

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII

**Melinda-Emilia Coriteac • Carmen Diana Cosman
Iosif-Viorel Curta • Amelia Stoian**

INFORMATICĂ și TIC

**Manual
pentru
clasa a VIII-a**

8



Acest manual este proprietatea Ministerului Educației și Cercetării.

Acest proiect de manual școlar este realizat în conformitate cu Programa școlară aprobată prin O.M.E.N. nr. 3393/28.02.2017.

Numărul **119** – număr unic
de telefon la nivel național pentru
cazurile de abuz împotriva copiilor și
116.111 – numărul de telefon
de asistență pentru copii

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII

Melinda-Emilia Coriteac • Carmen Diana Cosman
Iosif-Viorel Curta • Amelia Stoian

Informatică și TIC

Manual pentru clasa a VIII-a



București, 2025

Manualul școlar a fost aprobat prin Ordinul de ministru nr.

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:

- **Tușe Delia**, lector universitar, doctor în Informatică, titular la Universitatea din Oradea, Facultatea de Informatică și Științe, Departamentul de Matematică și Informatică
- **Crăciunescu Georgeta Antonia Rodica**, profesor titular la Colegiul Național „Elena Cuza” București
- **Gal-Chiș Călin-Eugen-Nicolae**, lector universitar dr. titular la Departamentul de Informatică al Facultății de Management, a Universității Emanuel din Oradea

Redactor: Ioana Nițulescu

Grafică și tehnoredactare: Adrian-Viorel Rașca

Credite foto și video: Canva, Freepik, Pixabay

Platforma e-learning, activități digitale interactive, voce, procesare sunet și animații:
IMAGINE INFINITY SRL

© 2025 – EDITURA SIGMA

Toate drepturile asupra prezentei ediții aparțin Editurii SIGMA. Nicio parte a acestei lucrări nu poate fi reprodusă fără acordul scris al Editurii SIGMA.

Editura SIGMA

Sediul central:

Str. Nicolae Cartojan, nr. 11-13, sector 2, București,
cod poștal 023815

Tel.: 0758 044 829; 0723 541 020

E-mail: office@editurasigma.ro

Web: editurasigma.ro

Distribuție:

Str. Nicolae Cartojan, nr. 11-13, sector 2, București,
cod poștal 023815

Tel.: 0758 044 825; 0758 044 828

E-mail: comenzi@editurasigma.ro

Prezentarea manualului de INFORMATICĂ ȘI TIC

Ce ne-am dorit?

Formarea deprinderilor de utilizare a sistemelor de calcul și a tehnologiei moderne este necesară încă de la cele mai fragede vârste. Acest manual vă va ghida în asimilarea de cunoștințe la disciplina *Informatică și TIC* și în formarea de abilități practice în utilizarea editoarelor de calcul tabelar, în crearea de site-uri web complexe și nu în ultimul rând în formarea deprinderilor de utilizare a șirurilor de valori pentru prelucrarea datelor, utilizând limbajul de programare C++.

Manualul conține trei capitole. Fiecare capitol este format din mai multe lecții. La finalul fiecărei lecții aveți la dispoziție câte o **fișă de lucru** care vă va ajuta în procesul de fixare și asimilare a cunoștințelor și o **fișă de autoevaluare** care vă va ajuta să determinați ce ați reușit să asimilați din lecție. Fiecare capitol conține instrumente moderne de evaluare (**Portofoliu, Proiect, Investighează**) și se încheie cu un **test de verificare a cunoștințelor**. La începutul și sfârșitul manualului sunt propuse **fișe de recapitulare** și **teste de evaluare inițială**, respectiv **finală**, menite să evalueze gradul de asimilare a noțiunilor studiate pe parcursul anului școlar. Manualul mai conține și o **fișă de observare a comportamentului și activității elevului**, cu ajutorul căreia puteți analiza comportamentul și activitatea de pe parcursul orei.

Titlul capitolului

Titlul lecției

Definiție

Observație

Continutul lecției

Fișă de lucru

Alte elemente grafice utilizate în manual



Cerință



Exemplu



Reține!



Să ne reamintim!



Investighează!



Verifică-ți cunoștințele!



Portofoliu



Proiect



Aplicație



Aplicații suplimentare

Capitolul 2. Pagini web
Competențe: 1.2, 2.2

Lecția 8. Limbajul HTML. Editare de pagini web

Limbajul HTML. Noțiuni generale

Pagina web este un fișier cu extensia **.htm** sau **.html**, scris cu ajutorul unui limbaj numit **HTML** (Hypertext Markup Language). Acest limbaj este interpretat cu ajutorul browserului, care arată utilizatorului conținutul paginii.

Limbajul **HTML**, folosit pentru crearea paginilor web (adică pentru scrierea codului sursă al paginii) este alcătuit din cuvinte speciale, numite **etichete** (tag-uri sau marcare). Acestea sunt scrise între caracterele **<** și **>** și marchează elementele paginii web. (Ex: **<HTML>**)

Majoritatea etichetelor au etichete pereche, care conțin în plus caracterul **>** și marchează finalul acțiunilor etichetelor ale celor pereche sunt. (Ex: **<HTML>**)

Unele etichete conțin cuvinte rezervate, numite **atribute**, cu ajutorul cărora se pot modifica proprietățile etichetelor (Ex: **** - în eticheta **FONT** e utilizat atributul **color**).

Știați că?

- 1989 - Tim Berners-Lee a inventat **web**.
- 1991 - Tim Berners-Lee a inventat **HTML**.
- 1995 - A fost lansat **HTML 2.0**.
- 1997 - Consorțiul W3C recomandă **HTML 3.2**.
- 1999 - Consorțiul W3C recomandă **HTML 4.01**.
- 2000 - Consorțiul W3C recomandă **XHTML 1.0**.
- 2014 - Consorțiul W3C recomandă **HTML5**.
- 2017 - Consorțiul W3C recomandă **HTML 5.2**.

Editor de pagini web - Instrumente de bază ale unui editor de pagini web

Un editor de pagini web este un program complex, specializat în elaborarea site-urilor. Există editori care pot fi instalate în computer (**Microsoft FrontPage, Notepad++, Dreamweaver, KompoZer**) și editoare web online (**Google Sites, WordPress, W3Schools, Joomla, Weebly**).

- O pagină web poate conține texte (titluri, paragrafe), imagini, liste, tabele și alte obiecte.
- Legătura între fișierele unui site se face prin elemente de tip **hyperlink**.
- Orice editor de pagini web trebuie să conțină instrumente de lucru cu ajutorul cărora se pot insera și modifica elementele de conținut ale paginii.

Fișă de lucru 3

Creați, în dosarul **Capitol1**, dosarul numit **Fișă3** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

(*) Deschideți registrul de calcul **operatii_fol.xlsx** din dosarul **Fișă2** și realizați cerințele următoare:

- În foaia de calcul **B**, stabiliți înălțimea primelor 5 rânduri la 25pt și lățimea primelor 4 coloane de 20 (caractere).
- Scrieți în celula **A1** textul **Fișă de lucru 1**, în celula **A2** textul **Fișă de lucru 3**, iar în celula **A3** textul **Fișă de lucru 4**.
- În foaia de calcul **C**, în celula **A1**, scrieți textul **Microsoft Excel - calcul tabelar**.
- Copiați conținutul coloanei **A** și lipiți-l în coloana **C**. Ștergeți coloana **B**.
- Ascundeți rândul 2.
- Stergeți textul **Fișă de lucru 2**, din ambele celule în care apare.
- În foaia de calcul **C**, în celula **A1**, scrieți textul **Microsoft Excel - calcul tabelar**.
- Ajustați lățimea coloanei **A**, astfel încât aceasta să fie potrivită automat la conținut.
- Faceți setările necesare astfel încât primele 3 rânduri să rămână mereu vizibile în cazul derulării conținutului în jos.
- În foaia de calcul **A** blocați la derularea primele 2 rânduri și prima coloană.

Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum **1 - da**, **1 - nu**, **1 - nu sunt sigur(a)**, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fișă3**.

La finalul lecției, știu...	Da	Nu	Nu sunt sigur(a)
1. să inserez o foaie de calcul.			
2. să șterg și să redenumesc o foaie de calcul.			
3. să copiez/mut o foaie în alt registru.			
4. să ascund o foaie de calcul.			
5. să inserez un rând nou și o coloană.			
6. să modific înălțimea unui rând.			
7. să modific lățimea unei coloane.			
8. să copiez conținutul unui rând/coloană.			
9. să mut conținutul unui rând/coloană.			
10. să blochez la derularea anumite rânduri/coloane.			
11. Am întâmpinat dificultăți în: ...			

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției! (cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit: 😊 😐 😞

Știați că?

Autoevaluare!

Manualul digital

Varianta digitală a manualului conține pe lângă varianta tipărită și activități multimedia interactive de învățare (AMII). Acestea vin în completarea noțiunilor și exemplelor prezentate în manualul tipărit și sunt de trei tipuri: statice, dinamice și interactive.



AMII statice: desene, fotografii sau planșe didactice.



AMII dinamice: filme sau animații.



AMII interactive: diverse exerciții, jocuri educative.



Accesare ajutor general manual (Help).

CUPRINS

Prezentarea manualului	3
Recapitulare inițială	6
Evaluare inițială	7

CAPITOLUL 1. CALCUL TABELAR (1.1.; 3.1.)

Lecția 1. Aplicația de calcul tabelar Microsoft Excel.	Serii de date	24
Operații cu registrul de calcul	Fișa de lucru 7	25
Fișa de lucru 1	Formule de calcul care utilizează	
Autoevaluare	operatori aritmetici (+ , - , * , /)	25
Lecția 2. Foi de calcul – operații cu foi de calcul,	Fișa de lucru 8	26
operații cu rânduri și coloane	Funcții specifice aplicației de calcul tabelar	26
Operații cu foi de calcul	Fișa de lucru 9	28
Fișa de lucru 2	Autoevaluare	29
Operații cu rânduri și coloane	Lecția 6. Reprezentarea grafică a datelor	30
Fișa de lucru 3	Fișa de lucru 10	32
Autoevaluare	Autoevaluare	32
Lecția 3. Operații de formatare a celulelor	Lecția 7. Pregătirea registrului pentru imprimare	33
Fișa de lucru 4	Fișa de lucru 11	34
Fișa de lucru 5	Autoevaluare	35
Autoevaluare	Proiecte	36
Lecția 4. Sortarea datelor	Evaluare sumativă	38
Fișa de lucru 6	Verifică-ți cunoștințele!	38
Autoevaluare	Investighează!	39
Lecția 5. Serii de date. Formule de calcul.	Portofoliu	39
Funcții specifice aplicației de calcul tabelar		

CAPITOLUL 2. PAGINI WEB (1.2.; 3.2.)

