 Intrare: 14 28 **Ieșire:** 42 392 **Intrare:** 3 8 **Ieșire:** Alte valori

- Se introduc de la tastatură valori pentru trei variabile de tip întreg: **a, b, c**. Scrieți în caiete care dintre seturile de valori prezentate mai jos conduc la afișarea unui număr egal cu valoarea de intrare a variabilei **a** în urma executării secvenței de cod alăturate.

- a=26 b=3 c=13**
- a=26 b=2 c=12**
- a=15 b=3 c=12**
- a=26 b=3 c=12**

```
int a, b, c;
cin>>a>>b>>c;
if (a>15)
    if (b>2)
        if (c>13) cout<<a%c+b;
        else cout<<a*b*c;
    else cout<<a*b+c;
else cout<<c*b*a;
```

- Care dintre următoarele secvențe de cod afișează valoarea 3, dacă variabila **a** primește la intrare valoarea 3 și variabila **b**, valoarea 4?

a. `int a, b;`
`cin>>a>>b;`
`if (a>5)`
`b++;`
`b--;`
`cout<<b;`

b. `int a, b;`
`cin>>a>>b;`
`if (a>5);`
`b++;`
`b--;`
`cout<<b;`

c. `int a, b;`
`cin>>a>>b;`
`if (a>5);`
`else b++;`
`b--;`
`cout<<b;`

Portofoliu



Realizați un program care să afișeze „Da.” dacă trei numere naturale **a, b, c** pot fi laturile unui triunghi, și „Nu.” altfel. De asemenea, afișați în continuare dacă acesta este dreptunghic/obtuzunghic/ascuțitunghic.

Exemplu:

Intrare: a=3 b=4 c=5, **Ieșire:** Da. Triunghiul este dreptunghic.;

Intrare: a=3 b=4 c=4, **Ieșire:** Da. Triunghiul este ascuțitunghic.;

Intrare: a=3 b=4 c=6, **Ieșire:** Da. Triunghiul este obtuzunghic.;

Intrare: a=3 b=3 c=7, **Ieșire:** Nu.

Indicație:

Dacă pătratul celei mai mari laturi a unui triunghi este mai mic decât suma pătratelor celorlalte două laturi, atunci triunghiul este ascuțitunghic, dacă este egal cu suma pătratelor celorlalte două laturi, atunci este triunghi dreptunghic (teorema lui Pitagora) și dacă este mai mare decât suma pătratelor celorlalte două laturi, atunci este obtuzunghic.



2 Structura repetitivă – recapitulare



Amintiți-vă

Există trei tipuri de structuri repetitive:

1. Structura repetitivă cu număr necunoscut de pași:
 - a. Condiționată anterior;
 - b. Condiționată posterior;
2. Structura repetitivă cu număr cunoscut de pași.

	Pseudocod	Sintaxa C++
Structura repetitivă condiționată anterior	<pre> Cât timp (condiție C) execută secvență de instrucțiuni A </pre>	<pre> while (condiție C) { secvență de instrucțiuni A; } </pre>
Structura repetitivă condiționată posterior	<pre> Execută secvență de instrucțiuni A Cât timp (condiție C) </pre>	<pre> do { secvență de instrucțiuni A; } while (condiție C); </pre>
Structura repetitivă cu număr cunoscut de pași	<pre> Pentru X ← val inițială, val finală [, pas p] execută secvență de instrucțiuni A </pre> Observație: Dacă <pas> nu este specificat, atunci el va fi implicit 1.	1) pas este număr întreg pozitiv <pre> for (X=val inițială; X<=val finală; X=X+pas p) { secvență de instrucțiuni A; } </pre> 2) pas este număr întreg negativ <pre> for (X=val inițială; X>=val finală; X=X+pas p) { secvență de instrucțiuni A; } </pre>

Observații legate de sintaxa C++: Pentru toate cele trei structuri repetitive, dacă secvența de instrucțiuni este formată dintr-o singură instrucțiune, atunci acoladele pot lipsi.



Descoperiți

Scrieți în Code::Blocks programele propuse la punctele **A** și **B** și urmăriți pas cu pas funcționarea lor. Redescoperiți logica de execuție a unui program care include structuri repetitive.

A. Problema piramidei

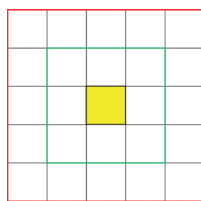
Egiptenii doresc să construiască o nouă piramidă, în trepte. Fiecare nivel al piramidei este construit din blocuri de piatră paralelipipedice cu baza de 1 m × 1 m și înălțimea variabilă în funcție de nivel. Pe primul nivel de jos sunt $n \times n$ blocuri de piatră cu înălțimea de 1 m, unde n este un număr natural diferit de 0. Pe al doilea nivel sunt $(n-2) \times (n-2)$ blocuri de piatră cu înălțimea de 2 m ș.a.m.d. Realizați un program care să citească de la tastatură numărul n și să afișeze numărul de niveluri ale piramidei, câte blocuri de piatră se folosesc și înălțimea piramidei în metri.

Exemplu:

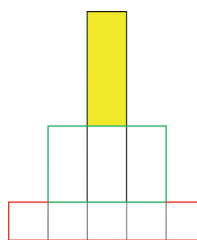
Intrare: $n=5$;

Ieșire: Nr. de niveluri este: 3; Nr. blocurilor de piatră folosite este: 35; Înălțimea piramidei este: 6 m

Pentru a înțelege cum anume se calculează rezultatele, priviți imaginile de mai jos ale unei piramide cu baza mare de 5 x 5 blocuri de piatră, văzută de sus și din lateral.



Vedere de sus



Vedere laterală

Înălțime bloc de piatră = 3 m

Înălțime bloc de piatră = 2 m

Înălțime bloc de piatră = 1 m



Indicații:

Date de intrare: n – variabilă de tip întreg, este numărul blocurilor de piatră de pe latura primului nivel de jos

Date de manevră: nrP – variabilă de tip întreg, reprezintă numărul de blocuri de piatră pe un nivel

Date de ieșire: $nrNiv$ – variabilă de tip întreg, reprezintă numărul de niveluri ale piramidei

$nrTP$ – variabilă de tip întreg, reprezintă numărul total de blocuri de piatră folosite

h – variabilă de tip întreg, reprezintă înălțimea în metri a piramidei

Algoritmul se bazează pe faptul că se repetă incrementarea numărului nivelurilor (**nrNiv**), calcularea numărului de blocuri de piatră de pe acel nivel (**nrP**), a numărului total de blocuri de piatră utilizate până la acel moment (**nrTP**) și a înălțimii piramidei până la acel moment (**h**), cât timp mai putem să ridicăm un nivel suplimentar ($n - 2 * nrNiv > 0$).

Observați următorul tabel pentru $n=5$:

Nr. nivel	Nr. blocuri pe o latură a nivelului actual	Nr. blocuri previzionat pt. o latură a următorului nivel
1	$5 - 2 * 0 = 5$	$5 - 2 * 1 = 3$
2	$5 - 2 * 1 = 3$	$5 - 2 * 2 = 1$
3	$5 - 2 * 2 = 1$	$5 - 2 * 3 = -1 \Rightarrow$ nu se mai poate construi

Programul C++ este următorul:

```
#include <iostream>
using namespace std;
int main()
{ int n, nrNiv=0, h=0, nrP=0, nrTP=0;
  cout<<"n="; cin>>n;
  do
  {
    nrNiv=nrNiv+1;;
    nrP=(n-2*(nrNiv-1))*(n-2*(nrNiv-1));
    nrTP=nrTP+nrP;
    h=h+nrNiv;
  } while (n-2*nrNiv>0);
  cout<<"Nr. de niveluri este: "<<nrNiv<<" ";
  cout<<"Nr. blocurilor de piatra folosite este: "<<nrTP<<" ";
  cout<<"Inaltimea piramidei este: "<<h<<" m";
  return 0;
}
```

Știați că...?



Cea mai veche piramidă din Egipt este Piramida în trepte din Saqqara sau Piramida lui Djoser. Se spune că a fost realizată de un înalt oficial, pe nume Imhotep, în secolul 27 î.H., și se remarcă prin noi standarde de construcție la vremea respectivă și prin stilul arhitectonic deosebit.



Aplicați și investigați

- ▶ În programul „Problema piramidei” introduceți la intrare un număr par pentru n . Rulați programul pas cu pas și observați câte blocuri de piatră are ultimul nivel în acest caz. Veți vedea că în cazul unui număr n impar veți avea pe ultimul nivel un bloc, iar în cazul unui n par veți avea pe ultimul nivel două blocuri pe o latură. Explicați de ce.



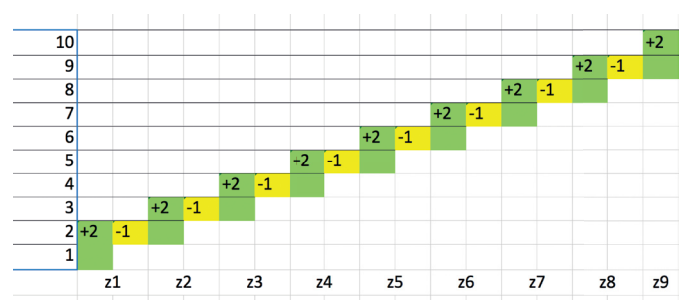
Descoperiți

B. Problema melcului

Un melc a căzut într-o fântână adâncă de n metri, unde n este un număr natural, $3 \leq n \leq 300$. El încearcă să iasă. Ziua urcă 2 m, iar noaptea cade 1 m. Scrieți un program care afișează în câte zile a ieșit melcul din fântână.

Exemplu: Intrare: $n=10$ **Ieșire:** Nr. de zile este: 9

Pentru a înțelege rezultatul, priviți imaginea alăturată în care $z_1, \dots, z_9$ sunt zilele scurse. În dreptul fiecărei zile figurează înălțimea la care a ajuns melcul.



Indicații:

Date de intrare: n – variabilă de tip întreg, reprezintă adâncimea fântânii

Date de manevră: u – variabilă de tip întreg, reprezintă distanța actuală până la care a urcat melcul

Date de ieșire: z – variabilă de tip întreg, reprezintă numărul de zile scurse până la acel moment

Programul C++ este următorul:

```
#include <iostream>
using namespace std;
int main()
{   int n, z, u;
    do
    {   cout<<"n= "; cin>>n;
    } while (n<3 || n>100);
    z=1; u=2; u=u-1;
    while (u<n)
    {   z=z+1; u=u+2;
        if (u<n) u=u-1;
    }
    cout<<endl; cout<<"Nr. de zile este: "<<z;
    return 0;
}
```



Aplicații și investigații

- Modificați „Problema melcului” astfel încât să introduceți de la tastatură numărul de metri urcați ziua de către melc, **du**, și numărul de metri coborâți noaptea, **dc**, unde **dc < du**, iar **dc** și **du** sunt numere naturale pozitive. Observați care este corelația dintre diferența **du-dc** și numărul de zile în care melcul iese din fântână sau dintre adâncimea fântânii și **du**, în condițiile în care **du-dc** este constant.



Descoperiți

C. Numere prime

Realizați un program care afișează dacă un număr natural mai mare decât 1 introdus de la tastatură este prim sau nu.

Exemplu: **Intrare:** $n=10$, **Ieșire:** Nr. nu este prim; **Intrare:** $n=17$, **Ieșire:** Nr. este prim.

Indicații:

Date de intrare: n – variabilă de tip întreg, reprezintă numărul analizat

Date de manevră: $nrDiv$ – variabilă de tip întreg, reprezintă numărul de divizori diferiți de 1 și de el însuși

Un număr natural n mai mare decât 1 este prim dacă nu are alți divizori diferiți de 1 și de el însuși. Cel mai mic divizor posibil dintre aceștia este 2, deci cel mai mare posibil este $n/2$. Astfel, în algoritm este suficient să încercăm numerele de la 2 până la $n/2$.

Programul C++ este următorul:

```
#include <iostream>
using namespace std;
int main()
{   int n, i, nrDiv;
    cout<<"n= "; cin>>n;
    nrDiv=0;
    for (i=2; i<=n/2; i++)
        if (n%i==0) nrDiv++;
    if (nrDiv==0) cout<<"Nr. "<<n<<" este prim.";
    else cout<<"Nr. "<<n<<" nu este prim.";
    return 0;
}
```





Aplicații și investigații

- În programul „Numere prime” modificați structura repetitivă **for**, astfel încât să fie `for (i=2; i<=n-1; i++)`. Rulați programul inițial și programul modificat și comparați timpii de execuție.



Exersați

1. Alegeți care dintre secvențele de cod următoare afișează valoarea lui 10!.

a. `p=1;`

`for (i=0; i<=10; i++)`

`p=p*i;`

`cout<<p;`

b. `p=1;`

`for (i=2; i<=10; i++)`

`p=p*i;`

`cout<<p;`

c. `p=1;`

`for (i=2; i=10; i++)`

`p=p*i;`

`cout<<p;`



2. Completați valoarea inițială care trebuie introdusă mai jos pentru a afișa 4 stelute ("****").

`int k, i;`

`k=1;`

`i=.....;`

`while (i<=10)`

`{ k=k*3;`

`i=i+2;`

`cout<<"*";`

`}`

`int i, l;`

`l=.....;`

`i=4;`

`do`

`{ if (i%2==0) {l++; i++;}`

`else {i=i+2; cout<<"*";}`

`} while (i<=l);`

3. Scrieți, în caiete, **A** (Adevărat) sau **F** (Fals) pentru fiecare propoziție de mai jos.

a. Secvențele următoare scrise în pseudocod sunt echivalente:

Cât timp (condiție C) **execută**
secvență de instrucțiuni A

Dacă (condiție C)
Atunci

Execută

secvență de instrucțiuni A

Cât timp (condiție C)

A / F

b. Secvențele următoare de cod vor afișa același rezultat:

`int i, j, k=1;`

`for (i=1; i<=10; i++)`

`for (j=1; j<=5; j++)`

`k++;`

`cout<<k;`

`int i, j, k=0;`

`for (i=1; i<=10; i++)`

`for (j=1; j<=5; j++)`

`k++;`

`cout<<k+1;`

A / F

c. Secvența de cod de mai jos afișează toate numerele naturale pare cuprinse în intervalul [1, 55].

`int n=55;`

`while (n>=1)`

`{ cout<<n<<" ";`

`n=n-2; }`

A / F

4. Se citesc de la tastatură mai multe numere naturale, până când se introduce valoarea 0. Realizați un program care afișează suma numerelor introduse care conțin exact trei cifre.

Exemplu: **Intrare:** 12 324 3 111 0 **Ieșire:** 435

5. Se citesc de la tastatură patru numere naturale. Realizați un program care afișează dacă numărul format din prima cifră a tuturor numerelor introduse este prim sau nu.

Exemplu: **Intrare:** 14 23 522 941 **Ieșire:** Nr. 1259 este prim

Intrare: 14 23 522 242 **Ieșire:** Nr. 1252 nu este prim



Portofoliu



Irina culege mere. Ea culege în prima zi 1 kg de mere, apoi în fiecare zi care urmează culege cu 1 kg în plus față de ziua anterioară. În câte zile va culege Irina necesarul de kilograme de mere pentru piață? Presupunem că la piață merge cu minim 10 kg de mere.

Exemplu: **Intrare:** 10 kg; **Ieșire:** Irina are nevoie de 4 zile în total.

3 Utilizarea algoritmilor în robotică



Amintiți-vă

Open Roberta Lab (<https://lab.open-roberta.org/>) este mediul online gratuit pe care îl puteți utiliza pentru a crea simulări cu roboți virtuali și are integrat un editor grafic, intuitiv, bazat pe blocuri, în stilul Scratch. Ne amintim că limbajul de programare grafic utilizat în platforma Open Roberta Lab pentru a programa roboții se numește **NEPO**.



Descoperiți

Vom păși în fascinanta lume a programării roboților virtuali și vom utiliza robotul NXT, robotul educațional realizat de către compania LEGO. Vom vedea două exemple de elaborare a codului sursă pentru a controla robotul didactic virtual prin utilizarea și interpretarea datelor primite de la senzorii acestuia: **senzorul de culoare** (*light sensor*), **senzorul de distanță** (*ultrasonic sensor*), **senzorul tactil** (*touch sensor*).

A. Ocolește obstacolele

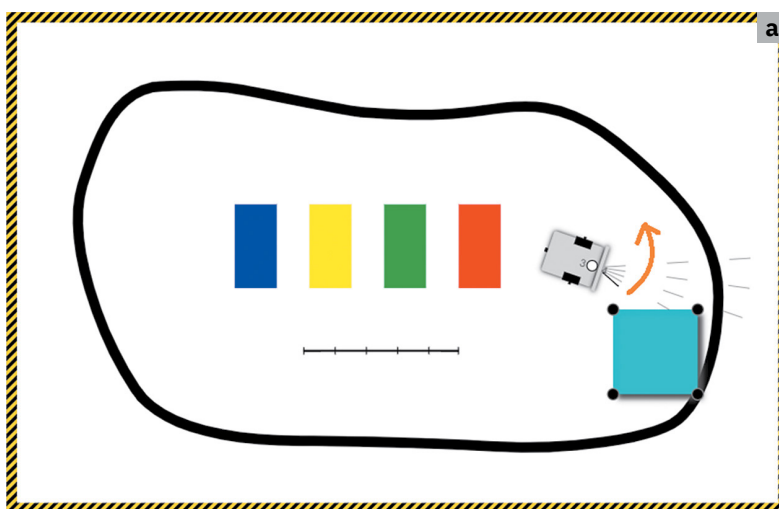
Vom realiza un program care controlează un robot ce **se deplasează și evită obstacolele** măsurând distanța față de acestea folosind un **senzor de distanță** (*ultrasonic sensor*) conectat la portul 4. Este un prim pas spre programarea comportamentelor autonome. Iată cum funcționează:

Program robotul	Algoritm de funcționare
	Pornește robotul. Se repetă la infinit: 1. Robotul măsoară distanța față de obstacol folosind senzorul ultrasonic conectat la portul 4. 2. Verifică dacă are suficient spațiu înainte: • Dacă distanța este mai mare de 20 cm, înseamnă că drumul este liber și robotul merge înainte cu o viteză de 30%. • Altfel (distanța este mai mică sau egală cu 20 cm), robotul detectează un obstacol și se întoarce spre stânga pentru a-l ocoli, tot cu o viteză de 30%.

Facem următoarele acțiuni în zona de simulare:

1. Alegem o scenă cât mai simplă, apăsând pe butonul
2. Desenăm un obstacol pătrat apăsând pe butonul
3. Pornim simularea apăsând pe butonul
4. Mișcăm, cu ajutorul mouse-ului, obstacolul, punându-l în fața robotului.

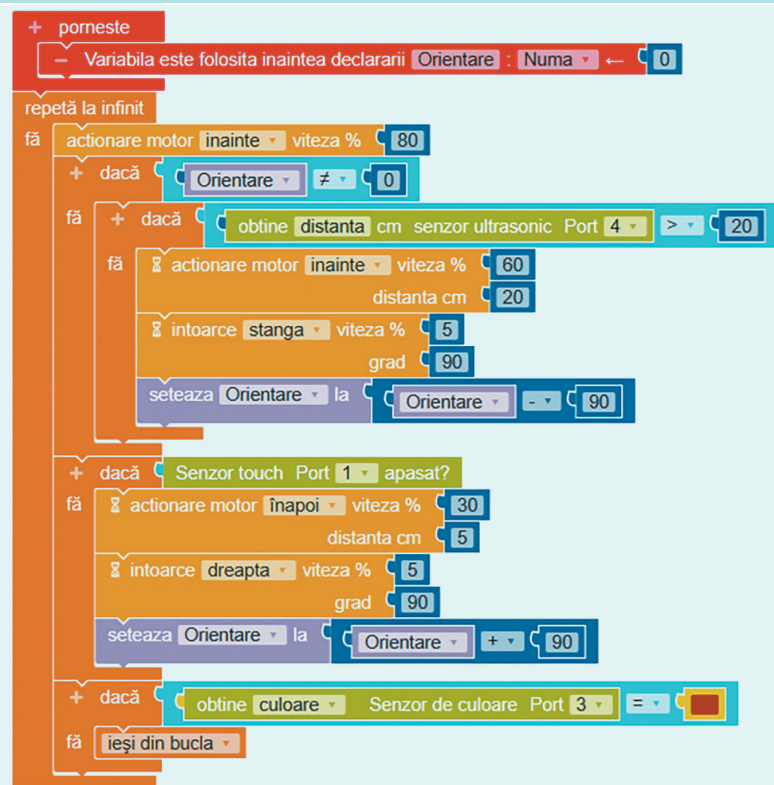
Rezultatul simulării este prezentat în *imaginea a*. Dacă în fața robotului apare obstacolul albastru, senzorul detectează când distanța este mai mică de 20 cm. În acel moment, robotul **nu mai merge înainte, ci se întoarce spre stânga** pentru a evita obstacolul. Apoi verifică din nou dacă drumul este liber și reia deplasarea.



B. Labirint

Programul următor controlează un robot echipat cu mai mulți senzori: **de distanță** (*ultrasonic sensor*), **tactil** (*touch sensor*) și **de culoare** (*light sensor*). Scopul este ca robotul să **navigheze într-un labirint**, evitând obstacolele și reacționând la culoarea roșie ca semnal de oprire. Folosim o variabilă „Orientare” pentru a reține direcția de mers a robotului.

Program robotul



Algoritm de funcționare

Pornește robotul, definind variabila **Orientare** cu valoarea inițială 0.

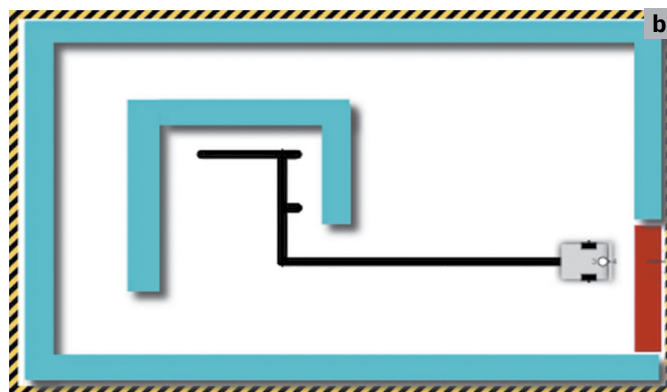
Se repetă la infinit:

1. Robotul începe să se deplaseze înainte cu o viteză de **80%**.
2. Dacă variabila **Orientare** are o valoare diferită de 0, robotul verifică dacă are suficient spațiu în față. **Dacă** distanța este **mai mare de 20 cm**, înseamnă că nu este obstacol și robotul:
 - se **deplasează înainte** cu viteză de **60%** pe o distanță de **20 cm**,
 - apoi **virează la stânga** cu viteză de **5%** și un unghi de **90°**,
 - scade 90 din valoarea variabilei „Orientare” (pentru a marca rotirea spre stânga).
3. **Dacă** senzorul tactil de la portul 1 detectează un obstacol, atunci:
 - robotul **merge înapoi** cu viteză de **30%** pe o distanță de **5 cm**,
 - apoi **virează la dreapta** cu viteză de **5%** și un unghi de **90°**,
 - adaugă 90 la valoarea variabilei „Orientare” (pentru a marca rotirea spre dreapta).
4. Robotul verifică apoi culoarea detectată de senzorul de culoare de la portul 3. **Dacă** se detectează culoarea roșie, robotul **iese din buclă** și oprește programul.

Facem următoarele acțiuni în zona de simulare:

1. Desenăm o scenă nouă care conține un labirint format din mai multe obstacole.
2. Activăm opțiunea de desenare a traiectoriei robotului la mișcarea acestuia, apăsând pe butonul .
3. Pornim simularea apăsând pe butonul .

▶ Rezultatul simulării este prezentat în **imaginea b**. Robotul navighează în labirint evitând obstacolele, rotindu-se alternativ stânga/dreapta în funcție de ce întâlnește. Când senzorul de culoare detectează **culoarea roșie**, robotul **se oprește** — semn că a ajuns la „ieșirea” din labirint.



Aplicații și investigații

1. Testați programele **A** și **B** în **simulator** și observați comportamentul roboților la întâlnirea obstacolelor.
2. Investigați cum ați modifica programul **A**, astfel încât robotul să vireze dreapta în loc de stânga atunci când întâlnește un obstacol.
3. Ce s-ar întâmpla dacă în loc de roșu am pune verde la ieșirea din labirint în programul **B**? Ce trebuie modificat în cod?



Exersați

1. Explicați în cuvintele voastre ce face robotul NXT în următoarele situații, conform programului **A**. **Ocolește obstacolele.**
 - a. Nu este niciun obstacol în fața lui.
 - b. Se apropie de un perete aflat la 15 cm.
2. În programul **B**, ce înseamnă faptul că robotul adună sau scade 90 în variabila „Orientare”?
3. Ce face robotul atunci când senzorul de culoare detectează culoarea **roșie**?
 - a. Se rotește la stânga.
 - b. Se oprește (iese din buclă).
 - c. Pornește din nou programul.



4 Șiruri de valori. Generarea șirurilor de numere cu proprietăți simple



Amintiți-vă

Un **șir de valori** este o succesiune de elemente de același tip. În matematică ați întâlnit până acum noțiunea de **șir de valori numerice** (șiruri de numere), ca o înșiruire de numere aflate într-o anumită ordine.

Cunoașteți, de exemplu, șirul numerelor naturale (0, 1, 2, 3, ...) sau șirul numerelor pare (0, 2, 4, 6, 8, 10, ...).



Descoperiți

Închipuiți-vă că aveți de rezolvat o problemă de matematică în care apare un număr foarte mare de numere. Cât timp ar trebui să alocăm rezolvării?

Pentru a fi eficienți, este necesar să avem un instrument digital, un program care să ne ajute.

Pentru identificarea câtorva șiruri de numere pe baza unor proprietăți, scrieți în Code::Blocks programele **A** și **B** propuse mai jos și urmăriți pas cu pas funcționarea lor.

A. Șir de numere pare – afișarea primelor n valori

► **Problemă:** Se dă un număr natural n . Afișați în ordine crescătoare primele n numere naturale pare nenule, unde $1 \leq n \leq 1000000$.

Exemplu: **Intrare:** $n=11$

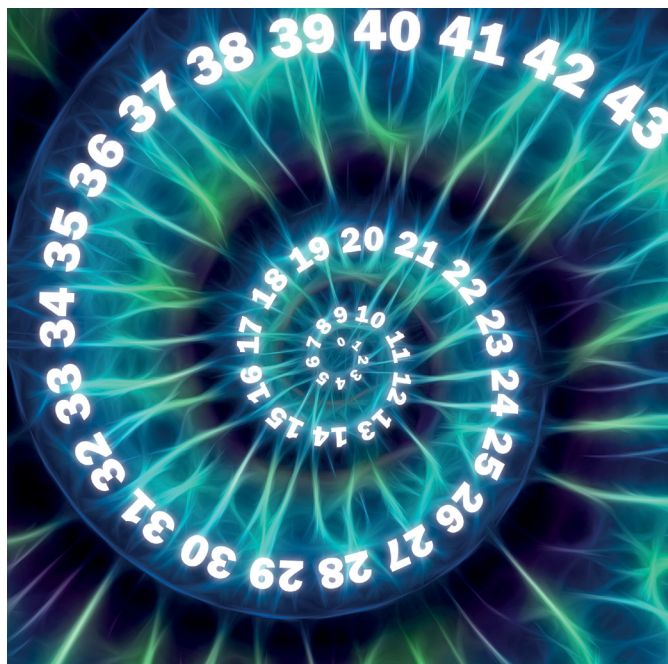
Ieșire: 2 4 6 8 10 12 14 16 18 20 22

Indicații:

Date de intrare: n – număr natural nenul, numărul elementelor care se vor afișa

Date de manevră: i – număr natural nenul, contor (al câtelea număr se afișează)

Date de ieșire: numerele din șir



Algoritmul în pseudocod	Programul în C++
<pre> algoritm SirNrPare_1 întreg n, i i ← 1 Scrie 'n=' Citește n Cât timp i ≤ n execută Scrie 2 * i, ' ' i ← i + 1 Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i=1; cout<<"n="; cin>>n; while (i<=n) { cout<<2*i<<' '; i++; } return 0; } </pre> n=11 2 4 6 8 10 12 14 16 18 20 22 Process returned 0 (0x0) execution time : 2.579 s Press any key to continue.

B. Șir de numere pare – afișarea valorilor mai mici decât n

Problemă: Se dă un număr natural n . Afișați în ordine crescătoare numerele naturale nenule mai mici sau egale cu n , unde $1 \leq n \leq 1000000$.

Exemplu: **Intrare:** $n=11$; **Ieșire:** 2 4 6 8 10

Indicații:

Date de intrare: n – număr natural nenul, maximul posibil al elementelor

Date de ieșire: i – număr natural nenul, reprezintă numărul care se afișează

Algoritmul în pseudocod	Programul în C++
<pre> algoritm SirNrPare_2 întreg n, i $i \leftarrow 2$ Citește n Scrie 'n=', n Cât timp $i \leq n$ execută Scrie i, ' ' $i \leftarrow i + 2$ Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i=2; cout<<"n="; cin>>n; while (i<=n) { cout<<i<<' '; i=i+2; } return 0; } </pre>

$n=11$
 2 4 6 8 10
 Process returned 0 (0x0) execution time : 2.503 s
 Press any key to continue.

Aplicați și investigați

- Inițializați variabila i cu 1 în programul **B**, apoi rulați-l. Ce observați? Ce șir de numere este generat?
- Identificați diferența dintre cerințele problemelor **A** și **B** („primele n ” vs. „mai mici sau egale cu n ”) și urmăriți diferențele de algoritm.
- Modificați programul **A** astfel încât să conțină o structură repetitivă cu număr cunoscut de pași.

Exersați

- Realizați un program C++ care să afișeze șirul primelor n numere naturale nenule divizibile cu 3, unde $n \leq 1000$.
Indicații: **Intrare:** 5; **Ieșire:** 3 6 9 12 15
- Realizați un program C++ care să afișeze șirul primelor numere naturale divizibile cu 3, mai mici decât un număr natural nenul n , introdus de la tastatură, unde $n \leq 1000$.
Indicații: **Intrare:** 20; **Ieșire:** 0 3 6 9 12 15 18
- Realizați un program C++ care să afișeze șirul primelor n numere naturale care au ultima cifră 8, unde $n \leq 1000$.
Exemplu: **Intrare:** 5; **Ieșire:** 8 18 28 38 48
- Realizați un program C++ care să afișeze șirul numerelor întregi cuprinse în intervalul $[-n, n]$, unde n este un număr natural nenul introdus de la tastatură, $n \leq 1000$.
Indicații: **Intrare:** 3; **Ieșire:** -3 -2 -1 0 1 2 3
- Se citesc două numere naturale nenule n și k de la tastatură, $1 \leq n \leq 1000$, $1 \leq k \leq 100$. Să se afișeze primii n multipli nenuli ai lui k .
Indicații: **Intrare:** 4 5; **Ieșire:** 5 10 15 20
- Se citesc două numere naturale nenule a și b ($a \leq b$, $1 \leq a, b \leq 10000$). Să se afișeze toate numerele impare din intervalul $[a, b]$.
Indicații: **Intrare:** 4 10; **Ieșire:** 5 7 9

5 Generarea șirurilor de numere pe baza unor reguli



Descoperiți

A. Șirul Fibonacci

Matematicianul italian Leonardo din Pisa, cunoscut și sub numele de Fibonacci, a popularizat un șir numeric ce îi poartă numele, *șirul Fibonacci*: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, Regula de formare este simplă: primele două valori sunt 0 și 1, iar fiecare termen nou se calculează ca suma ultimelor două ($1 = 1 + 0$, $2 = 1 + 1$, $3 = 2 + 1$, $5 = 3 + 2$, ...).

Dacă desenăm pătrate cu laturile egale cu valorile din șirul Fibonacci și unim colțurile opuse ale acestora prin arce circulare, se formează „spirală de aur” (*imagea a*). Putem observa această spirală și în natură, de exemplu, în cochiliile melcilor (*imagea b*), în dispunerea semințelor florii-soarelui (*imagea c*) sau în alte structuri naturale.

Problemă: Afișați primele n numere din șirul Fibonacci, unde n este un număr natural, $2 < n \leq 30$.