Lecția 8. Limbajul HTML.	Lecția 12. Inserarea hiperlegăturilor	52
Editoare de pagini web	Introducerea unei legături în cadrul	
Fișa de lucru 12	paginii web	52
Autoevaluare	Tipuri de legături în paginile web	52
Lecția 9. Crearea unei pagini web	Fișa de lucru 18	53
Elementele de structură ale unei pagini web:	Autoevaluare	54
antet, titlu, corp	Lecția 13. Inserarea și formatarea imaginilor	55
Fișa de lucru 13	Fișa de lucru 19	56
Fundalul unei pagini web	Autoevaluare	56
Fișa de lucru 14	Lecția 14. Inserarea și formatarea tabelelor	57
Autoevaluare	Fișa de lucru 20	59
Lecția 10. Titluri, paragrafe, texte în pagini web	Autoevaluare	59
Titluri	Lecția 15. Securitate cibernetică	60
Fișa de lucru 15	Fișa de lucru 21	61
Paragrafe	Autoevaluare	61
Texte	Investighează!	62
Fișa de lucru 16	Evaluare sumativă	62
Autoevaluare	Verifică-ți cunoștințele!	62
Lecția 11. Liste în pagini web	Proiect	63
Fișa de lucru 17	Portofoliu	63
Autoevaluare		

CAPITOLUL 3. ALGORITMI (2.1.; 2.2.; 3.3.)

Lecția 16. Limbaje de programare – noțiuni

<i>recapitulative</i>	64
<i>Fișa de lucru 22</i>	66
<i>Aplicații suplimentare</i>	67
<i>Autoevaluare</i>	67

Lecția 17. Șir de valori – noțiuni introductive.

<i>Operații cu șiruri de valori</i>	68
<i>Fișa de lucru 23</i>	69
<i>Operații cu șiruri de valori</i>	70
<i>Fișa de lucru 24</i>	72
<i>Autoevaluare</i>	72

Lecția 18. Algoritmi de bază pentru șiruri de valori – numărare, sumă/produs și maxim/minim

<i>Obținerea unei sume/produs utilizând elementele unui șir de valori</i>	73
<i>Determinarea valorii minime, respectiv maxime a unui șir</i>	73
<i>Probleme de numărare/frecvență</i>	75
<i>Fișa de lucru 25</i>	77
<i>Autoevaluare</i>	77

Lecția 19. Algoritmi de bază pentru șiruri de valori – verificare a unei proprietăți

<i>Verificarea unei proprietăți îndeplinită de elementele unui șir de valori</i>	78
<i>Verificarea existenței unor valori într-un șir</i>	79
<i>Verificarea unei proprietăți comune pentru toate elementele dintr-un șir de valori</i>	81
<i>Fișa de lucru 26</i>	83
<i>Autoevaluare</i>	83

Lecția 20. Codebot – controlul unui robot didactic

<i>virtual</i>	84
<i>Fișa de lucru 27</i>	88
<i>Aplicație suplimentară</i>	88
<i>Autoevaluare</i>	88

Evaluare sumativă

<i>Verifică-ți cunoștințele!</i>	89
<i>Investighează!</i>	89
<i>Proiect</i>	90
<i>Portofoliu</i>	90

<i>Aplicații suplimentare</i>	91
<i>Citirea unui șir de valori. Afișarea unor elemente ale șirului</i>	91
<i>Parcurgerea unui șir de valori</i>	91
<i>Generarea unui șir de valori</i>	92
<i>Sumă/Produs/Medie</i>	93
<i>Minim/Maxim</i>	93
<i>Numărare</i>	93
<i>Șiruri de frecvență</i>	94
<i>Verificarea unei proprietăți îndeplinită de elementele unui șir</i>	94
<i>Verificarea existenței unei valori într-un șir</i>	95
<i>Verificarea unei proprietăți comune pentru toate elementele unui șir</i>	95
<i>Robot didactic virtual</i>	96
<i>Recapitulare finală</i>	97
<i>Evaluare finală</i>	98
<i>Fișă de observare a comportamentului și activității elevului</i>	99
<i>Indicații și răspunsuri</i>	100

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 mini proiecte 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

Recapitulare inițială

Cele șapte minuni ale lumii antice



Cerință: Creați pe **Desktop** sau într-un loc din memoria externă unde aveți drept de scriere, un dosar numit **Recapitulare**. Salvați în acest dosar, toate fișierele create la recapitularea inițială.

Exercițiul 1. Scrieți un program în limbajul C++ pentru a rezolva următoarea problemă: Paul și Mădălina participă la o tombolă pentru a afla informații suplimentare despre una dintre cele 7 minuni ale lumii antice. Biletul lor ar putea fi câștigător dacă pe el ar fi scris un număr care are: un număr par de cifre, cea mai mare cifră impară și suma cifrelor un număr între 1 și 7. În cazul în care suma cifrelor este mai mare decât 9, se reia suma cifrelor pentru suma obținută, până când se ajunge la o valoare de o cifră. Ajutați-i pe cei doi să verifice dacă au extras un bilet câștigător și ce număr are minunea lumii despre care vor afla informații suplimentare (suma cifrelor). (Ex: 299871 – are 6 cifre, cifra maximă, 9, este impară, $2+9+9+8+7+1=36$ $3+6=9$ – număr necâștigător/ 2911 – are 4 cifre, cifra maximă impară $2+9+1+1=13$ $1+3=4$ – copiii vor afla informații despre cea de a patra minune a lumii)

(40p)



Exercițiul 2. Folosind aplicația **Microsoft Word**, creați un document, cu numele **Minune.docx**, care va conține o descriere a Colosului din Rodos. Elementele obligatorii de conținut sunt: fundal colorat, un titlu sugestiv, text explicativ și 2-4 imagini înconjurare de text.

(20p)

Exercițiul 3. Folosind aplicația **OpenShot Video Editor** realizați un fișier video în care prezentați 3 dintre cele șapte minuni ale lumii antice (titlu, text scurt, imagini). Adăugați o melodie potrivită în fundal. Salvați fișierul cu numele **Minuni.mp4**.

(20p)

Exercițiul 4. Folosind aplicația **Google Slides**, creați o prezentare care va conține obligatoriu: titlul **Minunile lumii antice**, imagini sugestive pentru celelalte 4 minuni neprezentate în exercițiul 3, și fișierul creat la exercițiul 3. La final, adăugați animații și tranziții, salvați prezentarea cu numele **Univers** și distribuiți-o unui coleg.

(20p)

Total: 100 puncte
Mult succes!

Barem de corectare:

1. **40p** – 2p – declararea variabilelor, 3p – utilizarea corectă a variabilelor, 3p – citirea datelor, 10p – determinarea numărului de cifre, 10p – determinarea cifrei maxime, 10p – calculul sumei cifrelor, 2p – afișarea rezultatului;
2. **20p** – 3p – nume fișier, 2p – fundal, 2p – titlu, 3p – text sugestiv, 5p – imagini, 3p – design, 2p – creativitate;
3. **20p** – 2p – nume fișier, 3p – melodie în fundal, 4p/ fiecare element prezentat (12p total), 3p – design și creativitate;
4. **20p** – 2p – nume fișier, 2p/ fiecare minune prezentată (8p total), 2p – animații, 2p – tranziții, 3p – există fișierul de la ex. 3, 3p – distribuirea.

Evaluare inițială

Cele șapte minuni ale lumii moderne



Cerință: Creați pe **Desktop** sau într-un loc din memoria externă unde aveți drept de scriere, un dosar numit **Evaluare**. Salvați în acest dosar, toate fișierele create la evaluarea inițială.

Exercițiul 1. Scrieți un program în limbajul C++ pentru a rezolva următoarea problemă: Mihai și Paula nu se pot decide pe care minune a lumii moderne să o viziteze. Pentru a determina numărul de ordine al acesteia au la dispoziție 2 numere cu mai multe cifre și trebuie să determine cifra minimă a primului număr, să o înmulțească cu suma cifrelor celui de al doilea număr și să numere câte cifre are valoarea obținută. Dacă numărul de cifre este între 1 și 7, numărul indică minunea care urmează a fi văzută, iar dacă este mai mare de 7 cei doi vor vizita toate minunile rând pe rând. (Ex: 3456 4590 – cifra minimă este 3. $3 \times (4+5+9+0) = 54$ – 2 cifre – copiii vizitează minunea nr. 2)



Exercițiul 2. Folosind aplicația **Microsoft Word**, creați un document, cu numele **Moderna.docx**, care va conține o descriere a orașului templu Machu-Picchu. Elementele obligatorii de conținut sunt: fundal particularizat la alegere, un titlu sugestiv, text explicativ și 2-4 imagini înconjurare de text. (20p)

Exercițiul 3. Folosind aplicația **OpenShot Video Editor** realizați un fișier video în care prezentați 3 dintre cele șapte minuni ale lumii moderne (titlu, text scurt, imagini, sunet). Adăugați efecte de tranziții și animații. Salvați fișierul cu numele **Minuni.mp4**. (25p)

Exercițiul 4. Folosind aplicația **Google Slides**, creați o prezentare care va conține obligatoriu: titlul **Minunile lumii moderne**, imagini sugestive pentru celelalte 4 minuni neprezentate în exercițiul 3, și fișierul creat la exercițiul 3. La final, adăugați animații și tranziții, salvați prezentarea cu numele **Univers** și distribuiți-o unui număr de 3 colegi și profesorului vostru. (15p)

Total: 100 puncte
Mult succes!