Exemplu: **Intrare:** 8 **Ieșire:** 0 1 1 2 3 5 8 13

Indicații:

Date de intrare:

n – număr natural nenul, numărul elementelor din șirul Fibonacci

Date de manevră:

i – număr natural nenul, contor (al câtelea număr urmează să fie găsit)

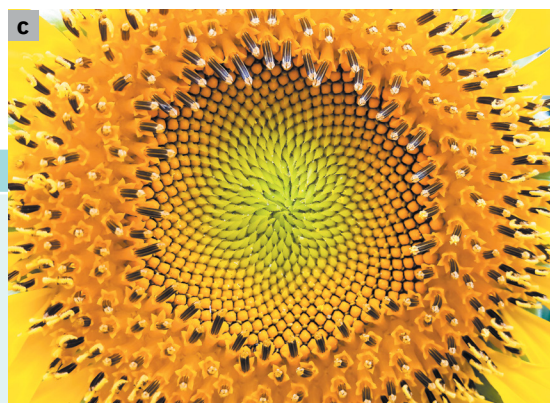
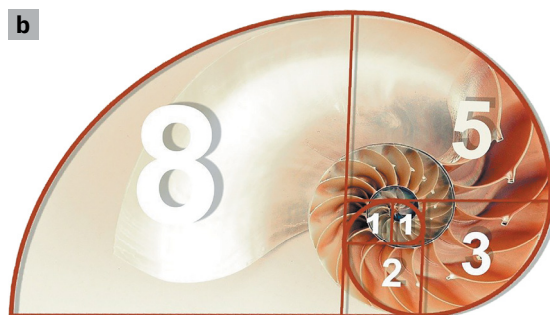
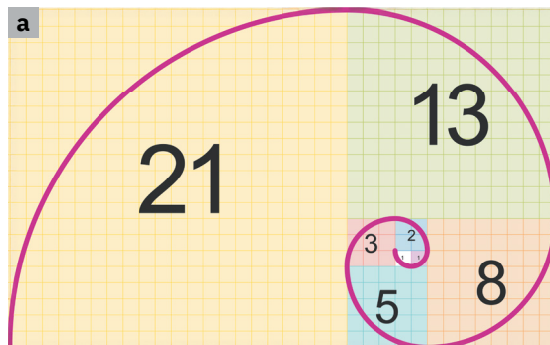
p – număr natural nenul (primul termen care intră în suma celor două anterioare)

d – număr natural nenul (al doilea termen care intră în suma celor două anterioare)

c – numărul curent de calculat, număr natural nenul

Date de ieșire:

p, d, c – termenii șirului



Algoritmul în pseudocod

algoritm Fibonacci

Întreg i, p, d, c, n

$i \leftarrow 3$

$p \leftarrow 0$

$d \leftarrow 1$

Citește n

Scrie $p, ', d, ''$

Pentru $i \leftarrow 3, n$ **execută**

$c \leftarrow p + d$

Scrie $c, ''$

$p \leftarrow d$

$d \leftarrow c$

Sfârșit

Programul în C++

```
#include <iostream>
using namespace std;

int main()
{
    int i=3,p=0,d=1,c,n;

    cin>>n;
    cout<<p<<" "<<d<<" "; //afisam primii doi termeni

    for (i=3; i<=n; i++) //pentru fiecare termen in parte
    {
        c=p+d; //calculam termenul urmator
        cout<<c<<" "; //afisam termenul nou calculat
        p=d; //actualizam termenul 1 din formula de calcul
        d=c; //actualizam termenul 2 din formula de calcul
    }
    return 0;
}
```

10

0 1 1 2 3 5 8 13 21 34

Process returned 0 (0x0) execution time : 5.315 s
Press any key to continue.

B. Șirul numerelor triunghiulare

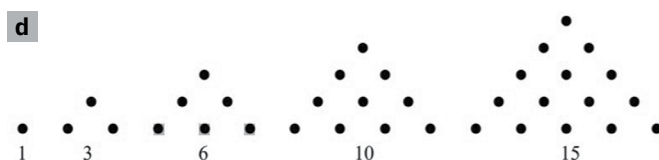
Lucrați în perechi. Desenați cu creioane colorate triunghiurile de mai jos, formate din biluțe (*imaginea d*). Numărul de bile care formează un triunghi se numește „**număr triunghiular**”. Dacă numărăm bilele pentru fiecare triunghi începând cu primul, observăm că aceste numere corespund unei reguli bine definite, evidențiate în tabelul din *imaginea e*.

Dacă scriem unul după altul aceste numere, vom obține un șir al „**numerelor triunghiulare**”.

Problemă: Afișați primele n numere triunghiulare, unde n este un număr natural, $1 < n \leq 100$.

Exemplu: **Intrare:** $n=10$; **Ieșire:** 1 3 6 10 15 21 28 36 45 55.

Indicații: Pentru a afișa șirul numerelor triunghiulare, aplicăm formula de calcul pentru fiecare element în parte, evidențiată în tabelul din *imaginea e*: elementul x al șirului este egal cu $(x*(x+1))/2$.



Nr. triunghiului	Nr. de bile care formează triunghiul
1	$(1*2)/2=1$
2	$(2*3)/2=3$
3	$(3*4)/2=6$
...	...
e n	$(n*(n+1))/2$



```
n=10
1 3 6 10 15 21 28 36 45 55
Process returned 0 (0x0)   execution time : 4.420 s
Press any key to continue.
```

Programul C++

```
#include <iostream>
using namespace std;
int main()
{ int n, i=1;
  cout<<"n="; cin>>n;
  while (i<=n)
  { cout<<(i*(i+1))/2<<" ";
    i++;
  }
  return 0;
}
```



Aplicați și investigați

- Aplicați principiul de formare a șirului Fibonacci (elementul x este egal cu suma ultimilor doi predecesori) și generați un nou șir pornind de la șirul numerelor triunghiulare astfel: primele elemente sunt 1 și 3, elementul x este egal cu suma elementelor $x-1$ și $x-2$ din șirul numerelor triunghiulare. Urmăriți cum cresc termenii șirului comparativ cu șirul Fibonacci original.

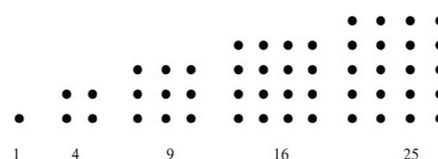
Indicații: 1, 3, 4 = 1 + 3, 9 = 3 + 6, ...



Exersați

- Dați exemple de două șiruri de numere ale căror elemente se construiesc după o regulă dată.
- Completați șirurile cu elementele care lipsesc:
 - 1, 4, 7, 10, ..., ..., 19, 22, ..., 28, 31;
 - 2, 3, 5, ..., 11, 13, 17, ..., 23, ..., 31, 37, 41, 43, 47, ..., 59.
- Se dă șirul de numere: 1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, Care este valoarea elementului al 44-lea al acestui șir?
 - 6;
 - 9;
 - 8;
 - 10.
- Numărul de bile care formează fiecare pătrat în parte din figura alăturată este un „număr pătratic”. Descoperiți algoritmul de formare a șirului numerelor pătratice și faceți un program care să afișeze al n -lea număr pătratic, unde $1 \leq n \leq 100$.
- Considerăm șirul numerelor piramidale. Termenul n al acestui șir este egal cu suma primelor n pătrate perfecte, începând de la 1. Scrieți un program care generează primii x termeni ai șirului numerelor piramidale, unde x este număr natural, $1 \leq x \leq 50$.

Indicație: $T_1 = 1^2$, $T_2 = 1^2 + 2^2$, $T_3 = 1^2 + 2^2 + 3^2$, ..., $T_x = 1^2 + 2^2 + 3^2 + \dots + x^2$



6 Operații cu șiruri de valori: citire, parcurgere, afișare



Amintiți-vă

Anterior, am învățat ce este un șir de valori, am generat mai multe tipuri de șiruri pe baza unei proprietăți simple a elementelor (pare/impare) sau pe baza anumitor reguli (fiecare termen este egal cu suma ultimilor doi aflați înaintea lui sau un element se generează în funcție de numărul ce reprezintă poziția sa în șir). În general, putem spune că pentru a genera un șir de numere este important să stabilim regula de formare a termenilor și apoi să o aplicăm în calcularea fiecăruia în parte.



Descoperiți

În programare, deseori avem nevoie să lucrăm cu o serie de valori introduse de la tastatură, să le procesăm și să le afișăm. Vom învăța în continuare cum **citim** o secvență de valori de la tastatură, una câte una, cum **parcurgem** secvențial fiecare element, cum **afișăm** elementele introduse, menținând ordinea în care au fost citite.

Scrieți în Code::Blocks programele de la rubricile **A** și **B** propuse mai jos și urmăriți pas cu pas funcționarea lor.

A. Citirea elementelor unui șir de valori numerice



Rețineți

Citirea elementelor unui șir de valori reprezintă procesul prin care utilizatorul introduce, unul câte unul, elementele unei secvențe de valori numerice sau de alt tip, iar acestea sunt preluate și procesate de program. Această citire se poate realiza utilizând diferite tipuri de structuri repetitive, în funcție de tipul problemei, fiecare valoare fiind stocată temporar într-o variabilă și utilizată conform cerințelor problemei.

1. Introducerea unui număr exact de termeni ai unui șir de valori

În cazul în care știți exact câți termeni are șirul de valori, este recomandat ca pentru citirea elementelor sale să utilizați o structură repetitivă cu număr cunoscut de pași. *Important:* introduceți numărul de elemente înainte de a începe citirea valorilor șirului.

Problemă: Introduceți n elemente ale unui șir de valori numerice, numere întregi din intervalul $[-1000, 1000]$, unde n este un număr întreg introdus de la tastatură, $1 \leq n \leq 100$.

Date de intrare: n – număr natural nenul, reprezintă numărul de elemente pe care le are șirul de valori
 val – număr întreg, stochează valoarea unui element al șirului

Date de manevră: i – număr natural nenul, reprezintă termenul șirului vizitat la un moment dat

Indicații:

Algoritmul în pseudocod	Codul în C++
Întreg n, i, val Citește n Pentru $i \leftarrow 1, n$ execută Citește val ...prelucrare val	<pre>int n, i, val; cin>>n; for (i=1; i<=n; i++) { cin>>val; ...prelucrare val }</pre>

2. Introducerea termenilor unui șir de valori până când o condiție este îndeplinită

Atunci când nu știm câți termeni are șirul de valori, vom citi valori de la tastatură până când o condiție va fi îndeplinită. Se va utiliza una dintre structurile repetitive cu număr necunoscut de pași (structura repetitivă condiționată anterior sau cea condiționată posterior) și se vor introduce valori pentru termenii șirului, atât timp cât este îndeplinită o condiție.

Problemă: Introduceți termenii unui șir de valori numerice, numere întregi nenule, atât timp cât nu s-a introdus valoarea 0, șirul având minimum un element.

Indicații: În acest caz, introducerea valorii 0 reprezintă o condiție de oprire.

Structura repetitivă condiționată anterior		Structura repetitivă condiționată posterior	
Algoritmul în pseudocod	Codul C++	Algoritmul în pseudocod	Codul în C++
Întreg val Citește val Cât timp $val \neq 0$ execută Citește val ...prelucrare val	<pre>int val; cin>>val; while (val != 0) { cin>>val; ...prelucrare val ... }</pre>	Întreg val Execută Citește val ...prelucrare val Cât timp $(val \neq 0)$	<pre>int val; do { cin>>val; ...prelucrare val } while (val != 0);</pre>

B. Parcurgerea numerelor dintr-un șir de valori și afișarea acestora

Vom prezenta două operații care se realizează asupra șirurilor de valori: **parcurgerea** șirului și **afișarea** elementelor.

Rețineți

Parcurgerea unui șir de valori reprezintă procesul de vizitare secvențială, pe rând, a fiecărui element în parte, cu scopul de a efectua anumite operații asupra acestuia, precum verificarea unei proprietăți, căutarea, modificarea, numărarea sau, pur și simplu, afișarea sa. Această acțiune se poate realiza utilizând diferite tipuri de structuri repetitive, în funcție de tipul problemei.

 **Problemă:** Se dă un interval $[a, b]$, unde $1 \leq a < b \leq 100$, **a** și **b** numere întregi. Se cere să se afișeze toate numerele întregi din acest interval, despărțite printr-un spațiu.

Exemplu: **Intrare:** 2 10; **Ieșire:** 2 3 4 5 6 7 8 9 10

Indicații: Pentru a rezolva problema, gândim toate numerele întregi cuprinse în intervalul $[a, b]$ ca pe un șir de numere pe care îl parcurgem de la un capăt la altul (de la **a** la **b**) și afișăm fiecare element.

Date de intrare: a, b – numere întregi, limitele intervalului

Date de manevră: i – termenul șirului vizitat și verificat la un moment dat

Date de ieșire: i – termenul afișat al șirului de valori

Algoritmul în pseudocod	Codul în C++
algoritm Afișare_numere Întreg a, b, i Citește a, b Pentru i ← a, b execută Scrie i, ' Sfârșit	<pre>#include <iostream> using namespace std; int main() { int a, b, i; cin>>a>>b; //introducem limitele intervalului for (i=a; i<=b; i++) //parcurgem sirul de valori cout<<i<<" "; //afisam valoarea elementului sirului return 0; }</pre>

```
2 10
2 3 4 5 6 7 8 9 10
Process returned 0 (0x0)   execution time : 4.930 s
Press any key to continue.
```

Aplicați și investigați

- Urmăriți pas cu pas funcționarea programului **B** de mai sus, pentru următorul set de date de intrare: $a=-10$, $b=10$. Câte elemente se afișează?
- Modificați instrucțiunea repetitivă din programul **B** așa cum este prezentat în *imaginea a*. Ce se afișează în acest caz pentru același set de date?
- În programul **B**, modificați pasul din instrucțiunea repetitivă la 2 ($i=i+2$).

Investigați câte elemente ale șirului se afișează acum. Dacă $a=1$ și $b=20$, ce șir particular se afișează?

```
int a, b, i;
cin>>a>>b;
for (i=b; i>=a; i--)
    cout<<i<<endl;
```

Exersați

-  Realizați un program C++ în care se dă șirul numerelor naturale mai mici decât un număr n ($1 \leq n \leq 10000$) și se afișează elementele sale, despărțite prin caracterul @. **Indicații:** **Intrare:** 5; **Ieșire:** 0@1@2@3@4@
- Scrieți un program C++ care să afișeze numerele naturale nenule mai mici decât n ($1 \leq n \leq 10000$), în ordine descrescătoare, despărțite de un spațiu. **Indicații:** **Intrare:** 5; **Ieșire:** 5 4 3 2 1

7 Verificarea unor proprietăți ale elementelor șirului de valori



Amintiți-vă

Amintiți-vă că în lecția anterioară am învățat operațiile de bază cu șirurile de valori: citire, afișare, parcurgere. De reținut este că parcurgerea șirului se realizează întotdeauna cu scopul de a efectua anumite operații asupra acestora.



Descoperiți

Iată două exemple practice în care avem nevoie să citim elementele unui șir de valori și să afișăm doar acei termeni care îndeplinesc o anumită proprietate (condiție):

1. Un magazin are o listă de coduri ale produselor și vrea să afișeze doar codurile care sunt **numere pare**, pentru a aplica o promoție specială asupra produselor aferente.
2. Facem un experiment la biologie: urmărim zilnic cât crește o plantă și avem nevoie să știm care sunt valorile de creștere mai mari decât o valoare dată, pentru a putea face asocieri, de exemplu, cu condițiile externe din acea zi.

Vom descoperi în cele ce urmează cum putem rezolva aceste tipuri de aplicații. Algoritmul este simplu și este prezentat mai jos.



Rețineți

Algoritm pentru afișarea elementelor unui șir de valori cu o anumită proprietate

1. Citim, pe rând, valoarea fiecărui element.
2. Verificăm dacă elementul respectă proprietatea cerută de problemă. Dacă respectă proprietatea, atunci afișăm elementul.

Scrieți în Code::Blocks programele de la rubricile **A** și **B** propuse mai jos și urmăriți pas cu pas funcționarea lor.

A. Afișarea elementelor pare dintr-un șir de valori

Problemă: Se dă un interval $[a, b]$, unde $1 \leq a < b \leq 100$, **a** și **b** numere întregi. Se cere să se afișeze toate numerele pare întregi din acest interval, despărțite printr-un spațiu.

Exemplu: **Intrare:** 3 8; **Ieșire:** 4 6 8

Indicații: Pentru a rezolva problema, gândim toate numerele întregi cuprinse în intervalul $[a, b]$ ca pe un șir de numere pe care îl parcurgem de la un capăt la altul (de la **a** la **b**) și verificăm fiecare element în parte dacă este par sau nu.