Barem de corectare:

- 40p** – 2p – declararea variabilelor, 3p – utilizarea corectă a variabilelor, 3p – citirea datelor, 10p – determinarea cifrei minime, 10p – calculul sumei cifrelor, 10p – determinarea numărului de cifre, 2p – afișarea rezultatului;
- 20p** – 3p – nume fișier, 2p – fundal, 2p – titlu, 3p – text sugestiv, 5p – imagini, 3p – design, 2p – creativitate;
- 25p** – 2p – nume fișier, 3p – melodie în fundal, 4p/ fiecare element prezentat (12p total), 5p – tranziții și animații, 3p – design și creativitate;
- 15p** – 1p – nume fișier, 2p/ fiecare minune prezentată (8p total), 1p – animații, 2p – tranziții, 1p – există fișierul de la ex. 3, 2p – distribuirea.

Capitolul 1. Calcul tabelar

Competențe: 1.1., 3.1.

În acest capitol vom învăța despre: ● Elemente de interfață ale unei aplicații de calcul tabelar ● Structura unui registru de calcul (foaie de calcul, coloană, rând, celulă, adresă de celulă) ● Operații cu registru de calcul (deschidere, închidere, salvare, creare) ● Operații cu foi de calcul (accesare, redenumire) ● Operații de editare (selectare, copiere, mutare, ștergere) ● Operații de formatare a rândurilor / coloanelor ● Operații de formatare a celulelor (aliniere conținut, culori de umplere, stiluri predefinite) ● Tipuri de date: numeric, text, dată calendaristică ● Sortarea crescătoare / descrescătoare a datelor dintr-un tabel după unul sau mai multe criterii ● Formule de calcul care utilizează operatori aritmetici (+, -, *, /) ● Funcții specifice aplicației de calcul tabelar pentru sumă, maxim, minim, medie aritmetică și decizie ● Grafice: tipuri de grafice ● Serii de date

Lecția 1. Aplicația de calcul tabelar Microsoft Excel. Operații cu registrul de calcul



Aplicațiile de calcul tabelar sunt programe specializate în realizarea de calcule ce implică volume relativ mari de date, organizate în tabele.



Microsoft Excel este o aplicație de calcul tabelar, face parte din pachetul **Microsoft Office** și este cel mai utilizat program de acest fel, în prezent. Fișierul Excel se numește **registru de calcul** și conține una sau mai multe foi, structurate sub formă de tabele.

Cu ajutorul aplicației de calcul tabelar **Microsoft Excel** se pot realiza:

- calcule utilizând formule simple sau complexe (de exemplu: mediile elevilor dintr-o clasă);
- tabele cu foarte multe date, interconectate;
- reprezentări grafice a datelor numerice din fișier (de exemplu: grafice cu mediile la diverse examene naționale ș.a.);
- calcule între diferite fișiere Excel sau fișiere de alt tip.



Știați că?

Prima versiune a unui program de calcul tabelar, și anume Excel 2.0, a fost lansată pe 31 octombrie 1987.

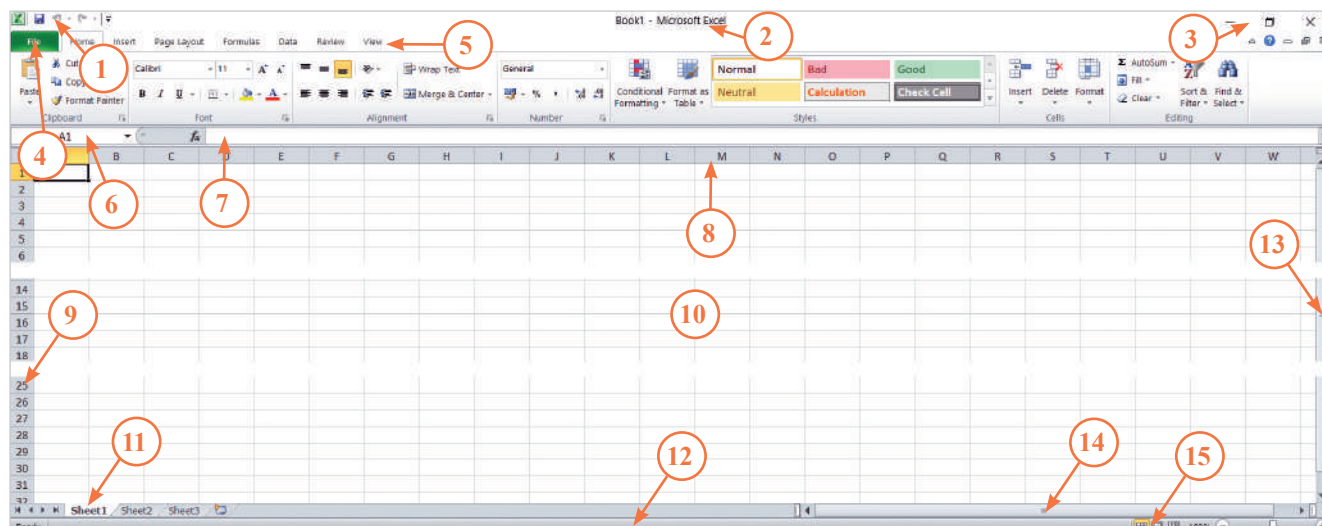
Deschiderea aplicației Microsoft Excel – se poate face astfel:

- În funcție de sistemul de operare și configurările acestuia, pe **Desktop** sau în meniul **Start** poate exista o pictogramă asociată programului.
- Se poate crea un fișier de tip **Microsoft Excel** și cu dublu click pe acesta se deschide aplicația.

Elemente de interfață grafică ale aplicației de calcul tabelar Microsoft Excel

În fereastra asociată aplicației Microsoft Excel se pot identifica următoarele elemente de interfață grafică:

- | | |
|--|--|
| 1 - Bara de instrumente Acces Rapid (Quick Access Toolbar) | 8 - Etichete coloană (litere care identifică, în mod unic, coloanele) |
| 2 - Bara de titlu (Title bar) | 9 - Etichete rând (numere care identifică, în mod unic, rândurile) |
| 3 - Butoane pentru minimizare, maximizare și închidere a ferestrei | 10 - Suprafață de lucru (sub formă de tabel) |
| 4 - Meniul File (Fișier) | 11 - Etichete (tab) asociate foilor de calcul |
| 5 - Panglica (ribbon) cu filele de lucru Home, Insert, Page Layout, Formulas, Data, Review, View ș.a. | 12 - Bara de stare (Status bar) |
| 6 - Bara de adrese | 13 - Bara de derulare verticală (Scroll bars) |
| 7 - Bara de formule (fx) | 14 - Bara de derulare orizontală |
| | 15 - Butoane de vizualizare a foilor de calcul |



Registrul de calcul – structura unui registru (foaie de calcul, coloană, rând, celulă, adresă de celulă)

Un **registru de calcul** (*workbook*) este un fișier cu extensia *xlsx* (*xls* în versiunile mai vechi) în care se pot stoca date, care pot fi prelucrate și formate ulterior prin diferite operații.

Un registru de calcul poate avea una sau mai multe **foi de calcul**. Fiecare foaie dintr-un registru are asociată o etichetă, vizibilă deasupra barei de stare. Foaia de calcul este configurată sub formă tabelară – **rânduri** (*row*), identificate prin numere și **coloane** (*column*), identificate prin litere. Dreptunghiul existent la intersecția unui rând cu o coloană este denumit **celulă** (*cell*).

EX. Exemplu: La intersecția dintre coloana B și rândul 2 vom avea celula B2. B2 este **adresa celulei** și aceasta apare scrisă în bara de adrese, în cazul în care celula este selectată. Dacă se scrie B12 în bara de adrese și se apasă **Enter**, este selectată automat celula cu adresa respectivă.

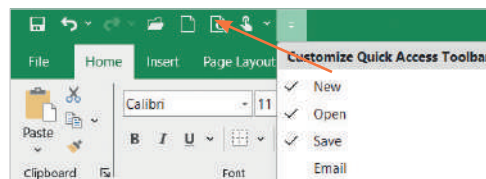
	A	B	C
1			
2		B2	
3			



O foaie de calcul Excel poate conține: text, numere, imagini, grafice, animații, link-uri etc!

Liniile de grilă, etichetele pentru rânduri și coloane, rigla și bara de formule se pot afișa/ascunde utilizând opțiunile corespunzătoare de pe fila **View**.

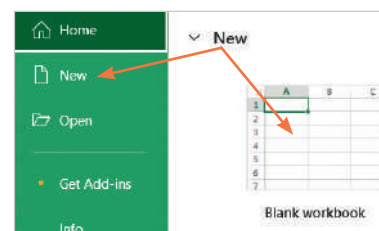
Operații cu registrul de calcul

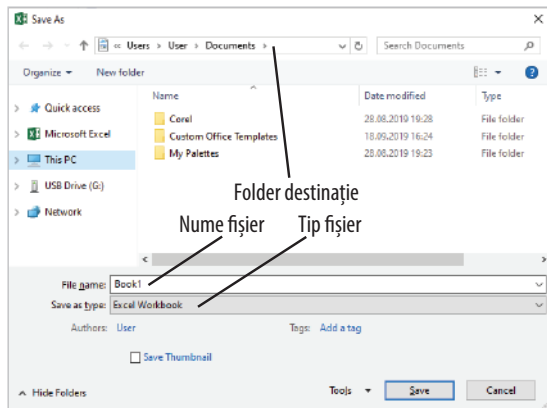


Instrumentele/butoanele care permit realizarea operațiilor elementare cu registre de calcul se găsesc în meniul **File** (**Fișier**) sau pe bara de instrumente **Acces Rapid**. Aceste operații sunt:

1. **Crearea unui registru de calcul nou** se realizează utilizând opțiunea **New** (**Nou**) (**CTRL + N**).

2. **Deschiderea unui registru de calcul existent** se face cu ajutorul opțiunii **Open** (**Deschidere**) (**CTRL+O**). Un registru utilizat recent, se poate găsi și în meniul **File**, secțiunea **Recent**.





3. Salvarea unui registru – pentru salvarea registrului se pot folosi opțiunile **Save (Salvare)** și **Save As (Salvare Ca)** (**CTRL + S – Save, F12 – Save As**).

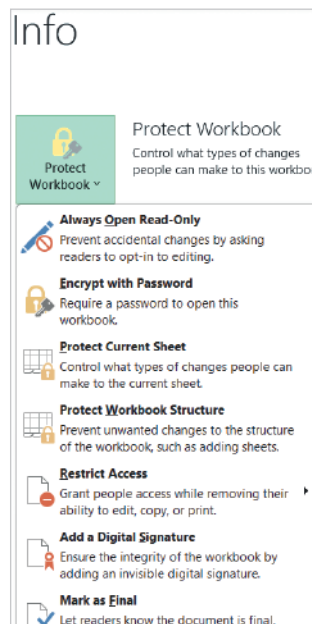
Aplicația **Excel** permite salvarea automată a datelor, după un anumit interval stabilit (meniul **File – Options – Save**).

4. Modificarea proprietăților unui registru de lucru

Meniul **File – Info**, secțiunea

Properties permite modificarea unor proprietăți precum titlul, autorul ș.a. Butonul **Protect Workbook**, din același meniu, permite parolarea fișierului (**Encrypt with Password**), restricționarea accesului la registru sau foile sale de calcul, marcarea documentului ca finalizat – read only (**Mark as Final**) ș.a.

5. Închiderea unui fișier Excel, respectiv a ferestrei asociată aplicației se realizează cu opțiunea **Close** din meniul **File**, respectiv butonul **Close** situat în partea dreaptă a barei de titlu.



Rețineți!

- 📖 Microsoft Excel este o aplicație de calcul tabelar.
- 📖 Fișierul Excel se numește registru de lucru (workbook) și are extensia xls sau.xlsx.
- 📖 Un registru de lucru poate conține una sau mai multe foi de calcul.



Fișa de lucru 1

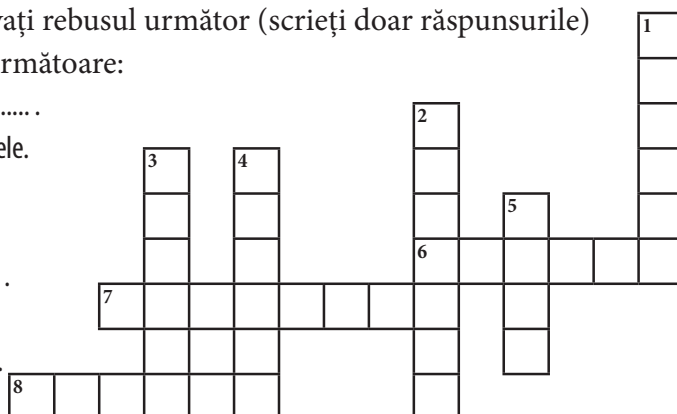
Creați un dosar numit **Capitol1**. În acesta veți salva toate fișierele realizate pe măsură ce parcurgeți acest capitol. Pentru fiecare fișă de lucru parcursă, creați, în dosarul **Capitol1**, un dosar cu numele **FisaNr**, unde Nr. va fi numărul de ordine al fișei de lucru. **Dosarul Capitol1 va face parte din portofoliul vostru personal.**



Cerință: Creați dosarul numit **Fisa1** și rezolvați exercițiile de mai jos salvând în acest dosar toate fișierele realizate pe parcurs.

1. ()** Creați un document **rebus.docx** și rezolvați rebusul următor (scrieți doar răspunsurile) pentru a identifica cuvintele lipsă din enunțurile următoare:

4. Aplicația **Microsoft Excel** face parte din pachetul **Microsoft**
2. **Microsoft Excel** este specializată în realizarea de în tabele.
7. Fișierul **Excel** se numește de lucru.
5. Extensia implicită a fișierului **Excel** este
8. Coloanele dintr-o foaie de calcul sunt identificate prin
3. Rândurile unei foi de calcul sunt identificate prin
6. La intersecția dintre un rând și o coloană se găsește o
1. B12 este un exemplu de de celulă.



2. Scrieți într-un fișier **raspunsuri.docx** trei situații în care este utilizat un registru de lucru Excel pentru a organiza și gestiona date.

3. (*) Deschideți aplicația **Microsoft Excel** și realizați următoarele cerințe:

a. Identificați principalele elemente de interfață grafică a unei foi de calcul și scrieți-le în celulele A1, A2, (Pentru a scrie în celule se dă click pe acestea și se tastează textul. Acesta va depăși probabil marginile celulei, dar nu e o problemă.)

b. Salvați fișierul cu numele **interfata.xls**.

c. Salvați fișierul cu numele **interfata_grafica.xlsx**.

d. Închideți fișierul, fără a închide și fereastra principală a aplicației **Microsoft Excel**.

4. (***) Deschideți fișierul **interfata.xls** și realizați următoarele cerințe:

a. Modificați proprietățile fișierului astfel: Title (Titlu) – **Registru1**, Tags (Etichete) – **Excel**, Comments (Comentarii) – **Acesta este primul meu registru**, Subject (Subiect) – **Proprietăți registru**, Author (Autor) – **Numele vostru**.

b. Realizați o captură a ferestrei în care ați modificat proprietățile și salvați-o cu numele **proprietati.jpg**.

c. Parolați registrul de lucru. Nu uitați că aveți nevoie de parolă pentru a-l redeschide.

d. Marcați registrul de lucru ca fiind finalizat (Read only). Salvați fișierul.



Autoevaluare

Dragi elevi, la finalul fiecărei lecții vă propunem o scurtă autoevaluare a cunoștințelor acumulate!

În acest sens, alegeți pentru fiecare element din tabel răspunsul potrivit (*da, nu sau nu sunt sigur*) și completați enunțul de la final cu ce dificultăți ați întâmpinat. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da, 1 – nu, 1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa1**.

	Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...			
1. ce este o aplicație de calcul tabelar.			
2. care sunt elementele de interfață grafică ale aplicației Excel.			
3. să creez un registru de calcul nou.			
4. să salvez un registru de lucru.			
5. să deschid/închid un registru de lucru.			
6. să modific proprietățile unui registru.			

7. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!

(Pentru aceasta aveți la dispoziție, la pagina 99, o fișă de observare a comportamentului și activității!)

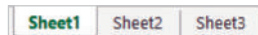
La finalul lecției, m-am simțit:



Lecția 2. Foi de calcul – operații cu foi de calcul, operații cu rânduri și coloane

Operații cu foi de calcul

1. **Accesarea unei foi de calcul** se face printr-un click pe eticheta asociată foi.



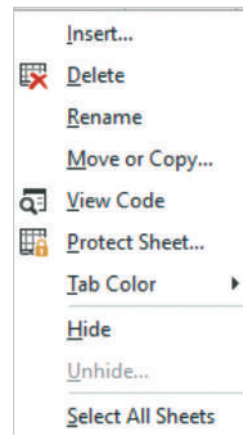
2. **Redenumirea unei foi de calcul**

Fiecare foaie din registrul de calcul are un nume unic, vizibil pe tab-ul (eticheta) asociat foi.

Redenumirea foi se face utilizând opțiunea **Rename (Redenumire)** din meniul contextual asociat foi sau utilizând opțiunea **Format (Formatare) → Rename Sheet (Redenumire Foaie)** de pe fila **Home (Pornire)**.

3. **Inserarea unei noi foi de calcul** se poate face utilizând:

- butonul **New sheet (Foaie nouă)** , existent lângă etichetele foilor de calcul;
- opțiunea **Insert (Inserare)** din meniul contextual, asociat etichetei oricărei foi din registru;
- meniul **Insert (Inserare) → Insert Sheet (Inserare Foaie)** de pe fila **Home (Pornire)**.

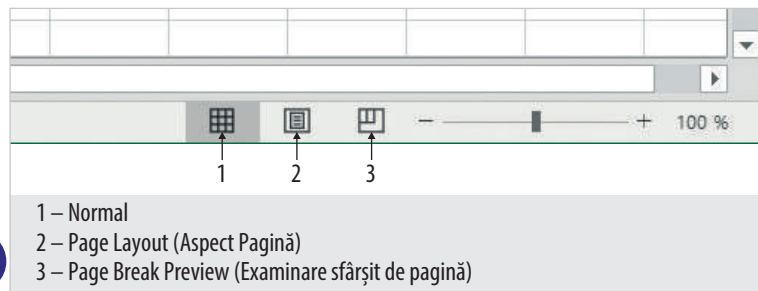
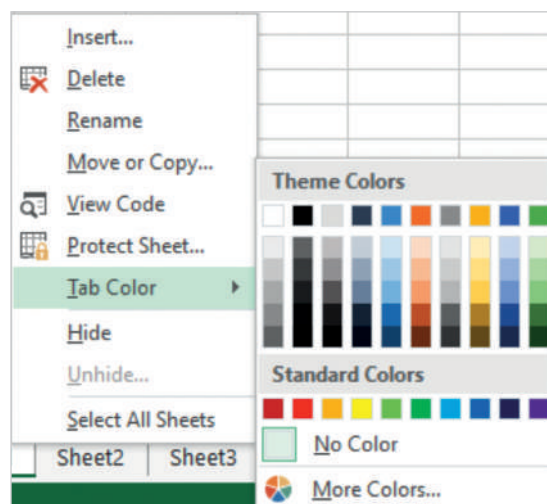


4. **Colorarea etichetei asociate foi de calcul** este benefică pentru o identificare ușoară a foi și se realizează utilizând opțiunea **Tab color**, fie din meniul contextual, fie din meniul **Format (Formatare)**, fila **Home (Pornire)**.

5. **Ascunderea și reafișarea unei/ unor foi de calcul** se pot face cu ajutorul opțiunilor **Hide (Ascundere)** și **Unhide (Afișare)**, existente în meniul contextual sau în meniul **Format**.

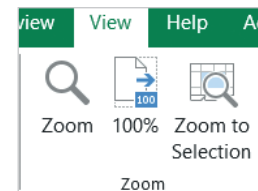
6. **Modificarea modului de vizualizare a foi de calcul**

Foile de calcul pot fi vizualizate în următoarele moduri: **Normal**, **Page Layout (Aspect Pagină)** și **Page Break Preview (Examinare sfârșit de pagină)**.



Modul de vizualizare **Normal** permite vizualizarea strictă a conținutului foi de calcul și este cel mai recomandat în lucrul cu date multe. Modul de vizualizare **Page Layout** prezintă conținutul foi de calcul, împărțit în pagini și oferă posibilitate gestionării ușoare a conținutului paginilor, respectiv a antetului și subsolului acestora.

Apropierea/îndepărtarea foi de calcul poate fi realizată cu ajutorul opțiunii de **Panoramare (Zoom)** existentă pe bara de stare sau pe fila **View (Vizualizare)**.