Date de intrare: a, b – numere întregi, limitele intervalului

Date de manevră: i – termenul șirului vizitat și verificat la un moment dat

Date de ieșire: i – termenul par al șirului de valori

Algoritmul în pseudocod	Codul în C++
<pre> algoritm Afișare_pare întreg a, b, i Citește a, b Pentru i ← a, b execută Dacă i % 2 = 0 atunci Scrie i, ' ' Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int a, b, i, nr; cin >> a >> b; //introducem limitele intervalului for (i=a; i<=b; i++) //parcurgem sirul de valori { if (i%2==0) //verificam dacă valoarea este para cout << i << " "; //afisam valoarea para } return 0; } </pre>

```

3 8
4 6 8
Process returned 0 (0x0)   execution time : 5.296 s
Press any key to continue.
        
```

B. Afișarea elementelor unui șir de valori, mai mari decât o anumită valoare

Problemă: Se dau două numere naturale n și a ($1 \leq n \leq 100$, $1 \leq a \leq 100$) și un șir de n numere întregi mai mici decât 2000000. Se cere să se afișeze toate valorile acestui șir, mai mari decât a , despărțite printr-un spațiu.

Exemplu: **Intrare:** 4 100 230 45 -1 546 **Ieșire:** 230 546

Indicații: Pentru a rezolva problema, citim valorile lui n și ale lui a , apoi citim elementele șirului și le verificăm unul câte unul pentru a vedea dacă îndeplinesc condiția de a fi mai mari decât valoarea lui a .

Date de intrare: n – număr natural, numărul de elemente ale șirului
 a – număr natural, limita peste care afișăm elementele
 val – număr întreg, valoarea unui element al șirului

Date de manevră: i – termenul șirului vizitat și verificat la un moment dat

Date de ieșire: val – număr întreg, termenul mai mare decât a al șirului de valori



Algoritmul în pseudocod	Codul în C++
<pre> algoritm Afișare_Nr_Mari întreg n, a, i, val Citește n, a Pentru i ← 1, n execută Citește val Dacă val > a atunci Scrie val, ' ' Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, a, i, val; cin >> n >> a; //introducem valorile lui a si b for (i=1; i<=n; i++) //parcurgem sirul de valori { cin >> val; //introducem valoarea unui element if (val > a) //verificam dacă valoarea este mai mare cout << val << " "; //daca da, afisam valoarea } return 0; } </pre>

```

4 200
230 45 -1 546
230 546
Process returned 0 (0x0)   execution time : 20.557 s
Press any key to continue.

```



Aplicații și investigații

- Urmăriți pas cu pas funcționarea programului **A** de mai sus, pentru următorul set de date de intrare: $a=32$, $b=41$. Câte elemente se afișează?
- Priviți codul din *imaginea a* și investigați dacă este similar cu cel al programului **A**.
- Urmăriți pas cu pas funcționarea programului **B**, pentru următorul set de date de intrare: $n=4$, $a=20$, șirul de numere: 10 34 22 25. Câte elemente se afișează în acest caz?
- Modificați programul **B**, astfel încât să fie afișate elementele cu valori mai mici decât a .

```

int a, b, i, nr;
cin >> a >> b;
i=a;
while (i<=b)
{
  if (i%2==0)
    cout << i << " ";
  i++;
}

```



Exersați



- Scrieți un program C++ în care introduceți două numere naturale n și k , $1 \leq n \leq 20$, $1 \leq k \leq 9$. Introduceți apoi n termeni ai unui șir de numere naturale, mai mici decât 2000. Afișați valorile termenilor șirului care au ultima cifră mai mare decât k .
Indicații: **Intrare:** 4 2 14 22 1 33 **Ieșire:** 14 33
- Scrieți un program C++ care citește un număr n , $1 \leq n \leq 20$, apoi un șir de n numere întregi din intervalul $[-1000, 1000]$. Afișați doar numerele divizibile cu 3 introduse.
Indicații: **Intrare:** 3 22 6 33 **Ieșire:** 6 33
- Scrieți un program C++ în care se introduc termenii unui șir de numere naturale nenule, până la apariția lui 0. Toți termenii sunt mai mici decât 2000000. Afișați termenii care au ultima cifră egală cu 2.
Indicații: **Intrare:** 2 6 33 422 0 **Ieșire:** 2 422

8 Numărarea elementelor care îndeplinesc o condiție



Amintiți-vă

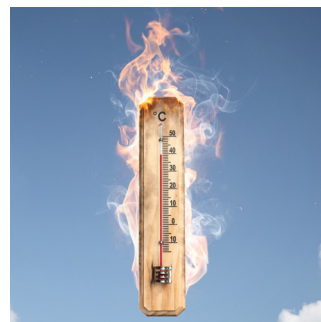
Până acum am învățat să generăm un șir de valori sau să îi citim elementele unul câte unul de la tastatură, să îl parcurgem și să verificăm anumite proprietăți ale elementelor, să afișăm elementele unui șir, fie în totalitate, fie parțial, pe cele care îndeplineau anume criterii. Vom învăța mai departe să și numărăm aceste elemente.



Descoperiți

Prezentăm două exemple practice în care este nevoie să numărăm elementele unui șir de valori care au o anumită proprietate:

1. Un meteorolog analizează temperaturile zilnice dintr-o lună și vrea să determine câte zile au fost caniculare (au avut temperatura **peste 30°C**).
2. Un manager de supermarket vrea să afle câte produse dintr-o listă au un preț rotund (adică un preț care este **divizibil cu 10**, cum ar fi: 20 lei, 50 lei, 100 lei). Acest lucru îl ajută să decidă asupra promoțiilor speciale pentru aceste produse.



Rețineți

Algoritm pentru numărarea elementelor unui șir de valori cu o anumită proprietate

1. Utilizăm o variabilă de tip „contor”, variabilă utilizată pentru a număra elementele, pe care o inițializăm cu 0. În această variabilă vom stoca numărul elementelor care îndeplinesc proprietatea dată în cerințe.
2. Citim, pe rând, valoarea fiecărui element.
3. Verificăm dacă elementul respectă proprietatea cerută de problemă. Dacă respectă proprietatea, atunci creștem valoarea variabilei „contor” cu 1 unitate (numărăm elementul).
4. După ce am terminat de parcurs tot șirul de valori, afișăm valoarea lui **cnt**.

Aplicăm algoritmul în cele ce urmează, pentru cele două exemple de mai sus. Scrieți în Code::Blocks programele de la rubricile **A** și **B** propuse mai jos și urmăriți pas cu pas funcționarea lor.

A. Analiza temperaturilor

Problemă: Se citește de la tastatură șirul valorilor temperaturilor zilnice dintr-un interval de timp de **z** zile ($1 \leq z \leq 31$), în grade Celsius. Să se afișeze câte dintre aceste temperaturi sunt caniculare (mai mari decât 30°C).

Exemplu: Intrare: 3 24 31.5 34 Ieșire: 2

Indicații: Pentru a rezolva problema, aplicăm algoritmul de mai sus.

Date de intrare: z – număr întreg, numărul de zile pe care facem analiza

temp – număr real, reprezintă temperatura în grade Celsius dintr-o zi

Date de manevră: i – număr natural, contor (pentru a câta zi introducem temperatura)

Date de ieșire: cnt – variabilă contor, număr întreg, reprezintă câte zile au fost caniculare

Algoritmul în pseudocod	Codul în C++
<pre> algoritm Afișare_NrZile întreg z, i, cnt real temp cnt ← 0 Citește z Pentru i ← 1, z execută Citește temp Dacă temp > 30.0 atunci cnt ← cnt + 1 Scrie cnt Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int z, i, cnt; float temp; cnt=0; //initializam contorul cu 0 cin>>z; //introducem numarul de zile for (i=1; i<=z; i++) //parcurgem sirul de valori { cin>>temp; if (temp>30.0) //verificam daca valoarea este > 30 cnt++; //crestem contorul cu o unitate } cout<<cnt; return 0; } </pre>

```

3
24 31.5 34
2
Process returned 0 (0x0)   execution time : 13.888 s
Press any key to continue.
        
```

B. Numărarea produselor într-un magazin

Problemă: La un magazin avem n produse ($1 \leq n \leq 30$). Managerul magazinului introduce lista prețurilor acestora, numere naturale, și, pentru a putea lua o decizie asupra promoțiilor pentru aceste produse, dorește să afle câte produse au un preț rotund (divizibil cu 10, cum ar fi 20 lei, 50 lei). Ajutați-l pe manager și afișați pe ecran această informație.

Exemplu: **Intrare:** 6 15 10 20 23 11 70 **Ieșire:** 3

Indicații:

Date de intrare: n – număr natural, numărul de produse din magazin

pret – număr real, reprezintă prețul unui produs

Date de manevră: i – număr natural, contor (pentru al câtelea preț introducem)

Date de ieșire: cnt – variabilă contor, număr întreg, reprezintă câte produse au preț rotund



Algoritmul în pseudocod	Codul în C++
<pre> algoritm Afișare_NrProduse întreg n, i, cnt, pret cnt ← 0 Citește n Pentru i ← 1, n execută Citește pret Dacă pret%10=0 atunci cnt ← cnt+1 Scrie cnt Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i, cnt, pret; cnt=0; //initializam contorul cu 0 cin>>n; //introducem numarul de produse for (i=1; i<=n; i++) //parcurgem sirul de valori { cin>>pret; if (pret%10==0) //verificam daca pretul este div. cu 10 cnt++; //crestem contorul cu o unitate } cout<<cnt; //afisam valoarea return 0; } </pre>

```

6
15 10 20 23 11 70
3
Process returned 0 (0x0)   execution time : 2.577 s
Press any key to continue.

```



Aplicații și investigații

- Urmăriți pas cu pas funcționarea programului **A** de mai sus, pentru următorul set de date de intrare: $z=4$, temperaturile: 23 17.5 24 29. Ce se afișează?
- Urmăriți pas cu pas funcționarea programului **B** de mai sus, pentru următorul set de date de intrare: $n=3$, șirul prețurilor: 34 20 45. Ce se afișează?



Exersați

- Într-o clasă sunt n elevi ($1 \leq n \leq 30$). Profesorul introduce șirul notelor elevilor la un test. Pentru a evalua performanța clasei sale, el analizează apoi toate rezultatele. Vrea să afle câți elevi au obținut o notă mai mare sau egală cu 8. Realizați un program C++ pentru a-l ajuta pe profesor.

Indicații: **Intrare:** 6 10 9 9 6 8 7 **Ieșire:** 4



- Scrieți un program C++ care rezolvă problema: Se introduce de la tastatură un șir de valori întregi nenule, până când se introduce valoarea 0. Afișați câte elemente are șirul de valori.

Indicații: Se aplică algoritmul de numărare învățat, dar fără a se mai verifica vreo condiție anume. Valoarea variabilei contor va crește la fiecare nouă citire a unui element. Valoarea 0 nu se numără.

Intrare: 10 9 3 4 5 0 **Ieșire:** 5



Portofoliu



Se introduce un șir de n numere naturale mai mici decât 1000. Afișați câte dintre ele au ultima cifră mai mică decât 6.

Intrare: 3 29 45 31 **Ieșire:** 2

9 Verificarea îndeplinirii unor proprietăți de către toate elementele unui șir



Amintiți-vă

Am învățat cum se verifică dacă un anumit element al unui șir îndeplinește anumite proprietăți, am învățat să numărăm elementele care îndeplinesc proprietățile respective.

Vom vedea în continuare cum putem utiliza aceste lucruri deja parcurse pentru a verifica dacă **toate** elementele unui șir de valori îndeplinesc anumite condiții.

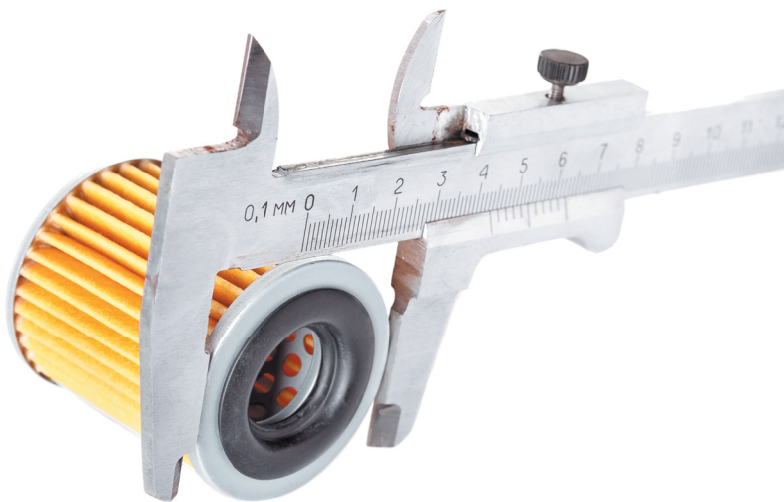


Descoperiți

Verificarea existenței unor elemente cu anumite proprietăți are aplicabilitate practică imediată. Să vedem câteva exemple:

1. Într-o fabrică de piese auto, fiecare piesă trebuie să respecte anumite dimensiuni (de exemplu, toate filtrele trebuie să aibă un diametru între 2,5 cm și 3 cm). Toate măsurătorile formează un șir de numere și se verifică dacă toate măsurătorile respectă standardul impus.
2. Într-un test de admitere, toți candidații trebuie să aibă un scor de minimum 70 de puncte pentru a fi eligibili pentru următoarea etapă. Se verifică dacă toate notele dintr-un șir îndeplinesc această condiție.
3. Într-un set de analize medicale, trebuie să se verifice dacă toate valorile unor parametri (glicemie, colesterol etc.) se încadrează în intervalul normal pentru un pacient sănătos.

Există două metode pe care le putem aplica pentru a verifica dacă toate elementele unui șir de valori îndeplinesc anumite proprietăți.



Rețineți

Metoda 1 – prin numărarea elementelor care îndeplinesc proprietățile date

1. Utilizăm o variabilă de tip „contor”, pe care o inițializăm cu 0. În această variabilă vom stoca numărul elementelor care îndeplinesc proprietățile date în cerințe.
2. Citim fiecare valoare, pe rând, și o verificăm în ceea ce privește proprietățile date. Dacă valoarea îndeplinește acele proprietăți, valoarea variabilei „contor” va crește cu 1.
3. După ce am terminat de parcurs tot șirul de valori, verificăm dacă valoarea variabilei „contor” este egală cu numărul de elemente al șirului de valori. Dacă da, înseamnă că toate elementele șirului îndeplinesc proprietățile cerute.

Observație: Dacă nu știm câte elemente are șirul de valori, putem utiliza algoritmul de numărare din lecția precedentă pentru a le număra sau orice alt algoritm care este potrivit.

Metoda 2 – cu utilizarea unei variabile tip „semafor” (care poate avea doar două valori posibile)

1. Utilizăm o variabilă de tip „semafor”, care are două valori posibile:
 - 1 = toate elementele îndeplinesc proprietățile
 - 0 = există cel puțin un element care nu îndeplinește proprietățile
 Inițializăm cu 1 această variabilă (presupunem că toate elementele îndeplinesc proprietățile).
2. Citim fiecare valoare, pe rând, și o verificăm în ceea ce privește proprietățile date. Dacă valoarea nu îndeplinește acele proprietăți, valoarea variabilei „semafor” va fi modificată la 0.
3. După ce am verificat toate valorile, în funcție de valoarea variabilei „semafor” afișăm un mesaj corespunzător.

În cele ce urmează, vom aplica ambele metode în două exemple de programe. Scrieți în Code::Blocks programele de la rubricile **A** și **B** propuse mai jos și urmăriți pas cu pas funcționarea lor.



A. Metoda 1

Problemă: Se citesc valorile unui șir de n elemente ($1 \leq n \leq 30$), conținând notele elevilor unei clase la o etapă de concurs. Notele sunt reprezentate prin numere întregi. Pentru a trece la faza următoare, elevii trebuie să obțină cel puțin nota 8. Afișați „DA”, dacă toți elevii se califică, și „NU”, altfel.

Exemplu: Dacă avem 6 elevi, iar notele lor sunt 8 9 10 10 10 9, atunci se va afișa „DA”.

Dacă avem 6 elevi, iar notele lor sunt 8 9 7 10 10 9, atunci se va afișa „NU”.