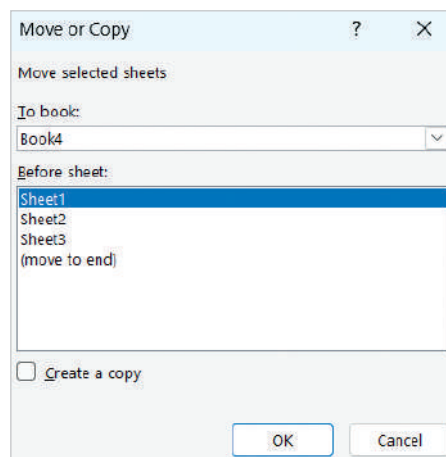


Operații de editare

7. Selectarea unor foi de calcul se face dând click pe etichetele asociate foilor, în combinație cu tasta CTRL, respectiv Shift. Pentru a selecta toate foile din registru se poate utiliza opțiunea **Select All Sheets** (**Selectează Toate Foile**), din meniul contextual asociat oricărei foi din registru.

8. Copierea/mutarea unei foi de calcul în același registru, într-un registru existent deja și deschis sau într-un registru nou, se face utilizând opțiunea **Move or Copy** (**Mutare sau Copiere**). Această opțiune se găsește în meniul contextual asociat foi sau pe fila **Home** (**Pornire**), în meniul **Format** (**Formatare**).

Fereastra **Move or Copy** permite alegerea registrului în care se va muta/copia foaia (**To book**) și stabilirea poziției foi, relativ la foile existente deja în registrul în care se va muta/copia (**Before sheet**). Pentru ca foaia să fie copiată, nu mutată, se bifează opțiunea **Create a copy**. Mutarea unei foi de calcul în fața sau după altă foaie se poate face și prin glisarea etichetei foi selectate cu ajutorul mouse-ului (drag and drop).



9. Ștergerea unei foi de calcul se face utilizând opțiunea **Delete** (**Ștergere**) din meniul contextual sau opțiunea **Delete** (**Ștergere**) → **Delete Sheet** (**Ștergere Foaie**), de pe fila **Home** (**Pornire**).



Fișa de lucru 2

Creați, în dosarul **Capitol1**, dosarul numit **Fisa2** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. () Creați un registru de calcul nou, salvați-l cu numele *operatii_foi.xlsx* și realizați cerințele următoare:**

- Inserați foi de calcul, astfel încât registrul de lucru să conțină 5 foi.
- Redenumiți foile de calcul astfel A, B, C, D, E.
- Colorați diferit etichetele asociate celor 5 foi.
- Mutați foaia A în dreapta foi C. Ștergeți foaia D. Ascundeți foaia E.
- Salvați fișierul.

2. (*) Creați un registru de calcul nou, salvați-l cu numele *date_elevi.xlsx* și realizați cerințele următoare:**

a. Realizați operațiile necesare astfel încât registrul de calcul să conțină trei foi, numite **Date_elevi**, **Medii** și **Absențe**.

b. Colorați diferit etichetele asociate celor trei foi de calcul.

c. Copiați din registrul *operatii_foi.xlsx* foaia A, după foaia **Absențe**.

d. Ascundeți foaia A, copiată anterior.

e. Modificați procentul de panoramare a foi **Date_elevi** la 115%, realizați o captură a ferestrei utilizate și salvați-o cu numele *zoom.jpg*.

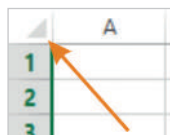
f. Scrieți în celula A1 a foi **A** (cea ascunsă), textul *Microsoft Excel* și modificați modul de vizualizare a paginii în **Print Layout** (**Aspect pagină imprimată**).

g. Salvați fișierul.

Operații cu rânduri și coloane

Operații de editare

1. Selectarea foii, selectarea de rânduri sau coloane

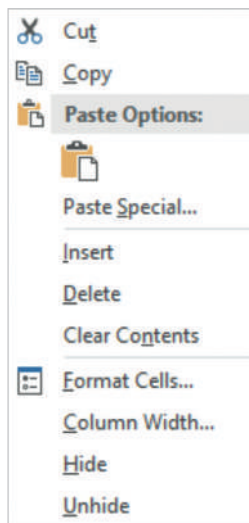


Pentru a selecta întreg conținutul foii de calcul se poate utiliza combinația de taste **CTRL+A** sau se dă click pe butonul existent între eticheta asociată coloanei A și cea asociată rândului 1.

Selectarea unui rând sau a unei coloane se face dând click pe eticheta asociată. Pentru a selecta mai multe rânduri și/sau coloane neadiacente se dă click pe etichetele asociate acestora, apăsând simultan tasta **CTRL**.

2. Copierea, mutarea, lipirea unui rând sau coloană selectat(ă) se face utilizând opțiunile **Cut** (**Decupare**), **Copy** (**Copiere**) și **Paste** (**Lipire**) de pe fila **Home** (**Pornire**) sau din meniul contextual. Opțiunea **Paste** (**Lipire**) înlocuiește conținuturile existente pe rândul/coloana unde se face lipirea, cu cele copiate (decupate). Pentru a insera rândul/coloana copiată/decupată și a păstra și datele existente deja în tabel se pot utiliza opțiunile **Insert Copied Cells** (**Inserare Celule Copiate**), respectiv **Insert Cut Cells** (**Inserare Celule Decupate**). Rândul copiat va apărea deasupra, coloana copiată va apărea la stânga celei selectate.

3. Ștergerea unui rând sau a unei coloane selectate se poate face utilizând tasta **Delete**, opțiunea **Delete** (**Ștergere**) din meniul contextual sau opțiunile corespunzătoare din meniul **Delete** (**Ștergere**) de pe fila **Home** (**Pornire**). Odată cu ștergerea unor rânduri, respectiv coloane, rândurile și coloane existente se vor rearanja, modificându-și automat etichetele asociate.

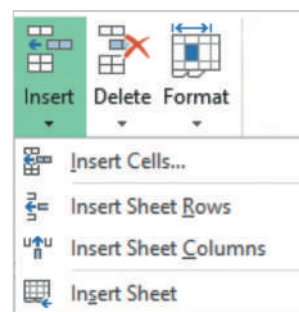
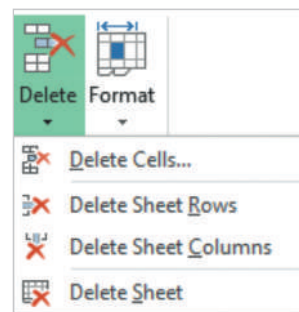


4. Inserarea unui rând sau a unei coloane se poate face utilizând opțiunea **Insert** din meniul contextual asociat etichetei oricărui rând sau oricărei coloane (adică cu click dreapta pe etichete) sau utilizând opțiunile corespunzătoare din meniul **Insert** (**Inserare**), de pe fila **Home** (**Pornire**).

Rândul nou va apărea mereu deasupra rândului curent (unde este plasat cursorul sau pe al cărei etichetă s-a dat click dreapta), iar coloana va apărea mereu în partea stângă a coloanei curente.

5. Golirea conținutului unui rând sau a unei coloane presupune ștergerea conținutului tuturor celulelor de pe rândul sau coloana respectivă și se realizează utilizând opțiunea **Clear Contents** (**Golire Conținuturi**), din meniul contextual.

? Știați că?
O foaie de calcul Excel conține 1.048.576 de rânduri și 16.384 de coloane.



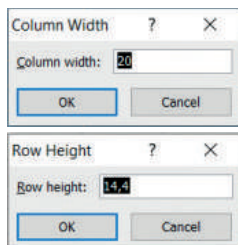
Operații de formatare

6. Stabilirea dimensiunii unui rând sau a unei coloane



Știați că?

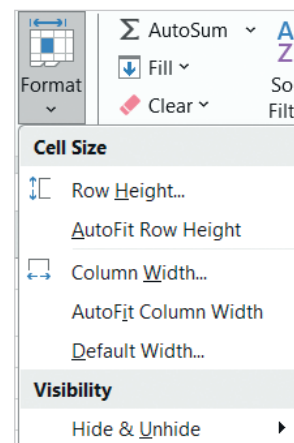
Înălțimea unui rând se măsoară în puncte (max. 409 points), iar lățimea unei coloane în numărul de caractere (max. 255 de caractere).



Înălțimea unui rând, respectiv lățimea unei coloane se pot modifica cu ajutorul opțiunilor **Row Height** (**Înălțime rând**), respectiv **Column Width** (**Lățime coloană**), existente în meniul contextual sau în meniul **Format (Formatare)**, de pe fila **Home**. Cele două mărimi se pot modifica și glisând cu ajutorul cursorului marginile etichetelor asociate rândului, respectiv coloanei selectate.

Pentru a potrivi dimensiunile rândului sau ale coloanei la conținut (adică pentru a modifica dimensiunile astfel încât să se ajusteze automat la conținut și acesta să încapă) se poate da dublu click pe marginea de jos a etichetei asociate rândului, respectiv marginea dreaptă a etichetei asociate coloanei sau se pot folosi opțiunile **AutoFit Row Height** (**AutoPotrivire Înălțime Rând**) și **AutoFit Column Width** (**AutoPotrivire Lățime Coloană**), din meniul **Format (Formatare)**.

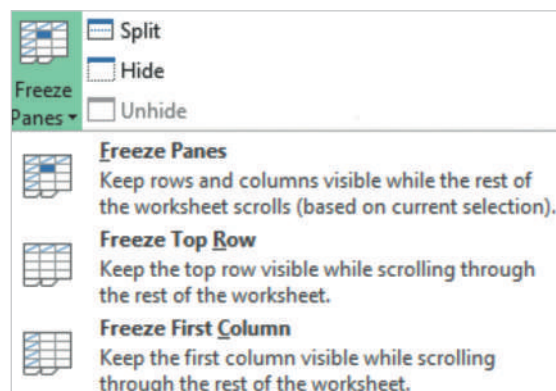
7. Ascunderea/reafișarea unui rând sau a unei coloane se face cu ajutorul opțiunilor **Hide** (**Ascundere**), respectiv **Unhide** (**Afișare**) existente în meniul contextual, sau în meniul **Format (Formatare)**. Pentru a reafișa rândurile sau coloanele ascunse, se selectează atât rândurile de deasupra cât și cele de sub rândurile ascunse, respectiv coloanele din stânga și din dreapta coloanelor ascunse.