Indicații:

Date de intrare: n – număr natural, numărul de note

$nota$ – număr natural, reprezintă nota

Date de manevră: i – număr natural, contor (a câta notă este introdusă)

cnt – variabila contor, număr întreg, reprezintă numărul de note peste 8

Algoritmul în pseudocod	Codul în C++
<pre> algoritm VerificareNote întreg $n, i, cnt, nota$ $cnt \leftarrow 0$ Citește n Pentru $i \leftarrow 1, n$ execută Citește $nota$ Dacă $nota \geq 8$ atunci $cnt \leftarrow cnt + 1$ Dacă $cnt = n$ atunci Scrie 'DA' Altfel Scrie 'NU' Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i, cnt, nota; cnt=0; //inițial avem 0 note peste 8 cin>>n; //introducem numărul de note for (i=1; i<=n; i++) { //parcurgem șirul de valori cin>>nota; //citim nota if (nota>=8) //verificam daca indeplinește condiția cnt++; //numărul de note peste 8 crește cu 1 } if (cnt==n) //verif. dacă n elemente îndeplinesc condiția cout<<"DA"; else cout<<"NU"; return 0; } </pre>

```

6
8 9 10 10 10 9
DA
Process returned 0 (0x0)   execution time : 3.855 s
Press any key to continue.

```

Observație: Dacă în urma numărării am obținut un număr strict pozitiv (mai mare decât 0), atunci înseamnă că există în șirul de elemente valori care îndeplinesc proprietățile respective.

B. Metoda 2

Problemă: Se citesc valorile unui șir de n elemente ($1 \leq n \leq 30$), numere reale cuprinse între 1 și 10, reprezentând diametrul unor piese auto, măsurat pentru verificare. Pentru a trece controlul de calitate, piesele trebuie să aibă diametrul între 2,5 cm și 3 cm). Afișați „DA”, dacă toate măsurătorile respectă standardul impus, și „NU”, altfel.

Exemplu: Dacă avem 4 piese, iar diametrele acestora au dimensiunile 2.6 2.5 2.7 3, atunci se va afișa „DA”.

Dacă avem 4 piese, iar diametrele acestora au dimensiunile 2.6 2.5 5 3, atunci se va afișa „NU”.

Indicații:

Date de intrare: n – număr natural, numărul de piese

d – număr real, reprezintă măsurătoarea

Date de manevră: ok – variabila semafor, număr întreg, arată dacă toate măsurătorile respectă standardul

i – număr natural, contor (a câta măsurătoare este introdusă)

Vom aplica algoritmul de la metoda 2. Citim fiecare valoare, pe rând, și o verificăm în ceea ce privește proprietățile date. Dacă valoarea nu îndeplinește acele proprietăți, valoarea variabilei „semafor” va fi modificată la 0.

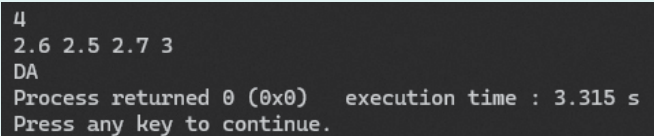
Explicăm acum cum anume construim condiția care va trebui să fie verificată pentru fiecare număr în parte. Pentru a respecta standardul, diametrul unei piese trebuie să fie între 2,5 cm și 3 cm. Condiția C++ este următoarea:

$$d \geq 2.5 \ \&\& \ d \leq 3$$

Înseamnă că, pentru a verifica faptul că o piesă nu respectă standardul, aplicăm negația asupra acestei expresii, amintindu-ne din clasele anterioare cum funcționează aceasta:

dacă P și Q sunt două expresii logice, atunci $Nu(P \ \&\& \ Q)$ este echivalent cu $Nu(P)$ sau $Nu(Q)$

Astfel, condiția noastră va fi: $!(d \geq 2.5 \ \&\& \ d \leq 3) = !(d \geq 2.5) \ || \ !(d \leq 3) = (d < 2.5 \ || \ d > 3)$

Algoritmul în pseudocod	Codul în C++
 <pre> algoritm VerificareDimensiuni întreg n, i, ok real d ok ← 1 Citește n Pentru i ← 1, n execută Citește d Dacă d < 2.5 sau d > 3 atunci ok ← 0 Dacă ok = 1 atunci Scrie 'DA' Altfel Scrie 'NU' Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i, ok; float d; ok=1; //presupunem ca toate piesele corespund cin>>n; //introducem numărul de piese for (i=1; i<=n; i++) //parcurgem șirul de valori { cin>>d; //citim masuratoarea if (d<2.5 d>3) //daca dimensiunea nu este conforma ok=0; //indicăm faptul că nu toate piesele corespund } if (ok==1) //verif. daca toate dimensiunile sunt conforme cout<<"DA"; else cout<<"NU"; return 0; } </pre> 



Aplicați și investigați

- Urmăriți pas cu pas funcționarea programelor **A** și **B** de mai sus, pentru același set de date de intrare: $n=4$ și șirul de valori 9 9 10 8. Ce se afișează?
- Investigați ce s-ar întâmpla dacă în programul **B**, la verificarea proprietății elementelor, am folosi chiar condiția din enunț ($d \geq 2.5 \ \&\& \ d \leq 3$), fără să aplicăm negația.



Exersați



- Alegeți condiția care trebuie completată în bucata de cod din *imaginea a*, astfel încât programul să afișeze dacă toate valorile unui șir de n numere întregi sunt pare.

a. $\text{numar} \% 2 \neq 0$ **b.** $\text{numar} / 2 \neq 0$ **c.** $\text{numar} \% 10 \neq 0$
- Se citesc n numere întregi nenule ($1 \leq n \leq 30$), cu valori din intervalul $[-200, 200]$. Realizați un program C++ care afișează: „Toate pozitive”, dacă toate numerele introduse sunt pozitive, „X dintre ele pozitive”, dacă doar X sunt pozitive ($x > 1$), „Unul pozitiv”, dacă avem un singur număr pozitiv sau „Niciunul pozitiv”, dacă toate sunt negative.

Indicații: Folosiți o variabilă contor care să numere câte dintre numere sunt pozitive (metoda 1). Ca exemple de execuție:

Intrare: $n=4$, șirul de numere: 2 4 6 7 **Ieșire:** „Toate pozitive”
Intrare: $n=4$, șirul de numere: 2 -4 6 -7 **Ieșire:** „2 dintre ele pozitive”
Intrare: $n=4$, șirul de numere: 2 -4 -6 -7 **Ieșire:** „Unul pozitiv”
Intrare: $n=4$, șirul de numere: -2 -4 -6 -7 **Ieșire:** „Niciunul pozitiv”

a

```

int n, numar, toatePare = 1;
for (int i = 0; i < n; i++)
{ cin >> numar;
  if (...)
    toatePare = 0;
}
if (toatePare==1)
  cout << "Toate pare";
else
  cout << "Nu toate pare";

```

Portofoliu



Se citesc n numere reale, $1 \leq n \leq 30$. Verificați dacă **toate se află în intervalul [10, 100]**.

Folosiți mesaje de eroare sugestive, la alegere.

Indicații: Exemple de mesaje: dacă n este 3 și șirul de numere este 7.5 12 18, atunci se poate afișa „Da”, iar dacă șirul este 1 12 34, se poate afișa „Nu”.

Pentru ca un element x să aparțină intervalului $[10, 100]$, el trebuie să îndeplinească următoarea condiție: $x \geq 10 \ \&\& \ x \leq 100$. Astfel, condiția ca acesta să nu aparțină acestui interval este: $!(x \geq 10 \ \&\& \ x \leq 100) = (x < 10 \ || \ x > 100)$.



10 Căutarea unui element în șir de valori. Verificarea existenței unui anume element



Amintiți-vă

În lecția anterioară am învățat cum anume putem citi, parcurge și afișa elementele unui șir de valori. În particular, am parcurs un interval $[a, b]$, unde $1 \leq a, b \leq 100$, a și b numere întregi, și am afișat toate numerele pare întregi din acest interval. În fond, această operație reprezenta o acțiune de căutare a elementelor pare din acel șir de numere. Vom detalia în cele ce urmează această acțiune de căutare a unui anume element al șirului.



Descoperiți

Căutarea unui element într-o listă este, de fapt, o particularizare a acțiunii de verificare a unei proprietăți a elementelor și are multe utilizări în viața reală. Iată două exemple:

1. Un paznic verifică într-o listă de coduri dacă un vizitator are un cod de acces valid pentru a intra într-o clădire.
2. Avem o listă cu numerele de pe tricourile unor sportivi participanți la un concurs, lista reprezentând, de fapt, ordinea de intrare în concurs. Trebuie să găsim un concurent căutând codul său și să vedem al câtelea va intra în competiție.

În astfel de cazuri, aplicăm căutarea secvențială, cu numărarea elementelor, dacă este cazul.



Rețineți

Căutarea unui element care îndeplinește o condiție într-un șir de valori reprezintă procesul de parcurgere a șirului pentru a identifica și extrage primul element sau toate elementele care respectă un criteriu prestabilit.

Algoritm de căutare secvențială

1. Folosim o variabilă tip „semafor” care are două valori posibile: 1 = am găsit numărul, 0 = nu am găsit numărul și o inițializăm cu 0 (inițial, nu am găsit încă elementul).
2. Citim fiecare valoare, pe rând, și verificăm dacă este egală cu numărul căutat. Dacă avem egalitate, înseamnă că am găsit numărul, variabila „semafor” va primi valoarea 1.
3. După ce am verificat toate valorile, în funcție de valoarea variabilei „semafor”, afișăm un mesaj corespunzător.

Căutarea secvențială este o metodă de a arăta existența unor elemente într-un șir de valori care îndeplinesc anumite proprietăți. **Dacă găsim minimum un element în șir care are proprietatea solicitată, înseamnă, evident, că există în șir elemente cu acea proprietate.**

Scrieți în Code::Blocks programul propus mai jos pentru căutarea secvențială a unui element cu anume valoare și urmăriți pas cu pas funcționarea lui.

Căutarea secvențială

Problemă: Se citesc două numere întregi, n și X ($1 \leq n \leq 30$, $-10000 \leq X \leq 10000$). Se citește apoi un șir de n numere întregi distincte, din intervalul $[-10000, 10000]$. Se cere să se afișeze „DA”, dacă numărul întreg X se regăsește printre elementele șirului, și „NU”, altfel.

Exemplu: Intrare: 3 8 10 8 13 Ieșire: DA

Intrare: 3 8 10 18 13 Ieșire: NU

Indicații:

Date de intrare:

n – număr întreg, numărul de elemente ale șirului

X – număr întreg, numărul căutat

nr – număr întreg, valoarea termenului șirului

Date de manevră:

gasit – număr întreg, variabilă semafor cu două valori posibile,

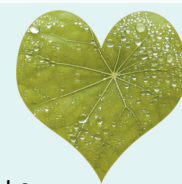
arată dacă am găsit sau nu numărul în șir

i – termenul șirului vizitat și verificat la un moment dat





Algoritmul în pseudocod	Codul în C++
<pre> algoritm Căutare_secvențială întreg n, X, i, nr, gasit gasit ← 0 Citește n, X Pentru i ← 1, n execută Citește nr Dacă nr=X atunci gasit ← 1 Dacă gasit = 0 atunci Scrie 'NU' Altfel Scrie 'DA' Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, X, i, nr, gasit; gasit=0; cin>>n>>X; //introducem numarul de elemente for (i=1; i<=n; i++) //parcurgem sirul de valori { cin>>nr; //introducem valoarea termenului sirului if (nr==X) //verificam daca am gasit numarul gasit=1; //marcam faptul ca am gasit numarul } if (gasit==0) //daca nu am gasit numarul cout<<"NU"; else cout<<"DA"; return 0; } </pre> 3 8 10 8 13 DA Process returned 0 (0x0) execution time : 23.734 s Press any key to continue.



Ca observație la programul de mai sus, putem spune că dacă găsim în șir un termen cu valoarea egală cu X, este similar cu a spune că există în șir termeni care au valoarea X.



Aplicații și investigații

- Urmăriți pas cu pas funcționarea programului de mai sus, pentru următorul set de date de intrare: $n=4$, $X=20$ și șirul de valori: -2 45 20 21. Ce se afișează?
- Modificați condiția de oprire a instrucțiunii *for*, din $i \leq n$ în $i < n \ \&\& \ gasit == 0$. Investigați și aflați care este diferența în execuția programului, față de versiunea anterioară.
- Modificați programul de mai sus, astfel încât, dacă elementul căutat este găsit în șir, să afișați după „DA” și pe a câta poziție se află acesta (utilizați variabila i). Ca exemplu, pentru valorile de intrare 3 8 pentru n și X , dacă șirul este 10 8 13, atunci se va afișa „DA 2”, iar dacă șirul este 10 18 13, atunci se va afișa „NU”.



Exersați

- Priviți codul din *imaginea a*. Alegeți cu ce ar trebui înlocuite ... pentru a afișa dacă există un element care are cifra zecilor egală cu 7, într-un șir cu n elemente, numere naturale de minimum două cifre, mai mici decât 200000.

a. if ($n \% 10 \% 10 == 7$)
gasit=0;

b. if ($nr \% 10 \% 10 == 7$)
gasit=1;

c. if ($nr \% 10 / 10 == 7$)
gasit=1;
- Să se verifice dacă într-un șir de n numere întregi pozitive, unde $1 \leq n \leq 100$, există numere pare.
Indicații: se va căuta în șir un număr par; dacă se găsește, atunci înseamnă că există numere pare în șir.
Exemplu: Intrare: $n=4$ 2 3 1 6 *Ieșire:* Da; *Intrare:* $n=4$ 7 3 1 5 *Ieșire:* Nu
- Priviți codul de mai jos. Scrieți în caiete ce afișează acesta dacă se introduc următoarele valori la intrare: $n=5$, iar șirul: 13 21 6 8 41.

a

```

int n, i, nr, gasit=0;
cin>>n;
for (i=1; i<=n; i++)
{
  cin>>nr;
  ...
}
if (gasit==0) cout<<"NU";
else cout<<"DA";

```



```

int n, i, nr, poz=0;
cin>>n;
for (int i=1; i<=n; i++)
{
  cin >> nr;
  if (nr%2==0 && poz==0)
    poz=i;
}
if (poz!=0) cout<<poz;
else cout<<"Nu exista";

```

Portofoliu



Se dă un șir de n numere naturale, mai mici decât 100000, $1 \leq n \leq 30$. Afișați toate pozițiile pe care apar numere impare, despărțite printr-un spațiu, știind că primul element se află pe poziția 1. Dacă nu există niciun număr par, să se afișeze „Nu există”.

Indicații: Intrare: 4 1 2 6 9

Ieșire: 1 4



11 Minimul și maximul dintr-un șir de valori



Descoperiți

Aflarea minimului sau maximului unui șir de numere este un caz particular al afișării elementelor care îndeplinesc condiția dată, și anume valoarea elementului să fie cea mai mare a șirului. Se va aplica algoritmul de mai sus pentru calcularea maximului.



Rețineți

Algoritmul pentru afișarea maximului/minimului unui șir de valori

Se consideră o variabilă pe care o numim **max**, care va conține la fiecare pas valoarea maximă a elementelor șirului analizate până la acel moment, variabilă de ieșire.

1. Se citesc elementele șirului.
2. Se parcurge șirul și se verifică, pentru fiecare element în parte, dacă valoarea acestuia este mai mare decât valoarea variabilei **max** (aceasta reprezintă maximul depistat până la acel moment). Dacă da, atunci noul maxim este chiar elementul analizat și, astfel, **max** primește valoarea sa.
3. Se afișează valoarea variabilei **max** care conține maximul șirului.

Observație: Variabila **max** se inițializează cu o valoare, astfel încât aceasta să nu influențeze rezultatul. Această valoare poate fi: 1) o valoare mai mică decât orice element; 2) valoarea elementului de la care se începe parcurgerea șirului pentru verificare.

Algoritmul pentru determinarea minimului unui șir de numere este similar. Diferă variabila de ieșire, care de această dată va reține minimul (numită **min**) și condiția verificată pentru fiecare element (dacă valoarea elementului este mai mică decât valoarea variabilei **min**, atunci noul minim este chiar elementul respectiv).

A. Maximul unui șir de valori

► *Problemă:* Aflați maximul dintr-un șir de **n** numere naturale pozitive, introduse de la tastatură, mai mici decât 1000000, unde $1 \leq n \leq 20$.

Exemplu: **Intrare:** n=5 21 54 3 23 5 **Ieșire:** Maximul=54

Indicații: În cadrul acestui exercițiu trebuie să afișăm o singură valoare, valoarea maximă a elementelor șirului. Este un caz particular al afișării elementelor care îndeplinesc condiția dată, și anume valoarea elementului să fie cea mai mare a șirului. Se va aplica algoritmul de mai sus pentru calcularea maximului.