8. Înghețarea unor rânduri/coloane

Există situații în care, prelucrând un volum mare de date, utilizatorul are nevoie să vadă anumite rânduri, din partea de sus a foii, sau coloane din partea stângă, pe măsură ce derulează foaia de calcul. Aplicația **Microsoft Excel** face posibil acest lucru, prin instrumentele existente în meniul **Freeze Panes** (**Înghețare Panouri**) de pe fila **View (Vizualizare)**.

Opțiunile **Freeze Top Row** și **Freeze First Column** permit vizualizarea în permanență a primului rând, respectiv a primei coloanei (A). Pentru a vizualiza mai multe rânduri sau mai multe coloane, se selectează rândul situat imediat sub acestea, respectiv coloana situată în dreapta celor care se vor a fi văzute și se utilizează opțiunea **Freeze Panes**.



Exemplu: Pentru a vedea în permanență primele 4 rânduri, se selectează rândul 5 și se apasă opțiunea **Freeze Panes**. Pentru a vedea coloanele A, B și C, se selectează coloana D și se apasă aceeași opțiune. Pentru a bloca la derulare primele 2 rânduri și coloanele A, B, C se selectează celula D3, după care se apasă **Freeze Panes**. Pentru a anula comanda de înghețare rânduri sau coloane se utilizează comanda **View – Unfreeze Panes**.



Rețineți!

- Foaia de calcul are o etichetă asociată, etichetă care conține numele foii.
- Principalele operații cu foi de calcul se pot accesa din meniul contextual asociat etichetei foii.
- Principalele operații cu rânduri și coloane se pot realiza utilizând opțiunile din meniul contextual asociat etichetelor acestora.



Fișa de lucru 3

Creați, în dosarul **Capitol1**, dosarul numit **Fisa3** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

(**) Deschideți registrul de calcul **operatii_foi.xlsx** din dosarul **Fisa2** și realizați cerințele următoare:

a. În foaia de calcul **B**, stabiliți înălțimea primelor 5 rânduri la 25pt și lățimea primelor 4 coloane de 20 (caractere).

b. Scrieți în celula A1 textul *Fișa de lucru 1*, în celula A2 textul *Fișa de lucru 3*, iar în celula A3 textul *Fișa de lucru 4*.

c. Inserați un rând nou sub rândul 1 și scrieți în celula A2 textul *Fișa de lucru 2*.

d. Copiați conținutul coloanei A și lipiți-l în coloana C. Ștergeți coloana B.

e. Ascundeți rândul 2.

f. Ștergeți textul *Fișa de lucru 2*, din ambele celule în care apare.

(***)

g. În foaia de calcul **C**, în celula A1, scrieți textul *Microsoft Excel – calcul tabelar*.

h. Ajustați lățimea coloanei A, astfel încât aceasta să fie potrivită automat la conținut.

i. Faceți setările necesare astfel încât primele 3 rânduri să rămână mereu vizibile în cazul derulării conținutului în jos.

j. În foaia de calcul **A** blocați la derulare primele 2 rânduri și prima coloană.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa3**.

	Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...			
1. să inserez o foaie de calcul.			
2. să șterg și să redenumesc o foaie de calcul.			
3. să copiez/mut o foaie în alt registrul.			
4. să ascund o foaie de calcul.			
5. să inserez un rând nou și o coloană.			
6. să modific înălțimea unui rând.			
7. să modific lățimea unei coloane.			
8. să copiez conținutul unui rând/coloană.			
9. să mut conținutul unui rând/coloană.			
10. să blochez la derulare anumite rânduri/coloane.			

11. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 3. Operații de formatare a celulelor

Celulele din foaia de calcul pot conține text, date numerice, date calendaristice, imagini, formule etc. Acestea pot fi introduse de la tastatură sau copiate din alte fișiere.

Pentru a introduce text în celulă, se dă click pe aceasta și se tastează textul sau se inserează un text copiat.

- Textul poate fi scris și în bara de formule, acesta fiind introdus automat în celula activă (selectată).
- Pe parcursul introducerii textului în celulă, o parte din instrumentele aplicației sunt inactive. Rămân active strict instrumentele care permit formatarea textului.

- Introducerea textului în celulă se finalizează la apăsarea tastei **Enter**.
- Dacă într-o celulă apare simbolul **#####**, acesta indică faptul că lățimea celulei este insuficientă și se impune modificarea acesteia.

Știați că?

Pentru a scrie text pe rânduri diferite, în aceeași celulă se folosește combinația de taste **ALT + Enter**.

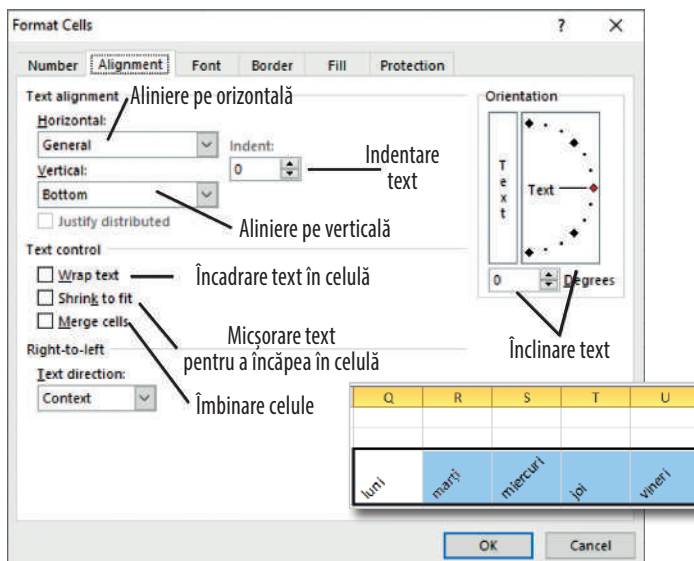
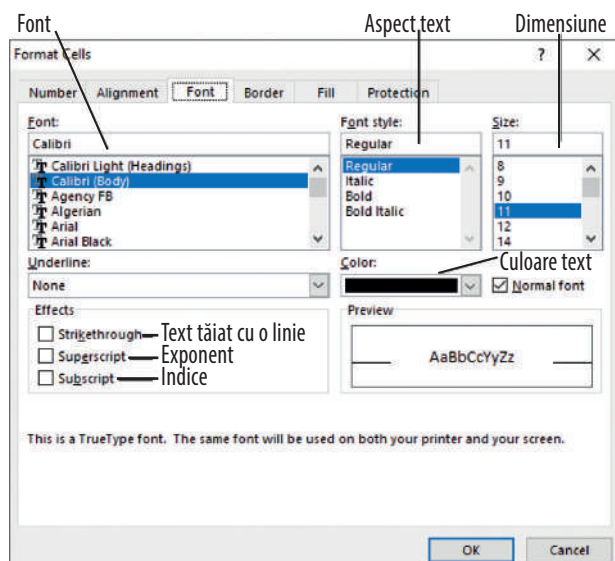
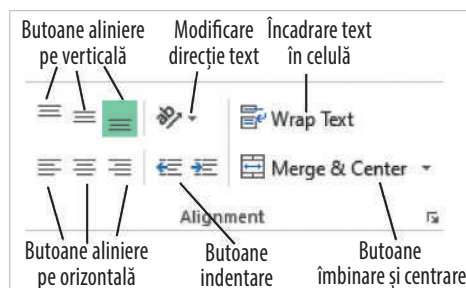
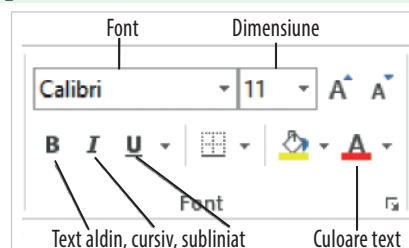
Formatarea celulei se referă la operațiile care se realizează la nivel de text, paragraf și celulă propriu-zisă pentru a modifica aspectul acestor elemente.

Celulele care trebuie formate se selectează, după care se accesează fereastra **Format Cells** fie folosind lansatorii grupurilor de butoane **Font**, **Alignment** (**Aliniere**), **Number** (**Număr**), fie folosind opțiunea **Format Cells** (**Formatare Celule**) din meniul contextual sau din meniul **Format** (**Formatare**) de pe fila **Home** (**Pornire**).

1. Formatarea textului se realizează cu ajutorul instrumentelor din grupul de butoane **Font** sau cu cele din fereastra **Format Cells**, fila **Font**.

2. Alinierea textului pe orizontală sau verticală, **indentarea** acestuia se realizează utilizând butoanele de pe fila **Home** (**Pornire**), grupul **Alignment** (**Aliniere**) sau cele din fereastra **Format Cells**, fila **Alignment** (**Aliniere**).

Modul de utilizare a instrumentelor amintite anterior se poate vedea în imaginile alăturate.



3. Încadrarea textului într-o celulă se face cu ajutorul opțiunilor:

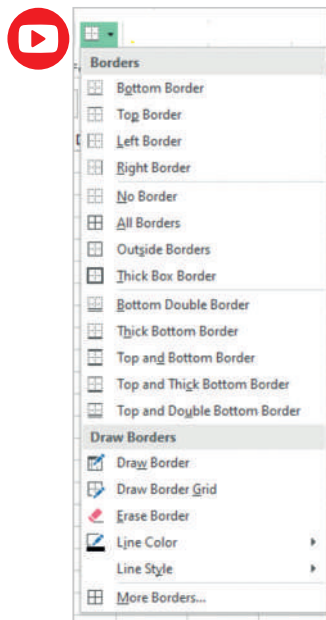
- **Wrap Text** (*Încadrare text*) - textul va apărea integral în celula selectată, modificându-se înălțimea acesteia.
- **Shrink to fit** (*Potrivire prin reducere*) – textul se va micșora astfel încât să încapă în celulă.

	A	B
	Text încadrat în celulă	
1		
2		Text potrivit prin reducere

4. **Îmbinarea celulelor și centrarea conținutului** se realizează cu ajutorul opțiunilor din meniul **Merge & Center** (*Îmbină și Centrează*). Celulele îmbinate vor fi unite într-o singură celulă, care preia adresa primei celule îmbinate.