Algoritmul în pseudocod	Codul în C++
<pre> algoritm Maxim întreg n, nr, i, max max ← 0 Scrie 'n=' Citește n Pentru i ← 1, n execută Citește nr Dacă nr > max atunci max ← nr Scrie 'Maximul=', max Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, nr, i, max; max=0; cout<<"n="; cin>>n; //se citește numărul de elemente for (i=1; i<=n; i++) //se parcurge șirul de numere { cin>>nr; //se citește valoarea unui element if (nr>max) //se verifică valoarea nouă cu maximul actual max=nr; //noul număr este mai mare, el devine noul maxim } cout<<"Maximul="<<max; return 0; } </pre> <pre> n=5 21 54 3 23 5 Maximul=54 Process returned 0 (0x0) execution time : 9.211 s Press any key to continue. </pre>

Vom urmări un exemplu pentru funcționarea structurii repetitive cu număr cunoscut de pași din program, pentru valorile de intrare n=2 și șirul de numere 11 2. Valoarea variabilei **max** este inițializată cu 0.

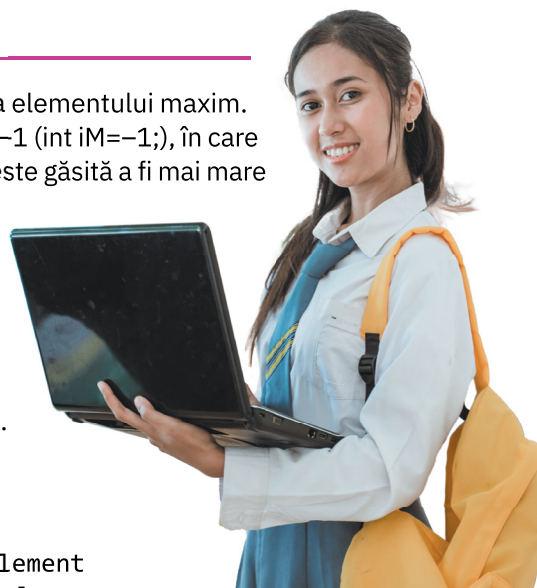
	n	i	nr	max	Valoarea de adevăr a expresiei (nr>max)	Acțiune
for (i=1; i<=n; i++)	2	1		0		Intrăm în instrucțiunea repetitivă (i<n este Adevărat)
{ cin>>nr;			11			Citim valoarea unui nou termen al șirului
if (nr>max)					Adevărat	Noua valoare este mai mare decât maximul de până atunci
max=nr;				11		Maximul devine chiar noua valoare
}						Reiterăm instrucțiunea repetitivă
for (i=1; i<=n; i++)	2	2				Reintrăm în structura repetitivă (i<n este Adevărat)
{ cin>>nr;			2			Citim valoarea unui termen nou al șirului
if (nr>max)					Fals	Noua valoare nu este mai mare decât maximul de până atunci, așadar nu facem nimic, ieșim din instrucțiunea decizională
max=nr;						Nu se execută
}						Reiterăm instrucțiunea repetitivă
for (i=1; i<=n; i++)	2	3				Ieșim din instrucțiunea repetitivă (i<n este Fals)
{ cin>>nr;						
if (nr>max)						
max=nr;						
}						



Aplicații și investigații

- Modificați programul de mai sus în așa fel încât să se afișeze poziția elementului maxim. Pentru aceasta, declarați o variabilă nouă întreagă **im**, inițializată cu -1 (int im=-1;), în care rețineți de fiecare dată poziția **i** a elementului șirului a cărei valoare este găsită a fi mai mare decât valoarea lui **max** (când nr>max, executați și im=i). Rulați programul în mod **Depanare** (*Debug*) și urmăriți ce se întâmplă cu valoarea variabilei **im**.
- Modificați programul anterior astfel încât variabila **max** să fie inițializată cu primul element al șirului. Observați că, în acest caz, numărul de verificări *dacă* (nr>max) este mai mic cu 1. Investigați care sunt diferențele dintre cele două tipuri de inițializare a variabilei **max**. Secvența de verificare este cea prezentată mai jos:

```
cout<<"n="; cin>>n;
cin>>nr;          //se citeste primul element
max=nr;           //initializam max cu valoarea primului element
for (i=2; i<=n; i++) //se parcurge in continuare sirul
{
    cin>>nr;
    if (nr>max)
        max=nr;
}
```



Descoperiți

B. Minimul unui șir de valori

Problemă: Aflați minimul dintr-un șir de **n** numere naturale pozitive introduse de la tastatură, mai mici decât 1000000, unde $1 \leq n \leq 20$.

Exemplu: **Intrare:** n=5 21 54 3 23 5; **Ieșire:** Minimul=3

Indicații: În cadrul acestui exercițiu trebuie să afișăm o singură valoare, valoarea minimă a elementelor șirului.



Algoritmul în pseudocod	Codul în C++
<pre> algoritm Minim întreg n, nr, i, min min ← 1000000 Scrie 'n=' Citește n Pentru i ← 1, n execută Citește nr Dacă nr < min atunci min ← nr Scrie 'Minimul=', min Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, nr, i, min; min=1000000; cout<<"n="; cin>>n; //se citește numărul de elemente for (i=1; i<=n; i++) //se parcurge șirul de numere { cin>>nr; //se citește valoarea unui element if (nr<min) //se verifică valoarea nouă cu minimul actual min=nr; //noul număr este mai mic, el devine noul minim } cout<<"Minimul="<<min; return 0; } </pre>

```

n=5
21 54 3 23 5
Minimul=3
Process returned 0 (0x0)   execution time : 3.437 s
Press any key to continue.

```



Aplicații și investigații

- Modificați programul de mai sus în așa fel încât să se afișeze poziția elementului minim. Pentru aceasta, declarați o variabilă nouă întreagă **im**, inițializată cu -1 (int im=-1;), în care rețineți de fiecare dată poziția **i** a elementului șirului a cărui valoare este găsită a fi mai mică decât valoarea lui **min** (când nr<min, executați și im=i). Rulați programul în mod **Depanare** (Debug) și urmăriți ce se întâmplă cu valoarea variabilei **im**.
- Modificați programul anterior astfel încât variabila **min** să fie inițializată cu primul element al șirului. Observați că, în acest caz, numărul de verificări *dacă* (nr<min) este mai mic cu 1. Investigați care sunt diferențele dintre cele două tipuri de inițializare a variabilei **min**. Secvența de verificare este cea de mai jos:

```

cout<<"n="; cin>>n;
cin>>nr; //se citește primul element
min=nr; //initializam min cu valoarea primului element
for (i=2; i<=n; i++) //se parcurge în continuare șirul
{
  cin>>nr;
  if (nr<min)
    min=nr;
}

```



Exersați

- Realizați un program C++ care să afișeze diferența dintre maximul și minimul unui șir de **n** ($1 \leq n \leq 100$) numere reale din intervalul $[-200, 200]$.
Indicații: Intrare: n=5 7 15 9 6 11; **Ieșire:** 9
- Realizați un program C++ care să afișeze care este cel mai mic număr par al unui șir de **n** ($1 \leq n \leq 100$) numere reale din intervalul $[-200, 200]$.
Indicații: Intrare: n=5 8 14 9 6 11; **Ieșire:** 6
- Realizați un program C++ care să afișeze care este cel mai mare număr impar al unui șir de **n** ($1 \leq n \leq 100$) numere reale din intervalul $[-200, 200]$.
Indicații: Intrare: n=5 8 14 9 6 11; **Ieșire:** 11
- O stație meteo înregistrează temperaturile din fiecare zi timp de 5 zile, numere reale. Scrieți un program C++ care să afișeze temperatura cea mai scăzută și cea mai ridicată din săptămână.
Indicații: Intrare: 8 14.5 10 9.5 11; **Ieșire:** Minima=8 Maxima=14.5
- Lucru în echipă.** Scrieți în caiete câte o problemă care să se rezolve cu ajutorul unuia dintre algoritmii învățați pentru calcularea minimului/maximului dintr-un șir de valori. Schimbați apoi între voi problemele și rezolvați-le în C++.



Portofoliu



Marcel este un om foarte calculat. În decursul unui an, el știe care a fost totalul cheltuielilor sale în fiecare lună în parte. În nicio lună nu a cheltuit mai mult de 1 500 RON. Ajuțați-l să facă un program care să îi afișeze luna în care a cheltuit cel mai puțin.

Exemplu:

Intrare: 1234 1432 1222
1001 978 987 1223 1455 1500
1378 1226 1190;
Ieșire: Luna mai

12 Suma și produsul elementelor unui șir de valori



Descoperiți

Gândiți-vă la situații în care trebuie să calculați suma și produsul unui număr mare de elemente ale unui șir de valori, introduse succesiv:

1. gestionarea bugetului: calculul sumei cheltuielilor dintr-o lună, știind cât s-a cheltuit în fiecare zi.
2. aflarea unei valori medii (calculul mediei aritmetice): media anuală a elevilor, media vânzărilor zilnice, temperatura medie, consumul mediu de carburant etc.
3. gestionarea ofertelor: calcularea numărului de combinații diferite de meniuri la un restaurant, având mai multe opțiuni la fiecare categorie de mâncare. De exemplu, avem categoriile de mâncare fel principal și desert, câte 3 și, respectiv, 2 feluri din fiecare, iar numărul total de combinații posibile este produsul acestor valori.

Pentru a eficientiza procesul de calcul, vom realiza programe informatice. Algoritmul de calcul al sumei și al produsului elementelor unui șir sunt descrise mai jos:



Rețineți

Algoritmul pentru calculul sumei elementelor

1. Se inițializează suma cu 0 pentru ca valoarea sa inițială să nu afecteze valoarea sumei.
2. Se citește fiecare element al șirului și se adună la suma anterioară.

Algoritmul pentru calculul produsului elementelor

1. Se inițializează produsul cu 1 pentru ca valoarea sa inițială să nu afecteze valoarea produsului.
2. Se citește fiecare element al șirului și se înmulțește cu produsul anterior.

Vă propunem să scrieți și să rulați în Code::Blocks programele **A** și **B** de mai jos, care determină suma și produsul elementelor unor șiruri de numere, citite de la tastatură.

A. Suma elementelor unui șir de valori numerice

Problemă: Se citește un număr natural n , $1 \leq n \leq 20$, apoi un șir de n numere întregi din intervalul $[-1000, 1000]$. Afișați suma tuturor elementelor.

Indicații: Pentru a calcula suma elementelor unui șir de valori numerice, vom utiliza o variabilă în care stocăm valoarea sumei, pe care o inițializăm cu 0. Citim fiecare număr din șir și îl adăugăm la sumă.

Date de intrare: n – număr natural nenul, reprezintă numărul de elemente din șir
 val – număr întreg, valoare a termenului șirului

Date de manevră: i – număr întreg, contor pentru numărul termenului șirului

Date de ieșire: s – număr întreg, suma elementelor șirului de valori

Algoritmul în pseudocod	Codul în C++	
<pre> algoritm Suma întreg n, i, s, val Citește n s ← 0 Pentru i ← 1, n execută Citește val s ← s + val Scrie 'Suma este:', s Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i, s, val; cout<<"n="; cin>>n; s=0; </pre>	<pre> for (i=1; i<=n; i++) { cin>>val; s=s+val; } cout<<"Suma este: "<<s; return 0; </pre>

```

n=4 3 4 5 6
Suma este: 18
Process returned 0 (0x0)   execution time : 5.625 s
Press any key to continue.
        
```

B. Produsul elementelor unui șir de valori numerice

Problemă: Se citește un număr n , $1 \leq n \leq 20$, apoi un șir de n numere întregi din intervalul $[-1000, 1000]$. Afișați produsul tuturor elementor.

Indicații: Pentru a calcula produsul elementelor unui șir de valori numerice, vom utiliza o variabilă în care stocăm valoarea produsului, pe care o inițializăm cu 1. Citim fiecare număr din șir și îl înmulțim la produsul anterior.

Date de intrare: n – număr natural nenul, reprezintă numărul de elemente din șir

val – număr întreg, valoarea termenului șirului

Date de manevră: i – număr întreg, contor pentru numărul termenului șirului

Date de ieșire: p – număr întreg, produsul elementelor șirului de valori

Algoritmul în pseudocod	Codul în C++
<pre> algoritm Produs întreg n, i, p, val Citește n p ← 1 Pentru i ← 1, n execută Citește val p ← p * val Scrie 'Produsul este:', p Sfârșit </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i, p, val; cout<<"n="; cin>>n; p=1; for (i=1; i<=n; i++) { cin>>val; p=p*val; } cout<<"Produsul este: "<<p; return 0; } </pre>

```

n=4 3 4 5 6
Produsul este: 360
Process returned 0 (0x0)   execution time : 15.750 s
Press any key to continue.
        
```

Aplicați și investigați

- Urmăriți pas cu pas funcționarea programului **A** de mai sus, pentru setul de date de intrare: $n=4$ -1 3 -2 10 . Ce se afișează în acest caz?
- Urmăriți pas cu pas funcționarea programului **B** de mai sus, pentru setul de date de intrare: $n=4$ -1 3 -2 10 . Ce se afișează în acest caz?
- Investigați ce s-ar întâmpla dacă ați inițializa suma s cu -1 în programul **A** și produsul p cu 0 , în programul **B**.

Exersați

- Realizați un program C++ în care se calculează suma elementelor șirului numerelor naturale cuprinse în intervalul $[a, b]$, unde $1 \leq a < b \leq 10000$, a și b numere naturale.

Indicații: **Intrare:** $a=1$ $b=10$; **Ieșire:** 55

- Selectați ce anume se afișează în urma execuției codului de mai jos, dacă valoarea de intrare a variabilei d este 5.

```

int d, i, s=0;
cin>>d;
for (i=11; i<=30; i++)
{
    if (i%d==0)
        s=s+i;
}
cout<<s;
        
```

a. 80

b. 90

c. 71

- Selectați valoarea pe care trebuie să o primească variabila n în programul de mai jos, astfel încât să se afișeze 12.

```

int n, i, p=1;
cin>>n;
for (i=3; i<=n; i++)
    p=p*i;
cout<<p;
        
```

a. 4

b. 5

c. 1

Portofoliu

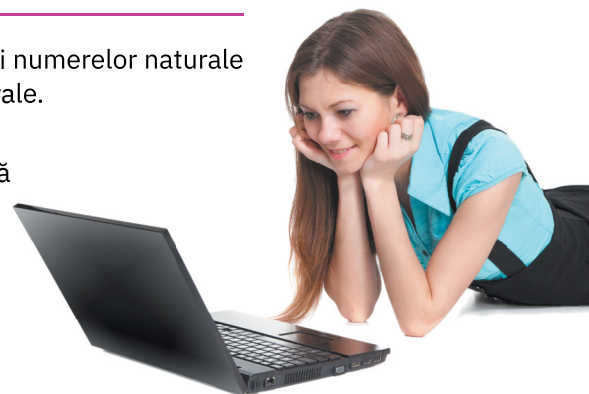


Realizați un program C++ care să calculeze diferența dintre suma și produsul elementelor pare ale unui șir de n elemente numere naturale mai mici decât 100000, unde $1 \leq n \leq 20$.

Indicații:

Intrare: $n=5$ 1 2 3 4 5

Ieșire: -2



Recapitulare

Scrieți programele C++ pentru rezolvarea problemelor următoare:

1. Afișați pe ecran primii n termeni nenuli ai șirului numerelor divizibile cu 7, unde $1 \leq n \leq 200$.

Indicații: **Intrare:** $n=5$

Ieșire: 7 14 21 28 35

2. Se introduc n elemente ale unui șir de numere naturale mai mici decât 1000, unde $1 \leq n \leq 20$. Afișați pe câte o linie separată valoarea fiecărui element citit.

Indicații: **Intrare:** $n=3$ 2 3 4

Ieșire: 2
3
4

3. Se citește un număr natural n , $1 \leq n \leq 20$, și apoi n numere întregi din intervalul $[-20000, 20000]$. Afișați doar numerele impare separate printr-un spațiu.

Indicații: **Intrare:** $n=5$ 4 7 0 15 21 **Ieșire:** 7 15 21

4. Se citesc trei numere naturale n , a și b ($1 \leq n \leq 20$, $1 \leq a < b \leq 20000$), urmate apoi de un șir de n numere întregi din intervalul $[-10000, 10000]$. Afișați doar numerele care sunt în intervalul $[a, b]$, separate printr-un spațiu.

Indicații: **Intrare:** $n=7$ $a=5$ $b=15$ 2 5 8 12 17 20 22 **Ieșire:** 5 8 12

5. Se citesc două numere naturale n , m ($1 \leq n \leq 20$, $1 \leq m \leq 10$), urmate de un șir de n numere întregi din intervalul $[-10000, 10000]$. Afișați doar numerele din șir care sunt multipli de m , separate printr-un spațiu.

Indicații: **Intrare:** $n=6$ $m=5$ 3 10 15 18 20 25 **Ieșire:** 10 15 20 25

6. Se citește un număr n ($1 \leq n \leq 20$), urmat de n numere întregi din intervalul $[-10000, 10000]$. Afișați primul număr negativ întâlnit. Dacă nu există, afișați „NU EXISTA”.

Indicații: **Intrare:** $n=5$ 4 7 -3 10 -5 **Ieșire:** -3

Intrare: $n=4$ 1 2 3 4 **Ieșire:** NU EXISTA

7. Se citesc n numere întregi din intervalul $[-10000, 10000]$, unde $1 \leq n \leq 20$. Afișați „Aceeși paritate” dacă fie toate numerele sunt pare, fie toate numerele sunt impare, și „Parități diferite”, altfel.

Indicație: Folosiți o variabilă tip semafor cu două valori (0 – dacă toate numerele sunt pare, 1 – dacă toate numerele sunt impare) și o inițializați cu valoarea 1. Se citește primul număr și se stabilește paritatea lui ($\text{primPar} = \text{numar} \% 2 == 0$). Parcurgeți șirul și verificați paritatea fiecărui element în parte. Dacă găsiți un număr cu **paritate diferită**, atunci modificați valoarea variabilei semafor la 0.

Indicații: **Intrare:** $n=4$ 2 4 6 8 **Ieșire:** Aceeași paritate

Intrare: $n=4$ 3 5 7 9 **Ieșire:** Aceeași paritate

Intrare: $n=4$ 2 5 6 8 **Ieșire:** Parități diferite

8. La o anumită disciplină, mediile elevilor dintr-o clasă cu n elevi ($1 \leq n \leq 30$) sunt înregistrate și studiate în vederea evaluării procesului educațional. Afișați numărul elevilor care au media 9 și 10, al elevilor cu mediile de 7 și 8, al celor cu mediile 5 și 6, al celor cu mediile mai mici de 5.

Indicații: Se va utiliza algoritmul prezentat în lecția **Numărarea elementelor care îndeplinesc o condiție**.

Intrare: $n=6$ 10 9 8 7 5 10 **Ieșire:** Mediile 9-10: 3, Mediile 7-8: 2, Mediile 5-6: 1, Medii sub 5: 0

9. Afișați numărul elementelor unui șir de n numere reale mai mici decât 1000, unde $1 \leq n \leq 100$, care sunt mai mari decât primul număr al șirului.

Indicații: **Intrare:** $n=5$ 7 15 9 6 11 **Ieșire:** 3

10. Afișați de câte ori apare, într-un șir de n numere naturale mai mici decât 1000, ultima cifră a primului număr.

Indicații: Dacă primul element al șirului conține o singură cifră, acesta se va număra.

Intrare: $n=6$ 1 2 1 7 5 3 **Ieșire:** Nr. aparitii: 2

11. Se dă un șir de n numere reale, unde $1 \leq n \leq 31$. Să se afișeze media aritmetică a elementelor șirului și numărul de elemente care au valoarea mai mare decât 15.

Indicații: Rezolvarea implică utilizarea unui algoritm simplu de calculare a mediei aritmetice, utilizând formula matematică.

Intrare: $n=6$ 10.5 11 9 15 17 16 **Ieșire:** Media este: 13.0833, Nr. elemente mai mari decat 15: 2



Rezolvați în caiete sau pe calculator exercițiile date. Evaluați-vă singuri rezultatele.

1. Șirul **X** conține elementele: 2, 4, 6, 8, 10, 12, Care este valoarea elementului de pe poziția 35 din șir?

a. 72; b. 70; c. 700; d. 35.

Indicație: Găsiți regula după care se formează elementele șirului.

2. Un șir de valori conține elementele: 1, 2, 3, 4, 5, 6, 7, 8, 9. Completați valoarea lipsă din secvența din *imaginea a*, astfel încât la rularea ei să se afișeze 357911.

Indicație: Calculați valoarea care se afișează la fiecare pas, valoare care compune rezultatul afișat.

3. Se consideră șirul numerelor întregi cuprinse în intervalul $[x, y]$, unde $-5000 \leq x < y \leq 5000$. Selectați ce anume se va afișa în urma rulării codului din *imaginea b*, dacă pentru **x** și **y** se introduc valorile 24 și, respectiv, 30.

a. 24 25 26 27 28 29 30 b. 24 26 28 30 c. 48 52 56 60

Indicație: Calculați valoarea care se afișează la fiecare pas, valoare care compune rezultatul afișat.

4. Ce valoare trebuie să primească variabila întreagă **x** pentru ca, în urma executării secvenței de cod din *imaginea c*, cel mai mare element al șirului de valori afișat să fie 38?

Indicație: Observați faptul că elementul cu valoarea maximă al șirului este afișat atunci când $i=5$. Calculați **x** pentru a fi afișată valoarea 38 în acest caz.

5. Se dă secvența de cod alăturată. Scrieți în caiete care este varianta corectă dintre cele de mai jos:

a. se afișează elementele impare de pe pozițiile impare ale șirului de numere;
b. se afișează elementele pare de pe pozițiile impare ale șirului de numere;
c. se afișează elementele impare de pe pozițiile pare ale șirului de numere.

Indicație: Analizați condiția din instrucțiunea decizională *if*, ținând cont de semnificația variabilelor (**nr** – elementul citit, **k** – poziția elementului în șir).

6. Ana știe care a fost totalul veniturilor ei din fiecare lună a unui an. Scrieți un program C++ care afișează luna în care ea a avut cel mai mare venit.

Indicații: **Intrare:** 4234 4432 4222 4001 4978 4987 5223 4455 5500 4378 3226 3190; **Ieșire:** Luna septembrie.

Problema inițială se transpune într-una de aflare a maximului elementului unui șir de valori.

7. Se dă un șir de **n** numere întregi, unde $1 \leq n \leq 100$. Scrieți un program C++ care să verifice dacă toate elementele șirului au ultima cifră egală cu 3.

Indicații: **Intrare:** $n=3$ 3 12 13; **Ieșire:** Nu

Intrare: $n=3$ 3 13 23; **Ieșire:** Da

Este o problemă de verificare a existenței elementelor care au anumite proprietăți. Amintiți-vă de problemele cu numere, în care lucrați cu cifrele acestora.

a

```
a=1;
for (i=1; i<=...; i++)
{
    cin>>nr;
    cout<<nr*2+a;
}
```

b

```
int x, y, i;
cin>>d;
for (i=x; i<=y; i=i+2)
    cout<<2*i<<" ";
```

c

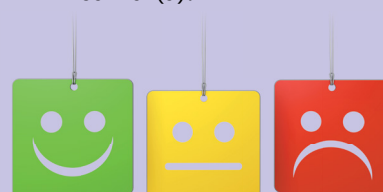
```
x=...;
for (i=0; i<=5; i++)
    cout<<x*i+3;
```

d

```
int nr;
for (k=30; k>=1; k--)
{
    cin>>nr;
    if (nr%2==0 && k%2==1)
        cout<<nr<<" ";
}
```

CUM VĂ APRECIĂȚI?

1. Evaluați-vă singuri rezultatele utilizând grila de punctaje.
2. Scrieți cum vă simțiți după ce ați rezolvat acest test, alegând unul dintre cuvintele de mai jos:
 - nemulțumit(ă);
 - mulțumit(ă);
 - încântat(ă).



Punctaje:

1 – 10 puncte;

2 – 5 puncte;

3 – 15 puncte;

4 – 20 de puncte;

5 – 20 de puncte;

6 – 10 puncte;

7 – 20 de puncte.

Insuficient: 0-50 de puncte

Suficient: 51-60 de puncte

Bine: 61-80 de puncte

Foarte bine: 81-90 de puncte

Excelent: 91-100 de puncte

Punctaj maxim: 100 de puncte. ● Timp de lucru: 50 de minute.

Răspunsurile se găsesc la finalul manualului.

Proiect: Liceul preferat

Descrierea proiectului

Acest proiect își propune să analizeze preferințele elevilor de clasa a VIII-a cu privire la continuarea studiilor la diferite licee. Vom colecta răspunsurile printr-un sondaj și vom folosi un program în C++ pentru a calcula și analiza statistic datele.

Timpul de lucru

Alocat de cadrul didactic.

Etapetele proiectului

1. Crearea sondajului

Fiecare elev din clasă va răspunde la întrebarea: **Cât de mult vă doriți să continuați clasa a IX-a la următoarele licee?** Chestionarul va conține o listă de licee care poate fi adaptată în funcție de preferințele clasei. Răspunsul elevilor va fi exprimat pe o scală de la 1 la 5 (1 – Foarte puțin, 2 – Puțin, 3 – Mediu, 4 – Mult, 5 – Foarte mult):

Nume liceu	1 – Foarte puțin	2 – Puțin	3 – Mediu	4 – Mult	5 – Foarte mult
...					
...					
...					

Elevii pot lucra colaborativ pe un fișier Excel creat în prealabil de profesor, fiecare având alocată o foaie de lucru.

2. Cumularea și prelucrarea datelor

Elevii vor fi împărțiți în grupe de câte 3 elevi, fiecărei grupe i se va alocă un număr egal de licee dintre cele analizate, minimum două. Elevii vor trebui să introducă răspunsurile legate de fiecare liceu în parte într-un program realizat de ei în C++ și să le prelucreze.

Programul citește n numere naturale cuprinse între 1 și 5 (răspunsurile elevilor), n fiind numărul de elevi ai clasei, și afișează procentul elementelor egale cu 1, cu 2, cu 3, cu 4 și cu 5, ca numere reale.

Exemplu: Intrare: $n=10$ 1 2 5 1 2 3 2 4 5 5; **Ieșire:** % valoare 1: 20%, % valoare 2: 30%, % valoare 3: 10%, % valoare 4: 10%, % valoare 5: 30%

Indicații: Programul utilizează algoritmul învățat pentru numărarea elementelor care îndeplinesc o condiție, pentru numărarea fiecărui tip de răspuns (1, 2, 3, 4 sau 5). Calculează, apoi, procentul elementelor de fiecare tip după formula: $(float)<Număr\ elemente\ care\ îndeplinesc\ condiția\ dată>/<Număr\ total\ de\ elemente\ în\ șir>*100$.

3. Interpretarea datelor

Fiecare echipă va prezenta în fața clasei rezultatele obținute pentru liceele la care a prelucrat răspunsurile la sondaje și va realiza o interpretare a acestora, identificând liceul preferat dintre cele analizate. Prezentarea va fi realizată în PowerPoint și va include grafice realizate în Excel, pe baza procentelor extrase ca rezultat din execuția programului.

Cum se face evaluarea

Profesorul va acorda un punctaj fiecărei echipe, de maximum 100 de puncte, utilizând următoarele criterii de punctare:

Criteriu	Punctaj (maxim)
Realizare program C++	25 de puncte
Interpretarea rezultatelor	25 de puncte
Realizarea prezentării concluziilor analizei datelor	25 de puncte
Colaborare eficientă între colegi	10 puncte
Finalizare la timp	5 puncte
Puncte din oficiu	10 puncte
Total:	100 de puncte



Autoevaluarea proiectului

- Am respectat cerințele proiectului:
 - În întregime
 - Parțial
- În timpul de lucru alocat am terminat proiectul:
 - În întregime
 - Parțial
- Am contribuit activ la realizarea proiectului:
 - Am participat activ la toate etapele
 - Am contribuit doar la unele părți
 - Am participat foarte puțin
- Comunicarea cu colegii a fost:
 - Foarte bună
 - Bună
 - Slabă
- Am utilizat corect limbajul C++ pentru implementarea programului:
 - Fără erori și optimizat
 - Cu mici erori corectate ulterior
 - Cu dificultăți majore, necesitând ajutor constant
- Prezentarea și interpretarea rezultatelor a fost:
 - Corectă, clară și bine argumentată
 - Acceptabilă, dar putea fi îmbunătățită
 - Nesatisfăcătoare, necesitând mai multă pregătire

Recapitulare finală

Alegeți răspunsurile corecte și răspundeți cerințelor următoare:

- Într-o foaie de calcul tabelar Excel, celula **G11** se află:
 - la intersecția dintre rândul **G** și coloana **11**;
 - la intersecția dintre rândul **11** și coloana **G**;
 - la intersecția dintre rândul **G1** și coloana **1**;
 - la intersecția dintre rândul **G111** și coloana **1**.

- Ce valoare se va afișa în celula **D5** dacă într-o foaie de calcul tabelar Excel introducem valorile la fel ca în imaginea de mai jos, iar în celula **D5** se introduce formula **=IF(B5<100,100,200)-IF(C5>20,20,10)**?

- 90;
- 190;
- 80;
- 180.

a

	A	B	C	D
5	1	75	21	
6	2	150	15	

- Marius face trei măsurători ale lungimii pentru trei piese și înscrie rezultatele într-o foaie de calcul tabelar, calculând în fiecare celulă de pe coloanele **E** și, respectiv, **F**, minimul și maximul pentru valorile înscrise pe rândul corespunzător acestuia.

Precizați cărei celule din tabelul prezentat în imaginea **b** îi corespunde fiecare dintre următoarele formule:

- **=MIN(B2,C2,D2)**
 - E2;
 - F2;
 - AE2;
 - niciunul.
- **=MAX(B4,C4,D4)**
 - E2;
 - EF4;
 - F4;
 - niciunul.

b

	A	B	C	D	E	F
1		I1	I2	I3	Min	Max
2	piesa 1	12.50	12.75	12.60		
3	piesa 2	45.00	45.56	44.97		
4	piesa 3	23.00	23.25	22.54		

- Paul a avut în clasa a V-a media generală 9,20. El și-a mărit media cu 0,20 în fiecare an, până la finalul clasei a VIII-a. Realizați un tabel care arată mediile anuale ale lui Paul în toți cei patru ani de studiu (în clasele V-VIII) și reprezentați grafic evoluția mediei anuale a lui Paul.

- Zona de celule **A4:E4** include următoarele celule:

- A4, E4;
- A4, C4, E4;
- A4, B4, C4, D4, E4;
- B4, C4, D4.

- Ce este o subpagină web?

- Odată activată opțiunea de partajare și colaborare pe Google Sites pentru un coleg (la un proiect X), se mai poate opri partajarea pentru a bloca accesul colegului la respectivul proiect?

- Alegeți **A**, dacă enunțul este adevărat, și **F**, dacă enunțul este fals:

- Un carusel de imagini nu poate avea mai multe pagini.
- În Google Sites, la pagina **Acasă**, nu poate fi adăugată o subpagină.
- Pe o pagină din Google Sites putem include filme YouTube, diagrame Excel, imagini.

A / F
A / F
A / F

- Asociați imaginea butonului cu acțiunea care-i corespunde în Google Sites:

-
-
-

- Inserarea unei hărți în pagină
- Partajarea accesului la proiect
- Adaugarea unui carusel de imagini
- Previzualizarea paginii

10. Cum putem publica un site realizat cu Google Sites?

- a. pe un domeniu propriu de tipul `www.domeniulmeu.ro`;
- b. la o adresă pusă la dispoziție de Google, de tipul `https://sites.google.com/view/siteulmeu`;
- c. niciuna dintre acestea; pentru publicare avem nevoie de un server propriu.

11. Ce afișează secvența de instrucțiuni din *imaginea c*?

- a. 012345
- b. 0246810
- c. 0 2 4 6 8 10

```
int n, i;
for(i=0; i<=5; i++)
    cout<<2*i<<" ";
```

12. Ce se va afișa după executarea secvenței de cod din *imaginea d*?

- a. 6 12 8 18 10 24 12 30 14 36
- b. 6 7 8 9 10 11 12 13 14 15
- c. 9 7 15 9 21 11 27 13 33 15

```
int i;
for(i=3; i<=12; i++ )
{ if (i%2==1)
    cout<<i+3<<" ";
  else
    cout<<i*3<<" ";
}
```

13. În secvența de cod din *imaginea e*, câte dintre numerele afișate au cifra unităților egală cu 8?

- a. 6
- b. 3
- c. 10

```
int i;
for(i=0; i<15; i++)
    cout<<3*i+i*i<<" ";
```

14. Selectați care este valoarea inițială lipsă a variabilei **S**, astfel încât secvența de cod din *imaginea f* să afișeze media aritmetică a elementelor divizibile cu 3 ale unui șir de numere.

- a. 2
- b. 0
- c. 10.0

```
int n, i, nr=0, val;
float S;
S=...;
cout<<"n=";cin>>n;
for(i=0; i<=n-1; i++)
{ cin>>val;
  if(val%3==0)
  { S=S+val;
    nr++;
  }
}
S=S/nr;
cout<<S;
```

Pentru fiecare dintre exercițiile de mai jos, scrieți programele C++ corespunzătoare și testați-le pe calculator.

15. Se dă un șir de numere întregi cu **n** elemente, unde $1 \leq n \leq 150$. Triplați toate elementele șirului și afișați-le. Toate elementele sunt mai mici decât 100.

Indicații: **Intrare:** `n=5 1 2 3 4 5` **Ieșire:** `3 6 9 12 15`

16. În cadrul unei campanii de marketing, toate prețurile unor produse se reduc cu 20%. Dacă prețurile inițiale dinainte de campanie sunt incluse într-un șir de valori **p**, iar numărul de produse este **n**, unde $1 \leq n \leq 150$, afișați lista prețurilor noi din campanie. Prețurile inițiale sunt mai mici decât 2000 RON.

Indicații: **Intrare:** `n=3 7.5 10.0 20.5` **Ieșire:** `6 8 16.4`

17. Se consideră un șir de **n** numere naturale mai mici decât 200, unde $1 \leq n \leq 100$. Scrieți un program care:

- a. afișează suma elementelor multipli de 11 din șir;
- b. afișează suma dintre primul element din șir și maximul șirului;
- c. afișează dacă există în șir numărul 77.

Indicații: **Intrare:** `n=7 1 22 33 44 55 66 77`

Ieșire: a. 297; b. 78; c. Exista.

18. Se dă un șir de numere naturale de **n** elemente, unde $1 \leq n \leq 100$. Scrieți un program care să afișeze câte dintre elementele acestui șir au suma cifrelor egală cu 7, dacă numerele au maximum 3 cifre.

Indicații: **Intrare:** `n=7 25 12 34 56 38 22 47`

Ieșire: 2



- 19.** Un lanț de supermarketuri are 6 magazine numerotate de la 1 la 6. Fiecare magazin are un număr de maximum 10 000 de produse. Realizați un program care să afișeze magazinele în care produsele pot fi împachetate în cutii cu o capacitate de 50 de produse, fără a mai rămâne vreun produs în plus.
Indicații: **Intrare:** 2300 7200 5324 3748 9867 8550 **Ieșire:** 1 2 6
- 20.** Rezultatele la testul de la matematică ale elevilor unei clase sunt înscrise într-un șir de n numere ($1 \leq n \leq 30$). Afișați media pe clasă și numărul de elevi care au nota mai mare decât 7. Afișați și dacă există copii care au nota 5 la test.
Indicații: Dacă $n=5$, iar șirul este 10 8 9 10 5, atunci se afișează: Media=8.4, Peste 7=4, Exista nota 5
- 21.** Ajută un angajat să își planifice concediul. Acesta vrea să afle câte zile de weekend (sâmbătă și duminică) există într-o perioadă de n zile ($1 \leq n \leq 31$). Pentru aceasta, el introduce un șir de n valori, fiecare valoare indicând ziua săptămânii (de la 1 = luni până la 7 = duminică). Trebuie afișat câte valori 6 (sâmbătă) și 7 (duminică) sunt în șir.
Indicații: **Intrare:** $n=10$ 1 2 3 4 5 6 7 1 2 6 **Ieșire:** 3
- 22.** Se dă un șir de n numere reale, unde $1 \leq n \leq 100$. Numerele sunt cuprinse în intervalul $[1, 100]$. Să se afișeze dacă acesta conține numere cuprinse în intervalul $[1, 25]$.
Indicații: **Intrare:** $n=3$ 3.4 72.0 5.2 **Ieșire:** Da
Intrare: $n=3$ 27.1 34.3 76.5 **Ieșire:** Nu
- 23.** Se dă un șir de n numere întregi mai mici decât 1000, unde $1 \leq n \leq 100$. Afișați „Da”, dacă oricare două numere consecutive ale șirului au parități diferite (unul par și celălalt impar), respectiv „Nu”, în caz contrar.
Indicații: **Intrare:** $n=5$ 2 5 4 3 12 **Ieșire:** Da
Intrare: $n=5$ 2 6 4 3 12 **Ieșire:** Nu
- 24.** Se dă un șir de n numere naturale mai mici decât 10000, unde $1 \leq n \leq 100$. Scrieți un program care calculează suma elementelor șirului a căror ultimă cifră este 6.
Indicații: **Intrare:** $n=5$ 12 13 56 3 36 **Ieșire:** 92
- 25.** Să se calculeze produsul celui mai mic element al unui șir de n numere naturale mai mici decât 20000, cu cel mai mare element al aceluiași șir, unde $1 \leq n \leq 100$.
Indicații: **Intrare:** $n=5$ 12 13 56 3 36 **Ieșire:** 168
- 26.** În cadrul orei de educație fizică au fost înregistrate rezultatele elevilor la proba de alergare viteză 50 m (în secunde). Faceți un program care să afișeze cel mai mic timp realizat, cel mai mare timp realizat și timpul mediu, utilizând media aritmetică. Numărul total al elevilor este 35 și timpii de alergare sunt mai mici de 30s.
Indicații: **Intrare:** $n=5$ 6.3 7 7.5 8.5 10 **Ieșire:** Minim: 6.3; Maxim: 10; Mediu: 7.86

Fișă de autoevaluare a activității și a comportamentului

Realizează o fișă în genul celei de mai jos, folosind culorile indicate la casete.

Numele și prenumele: Clasa:

Cum te apreciezi? Scrie în caseta albastră numărul din dreptul variantei tale de răspuns.

1 Într-o foarte mare măsură 2 Într-o mare măsură 3 Într-o oarecare măsură 4 În mică măsură 5 Deloc

Doamna/domnul profesor va scrie în caseta verde varianta sa de răspuns.

Am participat cu plăcere la rezolvarea sarcinilor de învățare.		
Am participat cu plăcere la discuții, dezbateri.		
Răspunsurile mele au fost în legătură cu tema discuției.		
Am fost atent/atentă la desfășurarea activităților din timpul orelor de curs.		
Mi-a plăcut să particip la activități organizate în echipă.		
Am cerut ajutor atunci când am avut nevoie.		
Am dus la bun sfârșit sarcinile primite la școală, în cadrul orelor de <i>Informatică și TIC</i> .		
Am împărtășit rezultatele investigațiilor sau proiectelor celorlalți colegi.		

Răspunsuri (selectiv)

Unitatea I – Autoevaluare (p. 33)

1. A se vedea p. 12. **2.** a, b, c, d, e, f. **3.** a – 4; b – 1; c – 2; d – 3. **4.** a, b, c, e, f. **5.** a, c, d, e. **6.** A se vedea p. 17. **7.** a – DA; b – NU (butonul **Sortare** este în fila **Date**); c – DA. **8.** a. comenzile și elementele, calcul, tabele, diagrame, texte (p. 15); b. calcul, Imprimare, imprimantă (p. 18); c. decizie, Excel, logice, două (p. 27). **9.** A se vedea p. 18. **10.** A se vedea p. 29.

Unitatea II – Autoevaluare (p. 55)

1. A se vedea p. 36. **2.** a, b, c, d, e, f. **3.** c – 1; d – 2; a – 3; b – 4. **4.** a, b, d, e, f. **5.** A se vedea p. 52. **6.** A se vedea p. 46. **7.** a – DA; b – DA; c – DA. **8.** a. utilizator, motoarele, sugestiv, informații; b. subsolul, subsol, subsol, subsol, subsolul. **9.** d. **10.** A se vedea p. 43.

Unitatea III – Autoevaluare (p. 89)

1. b. **2.** $i \leq 5$. **3.** c. **4.** $x=7$; **5.** b.

```
6. #include <iostream>
using namespace std;
int main() {
    int venit, i, M=0, luna=0;
    for (i=1; i<=12; i++) {
        cin>>venit;
        if (M<venit) {
            M=venit;
            luna=i;
        }
    }
    switch (luna) {
        case 1: cout<<"Luna ianuarie.";break;
        case 2: cout<<"Luna februarie.";break;
        case 3: cout<<"Luna martie.";break;
        case 4: cout<<"Luna aprilie.";break;
        case 5: cout<<"Luna mai.";break;
        case 6: cout<<"Luna iunie.";break;
        case 7: cout<<"Luna iulie.";break;
        case 8: cout<<"Luna august.";break;
        case 9: cout<<"Luna septembrie.";break;
        case 10: cout<<"Luna octombrie.";break;
        case 11: cout<<"Luna noiembrie.";break;
        case 12: cout<<"Luna decembrie.";
    }
    return 0;
}
```

```
7. #include <iostream>
using namespace std;
int main() {
    int nr, i, eEgal=1, n;
    cout<<"n="; cin>>n;
    for (i=1; i<=n; i++) {
        cin>>nr;
        if (nr%10!=3) eEgal=0;
    }
    if (eEgal==1)
        cout<<"Da";
    else
        cout<<"Nu";
    return 0;
}
```

Proiect: Liceul preferat (p. 90)

```
#include <iostream>
using namespace std;
int main() {
    int n, nr i;
    int nr1=0, nr2=0, nr3=0, nr4=0, nr5=0;
    cout<<"n="; cin>>n;
    for (i=1; i<=n; i++) {
        cin>>nr;
        if (nr==1) nr1++;
        if (nr==2) nr2++;
        if (nr==3) nr3++;
        if (nr==4) nr4++;
        if (nr==5) nr5++;
    }
    //se afiseaza procentul elementelor 1, 2, 3, 4, 5
    cout<<"% valoare 1: "<<(float)nr1/n*100<<"%"<<endl;
    cout<<"% valoare 2: "<<(float)nr2/n*100<<"%"<<endl;
    cout<<"% valoare 3: "<<(float)nr3/n*100<<"%"<<endl;
    cout<<"% valoare 4: "<<(float)nr4/n*100<<"%"<<endl;
    cout<<"% valoare 5: "<<(float)nr5/n*100<<"%"<<endl;
    return 0;
}
```

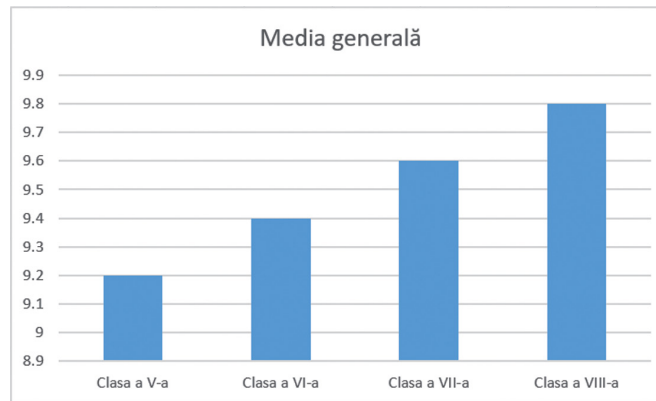


Recapitulare finală (p. 91)

1. b. 2. c. 3. a; c.

4.

	A	B
1	Anul de studiu	Media generală
2	Clasa a V-a	9.2
3	Clasa a VI-a	9.4
4	Clasa a VII-a	9.6
5	Clasa a VIII-a	9.8



5. c. 6. A se vedea p. 47. 7. Da. 8. a – F; b – F; c – A. 9. a – 4; b – 1; c – 2. 10. a; b. 11. c. 12. a. 13. a. 14. b.

```
15. #include <iostream>
using namespace std;
int main() {
    int n, i, nr;
    cout<<"n="; cin>>n;
    for(i=1; i<=n; i++){
        cin>>nr;
        cout<<nr*3<<" ";
    }
    return 0;
}
```

```
16. #include <iostream>
using namespace std;
int main() {
    int n, i;
    float nr;
    cout<<"n="; cin>>n;
    for(i=1; i<=n; i++){
        cin>>nr;
        cout<<nr*(1-0.2)<<" ";
    }
    return 0;
}
```

```
17. #include <iostream>
using namespace std;
int main() {
    int i, n, nr, sM11=0, pnr;
    int s=0, max=0, exista=0;
    cout<<"n="; cin>>n;
    cin>>nr;
    pnr=nr;
    max=nr;
    for (i=2; i<=n; i++){
        cin>>nr;
        if (nr%11==0) s=s+nr;
        if (nr>max) max=nr;
        if (nr==77) exista=1;
    }
    cout<<"a) "<<s<<" ";
    cout<<"b) "<<pnr+max<<" ";
    if (exista==1) cout<<"c) Exista";
    else cout<<"c) Nu exista";
    return 0;
}
```

```
18. #include <iostream>
using namespace std;
int main() {
    int n, i, s, cnt=0, nr;
    cout<<"n="; cin>>n;
    for(i=1; i<=n; i++){
        cin>>nr;
        s=0;
        while (nr!=0){
            s=s+nr%10;
            nr=nr/10;
        }
        if (s==7) cnt++;
    }
    cout<<cnt;
    return 0;
}
```

```
19. #include <iostream>
using namespace std;
int main() {
    int i, nrprod;
    for(i=1; i<=6; i++){
        cin>>nrprod;
        if (nrprod%50==0)
            cout<<i<<" ";
    }
    return 0;
}
```

```
20. #include <iostream>
using namespace std;
int main() {
    int n, i, nota, s=0, nPeste7=0, n5=0;
    float media=0;
    cout<<"n="; cin>>n;
    for(i=1; i<=n; i++){
        cin>>nota;
        s=s+nota;
        if (nota>7) nPeste7++;
        if (nota==5) n5++;
    }
    media=(float)s/n;
    cout<<"Media="<<media<<" ";
    cout<<"Peste 7="<<nPeste7<<" ";
    if (n5>0) cout<<"Exista nota 5";
    else cout<<"Nu exista nota 5";
    return 0;
}
```



```

21. #include <iostream>
using namespace std;
int main() {
    int n, zi, cnt = 0;
    cout<<"n="; cin>>n;
    for (int i=1; i<=n; i++){
        cin>>zi;
        //dc. e sambata sau duminica
        if (zi==6 || zi==7)
            cnt++;
    }
    cout<<cnt;
    return 0;
}

```

```

22. #include <iostream>
using namespace std;
int main() {
    int n, cnt=0;
    float nr;
    cout<<"n="; cin>>n;
    for (int i=1; i<=n; i++){
        cin>>nr;
        if (nr>=1 && nr<=25)
            cnt++;
    }
    if (cnt>0) cout<<"Da";
    else cout<<"Nu";
    return 0;
}

```

```

23. #include <iostream>
using namespace std;
int main() {
    int nr_ant, nr, n, ok=1;
    cout<<"n="; cin>>n;
    cin>>nr;
    nr_ant=nr;
    for (int i=2; i<=n; i++){
        cin>>nr;
        //verif dc au aceeasi paritate
        if ((nr+nr_ant)%2==0)
            ok=0;
        nr_ant=nr;
    }
    if (ok==1) cout<<"Da";
    else cout<<"Nu";
    return 0;
}

```

```

24. #include <iostream>
using namespace std;
int main() {
    int nr, n, s=0;
    cout<<"n="; cin>>n;
    for (int i=1; i<=n; i++){
        cin>>nr;
        if (nr%10==6)
            s=s+nr;
    }
    cout<<s;
    return 0;
}

```

```

25. #include <iostream>
using namespace std;
int main() {
    int nr, n, min=20000, max=0;
    cout<<"n="; cin>>n;
    for (int i=1; i<=n; i++){
        cin>>nr;
        if (nr>max) max=nr;
        if (nr<min) min=nr;
    }
    cout<<min*max;
    return 0;
}

```

```

26. #include <iostream>
using namespace std;
int main() {
    int n, i;
    float nr, p=1;
    cout<<"n="; cin>>n;
    for (i=1; i<=n; i++){
        cin>>nr;
        if (i%2==0) p=p*nr;
    }
    cout<<p;
    return 0;
}

```