EX. Exemplu: Un tabel cu 4 coloane și 2 rânduri - titlul tabelului va fi scris într-o singură celulă care să acopere lățimea întregului tabel. Se selectează celulele din rândul respectiv (4 celule) după care se alege opțiunea Merge & Center. Cele 4 celule vor fi unite într-una singură, care va prelua adresa primei celule, iar textul introdus va fi aliniat în centrul celulei. (Celulele B2, C2, D2, E2 au fost transformate într-o singură celulă care va avea adresa B2.)

	A	B	C	D	E
1					
2		ORAR			
3		Matematică	Fizică	Chimie	Biologie
4					

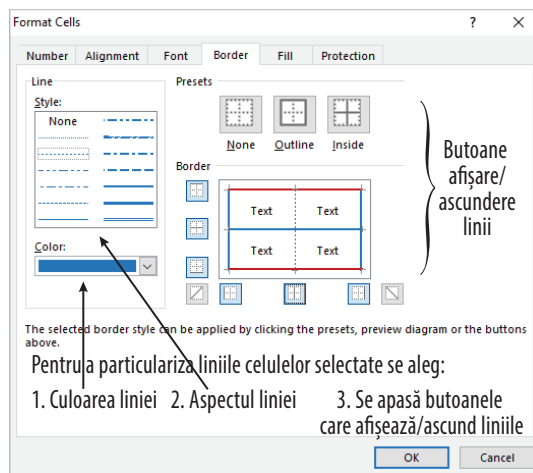


5. **Aplicarea și formatarea bordurilor (liniilor de contur) unei celule sau a unui grup de celule selectate** se realizează utilizând fila **Border** (*Borduri*) din fereastra **Format Cells** sau meniul **Borders**, de pe fila **Home** (*Pornire*), din grupul de butoane **Font**.

Modul de utilizare a instrumentelor de pe fila **Border** (*Borduri*) poate fi văzut în imaginea alăturată.

EX. În imaginea de mai jos se văd celule cu și fără linii de contur.

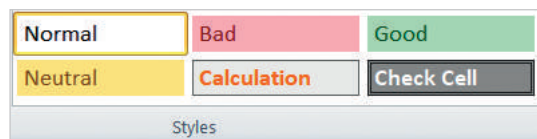
	A	B	C	D	E	F
1						
2						
3						
4						
5						
6						



6. **Particularizarea fundalului unei celule/ unor celule selectate** se poate face cu o culoare, cu un gradient sau un model (pattern), cu ajutorul instrumentelor existente pe fila **Fill** (*Umplere*) din fereastra **Format Cells** (*Formatare Celule*) sau cu ajutorul instrumentului **Fill color** (*Culoare de umplere*), de pe fila **Home** (*Pornire*), grupul de butoane **Font**.

7. Aplicarea stilurilor predefinite de formatare unor celule selectate se realizează utilizând

stilul dorit din grupul de butoane **Style** (*Stiluri*) de pe fila **Home** (*Pornire*). Stilurile existente pot fi modificate (click dreapta - *Modify*) și se pot crea stiluri noi de formatare.



Știați că? Microsoft Excel permite introducerea de comentarii atașate fiecărei celule.



Fișa de lucru 4

Creați, în dosarul **Capitol1**, dosarul numit **Fisa4** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (**) Creați un registru de calcul nou, denumiți-l **exercitiu.xlsx** și realizați în prima foaie de lucru din acesta tabelul alăturat.

Respectați formataările (alinieri, culori și aspect texte, aspect linii de grilă). Textul din tabel are fontul *Book Antiqua*, dimensiune 11.

	A	B	C
1	CURS VALUTAR		
2	Euro	Dolar	Liră sterlină
3	4,7794	4,2927	5,5717

2. (***) Creați un registru de calcul nou, salvați-l cu numele **angajati.xlsx** și realizați cerințele următoare:

- Păstrați o singură foaie de calcul în registru (dacă sunt mai multe) și dați-i numele **angajati**.
- Începând cu celula A1 creați tabelul de mai jos. Detalii de formatare: tot textul – font *Garamond*, dimensiune 12pt, bold (aldin) pentru datele din primul rând, 11pt în rest.
- Colorați diferit numele și prenumele celor 4 persoane.
- Inserați deasupra rândului 1 două rânduri libere.
- Îmbinați celulele A1, B1, ..., F1 și centrați conținutul celei obținute. Scrieți în celulă textul **DATE ANGAJAȚI**.
- Colorați fundalul celei cu o culoare la alegere și particularizați linia de la baza celei astfel încât să fie dublă, albastră.

	A	B	C	D	E	F
1	Nume	Prenume	Vârsta	Data_nașterii	Funcția	Salariu
2	Alb	Diana	40	20.02.1980	Asistent manager	3400
3	Abrudan	Paul	32	11.03.1988	Sales manager	4000
4	Costin	Alina	20	01.05.2000	Programator	4300
5	Popescu	Mircea	50	08.06.1970	Programator	4300

8. Tipuri de date

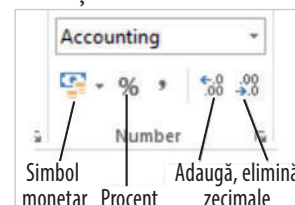
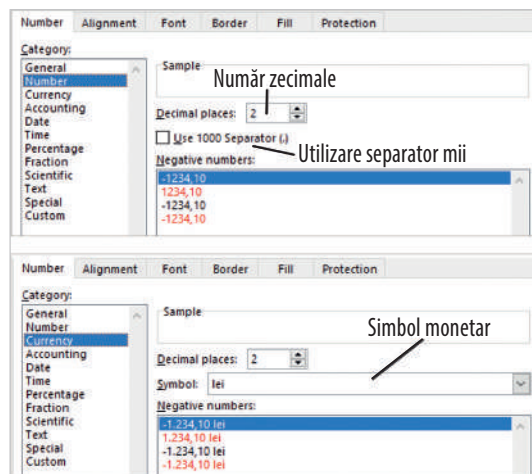
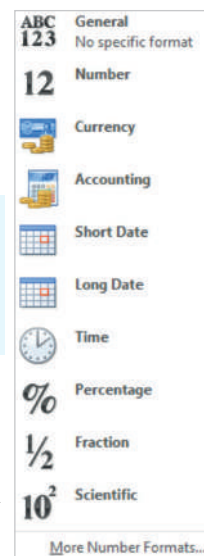
În celulele unei foi de calcul **Excel** pot exista date având diverse tipuri: text, numeric, dată calendaristică, oră, monedă, procent, științific ș.a. Acest lucru facilitează realizarea de calcule cu datele din tabele.

 Aplicația **Microsoft Excel** recunoaște diverse tipuri de date. Numerele și datele calendaristice scrise în celule, de exemplu, sunt aliniate automat în partea dreaptă a celei. Dacă nu se întâmplă acest lucru, înseamnă că nu este respectat formatul pe care îl au valorile numerice sau de tip dată calendaristică.

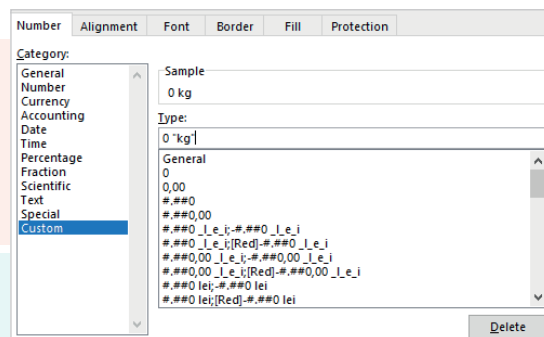
Pentru a stabili tipul de date din diverse celule selectate, se poate utiliza meniul existent

pe fila **Home (Pornire)**, grupul de butoane **Number (Număr)** sau instrumentele de pe fila cu același nume, din fereastra **Format Cells (Formatare Celule)**. În funcție de tipul de date ales, se pot stabili diverse proprietăți ale datelor. De exemplu, pentru datele de tip numeric sau monedă se pot stabili: numărul de zecimale, simbolul monetar utilizat ș.a.

Opțiunea **Custom (Particular)** permite crearea de tipuri de date particularizate, bazate pe un șablon existent (număr întreg – 0; număr cu 2 zecimale 0,00; procent număr întreg 0% ș.a).



EX. **Exemplu:** Pentru a adăuga în celule care conțin numere naturale textul *kg* imediat după numere și acestea să fie recunoscute de aplicație ca numere, se selectează șablonul **0**, după care se scrie textul “kg”, în caseta **Type**. În secțiunea **Sample** se poate vedea rezultatul.



Fișa de lucru 5

Creați, în dosarul **Capitol1**, dosarul numit **Fisa5** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (**) Deschideți registrul de calcul **exercitiu.xlsx** (Fisa 4) și realizați următoarele cerințe:
 - a. Inșerați deasupra rândului 3 un rând nou și scrieți în celulele A3, B3, și C3 valoarea 1.
 - b. Stabiliți pentru celulele A3, B3 și C3 tipul de dată monedă, simbolurile monetare fiind cele pentru dolar, euro și liră sterlină.
 - c. Stabiliți pentru datele de pe rândul 4 tipul de date Monedă – lei. Fiecare număr va avea 4 zecimale.
2. (***) Deschideți registrul de calcul **angajati.xlsx** creat anterior și realizați cerințele următoare:
 - a. Stabiliți pentru datele din tabel existente pe coloanele A, B și E tipul de date *Text*.
 - b. Stabiliți pentru datele numerice de pe coloana F tipul de date *Monedă (Currency)* – RON, fără zecimale.
 - c. Alegeți pentru datele calendaristice din tabel tipul *Data calendaristică*, format la alegere.
 - d. Creați un tip particular de date pentru datele numerice de pe coloana C, astfel încât acestea să fie urmate de textul *ani* (40 ani, 32 ani etc.).



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da*, *1 – nu*, *1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa5**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. să introduc text într-o celulă.			
2. să formatez textul din celule.			
3. să alinez conținutul unei celule.			
4. să încadrez textul în celulă.			
5. să îmbin mai multe celule.			
6. să particularizez liniile de grilă ale celulelor selectate.			
7. să particularizez fundalul celulelor.			
8. să aplic stiluri predefinite de formatare.			
9. să aleg tipurile de date potrivite pentru valorile din celule.			
10. să creez un tip particular de date.			

11. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



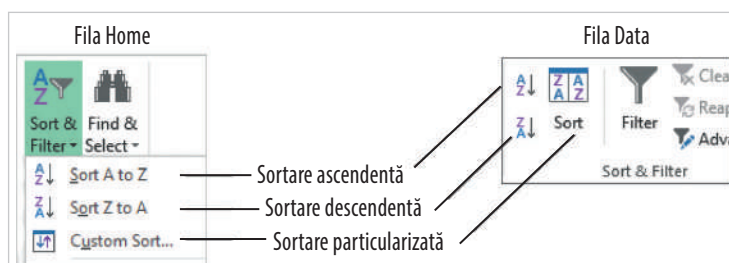
Lecția 4. Sortarea datelor

Pe parcursul prelucrării unui volum mare de date din foile de calcul Excel, există situații în care este nevoie de identificarea/găsirea rapidă a anumitor valori, de aranjarea acestora.

Sortarea este operația prin care datele din tabele sunt aranjate în funcție de valorile din anumite coloane, crescător sau descrescător.

Instrumentele utilizate pentru sortarea datelor se găsesc pe fila **Home** (**Pornire**), în meniul **Sort & Filter** (**Sortare & Filtrare**), din grupul de butoane **Editing** (**Editare**) și pe fila **Data**, în grupul de butoane **Sort & Filter** (**Sortare & Filtrare**).

Sortarea datelor dintr-o anumită zonă a foi de calcul, poate fi realizată ascendent sau descendent, cu un click pe oricare celulă completată din coloana după care se sortează, apăsând unul dintre butoanele corespunzătoare (A-Z / Z-A - *vezi imaginea*).



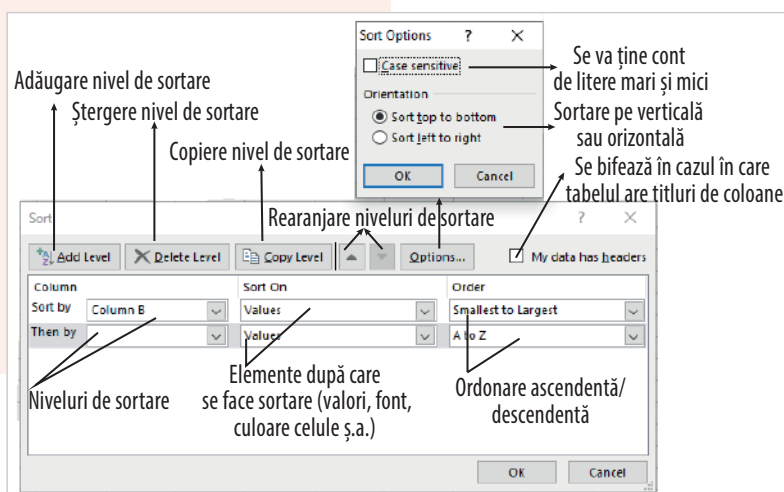
EX. **Exemplu:** Pentru a sorta alfabetic crescător datele dintr-un tabel cu informații despre elevi, după numele acestora, se dă click în coloana care conține numele elevilor, după care se apasă butonul de sortare ascendentă.

i **Obs.** Dacă în foaia de calcul, există mai multe seturi de date, separate prin coloane sau rânduri libere, sortarea se va face strict în setul de date în care se lucrează, restul foi rămânând nemodificată.

<def/> Într-o foaie de calcul, există situații în care datele de pe anumite coloane se repetă. În acest caz, pentru o analiză mai eficientă, este nevoie de sortare și după o altă coloană. Acest tip de sortare, care implică mai multe niveluri de sortare, se numește **sortare multinivel** și este realizabilă cu ajutorul instrumentului de **Sortare Particularizată** (**Custom Sort**).

i **EX.** **Exemplu:** Într-o foaie de calcul care conține informații despre elevii unei școli, pot exista doi, sau mai mulți elevi, cu același nume, poate chiar nume și prenume. În acest caz, sortarea elevilor doar după nume nu ar duce la rezultatele așteptate. Sortarea s-ar putea face după nume, ca prim nivel, prenume, ca nivel doi și data nașterii sau CNP ca nivel trei de sortare. Astfel, elevii ar fi aranjați după nume, iar în cazul în care există elevi cu același nume, aceștia și doar aceștia ar fi aranjați după prenume. Dacă există elevi care au și același nume și prenume, sortarea s-ar face după cel de al treilea nivel definit.

Modul de definire a nivelurilor de sortare și a criteriilor după care se realizează această operație poate fi analizat în imaginea alăturată.





Rețineți!

- 📖 Sortarea este acțiunea prin care datele sunt aranjate crescător sau descrescător în funcție de valorile de pe anumite coloane.
- 📖 Sortarea se poate face pe unul sau mai multe niveluri.
- 📖 În cazul sortării multinivel, datele sunt sortate în funcție de nivelurile definite. Sortarea se realizează pe primul nivel. În cazul datelor de pe acest nivel care coincid, se realizează sortarea în funcție de nivelul al doilea definit ș.a.m.d.



Fișa de lucru 6

Creați, în dosarul **Capitol1**, dosarul numit **Fisa6** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs (cele noi și cele cu care s-a lucrat în alte fișe).

1. (**) Deschideți aplicația **Microsoft Excel** și realizați următoarele cerințe:
 - a. Salvați registrul cu numele **rauri.xlsx**.
 - b. Realizați setările necesare astfel încât în registrul de calcul să existe 3 foi de calcul numite **râuri_top10_lungime**, **râuri_alfabetic**, **râuri_top10_bazin_hidrografic**.
 - c. În foaia de calcul **râuri_top10_lungime** realizați tabelul următor, începând cu celula A1:

Numele râului	Lungimea râului (km)	Lungimea pe teritoriul României	Bazin hidrografic (km ²)
Dunărea (fluviu)	2 860	1 075	817 000
Mureș	789	761	30 332
Prut	953	742	27 500
Olt	615	615	24 050
Siret	706	559	44 835
Ialomița	417	417	10 350
Someș	465	465	15 740
Argeș	350	350	12 550
Jiu	339	339	10 080
Buzău	302	302	5 264

Datele vor respecta următoarele formataări: antetul tabelului – font *Times New Roman*, 11pt, centrat pe orizontală și verticală, aldin; datele din restul tabelului – font *Bodoni MT*, 11pt, alinierea pe verticală la centru, pe orizontală vizibilă în tabel; înălțimea primului rând de 25pt, fundal gri deschis; liniile de grilă simple, negre.

d. Sortați datele crescător după lungimea totală a râului.

e. Copiați tabelul în foaia de calcul **râuri_alfabetic** și sortați datele în ordine alfabetică, după nume.

f. Copiați tabelul și în foaia de calcul **râuri_top10_bazin_hidrografic** și sortați datele descrescător după mărimea bazinului lor hidrografic.

2. (***) Deschideți registrul de calcul **angajati.xlsx**, din dosarul **Fisa5** și realizați următoarele cerințe:

a. Adăugați în tabel, pentru cinci angajați, următoarele date: (Alb, Diana, 34, ...; Alb, Maria, 32, ...; Popescu, Adrian, 23, ...; Coste, Paula, 50, ...; Albu, Alina, 48, ...). Restul datelor, care nu au fost specificate, completați-le cu valori la alegere.

b. Sortați datele alfabetic, crescător după nume, în cazul persoanelor cu același nume, descrescător după prenume, iar în cazul persoanelor care au și numele și prenumele identice, crescător după vârstă.

c. Inerați în registru o foaie de calcul cu numele **salariu** și copiați în această foaie datele din foaia de calcul **angajati**.

d. Sortați datele descrescător după salariu, iar în cazul în care salariile coincid, crescător după nume.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa6**.

		Da 	Nu 	Nu sunt sigur(ă) 
	La finalul lecției, știu...			
1.	să sortez crescător/descrescător datele dintr-un tabel folosind un criteriu de sortare.			
2.	să sortez crescător/descrescător datele dintr-un tabel folosind mai multe criterii de sortare.			

3. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 5. Serii de date. Formule de calcul. Funcții specifice aplicației de calcul tabelar

Serii de date



O **serie de date** se referă la un set de elemente (numerice, texte, date calendaristice ș.a.) existente în celule alăturate, elemente care se obțin după anumite reguli.

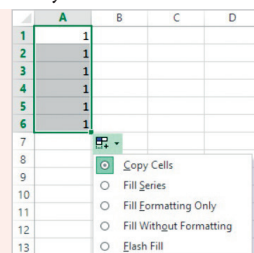
Aplicația **Microsoft Excel** permite generarea de serii de date, pe baza unor valori existente în celule, cu ajutorul instrumentului/ butonului **AutoFill** (**AutoUmplere**) sau a instrumentelor existente pe fila **Home** (**Pornire**), în meniul **Fill** (**Umplere**).



Marcajul de umplere (**AutoFill**) există, asociat fiecărei celule din foaia de calcul, fiind situat în colțul din dreapta jos al acesteia. Are forma unui pătrat mic. Cursorul are forma + deasupra sa, iar pentru a genera o serie de date, pe coloană sau rând, se dă click pe marcaj și cu butonul stâng al mouse-ului apăsat se deplasează cursorul în jos sau în lateral.

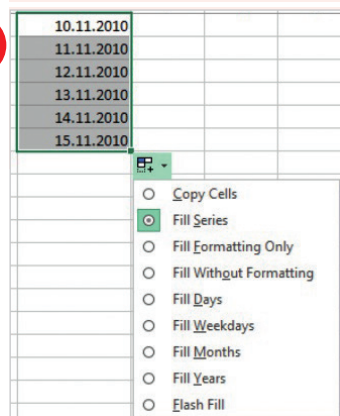


Exemplul 1. Dacă scriem în celula A1 valoarea 1 și acționăm marcajul de umplere, se obține o serie de valori identice cu aceasta. În lateralul seriei, apare un buton numit **AutoFill Options** (**Opțiuni AutoUmplere**) care permite copierea valorii, generarea unei serii de valori din 1 în 1, copierea formatului celulei și aplicarea pe celulele următoare, umplerea fără a păstra formatul celulei inițiale.



Exemplul 2. În imaginea alăturată, există serii numerice, în care valorile cresc din 1 în 1 (coloana C), 2 în 2 (coloanele D și E), -4 în -4 (coloana F) și o serie cu date de tip text, în care se repetă literele a și b. Pentru a obține automat aceste serii, se completează două celule, C1 și C2, D1 și D2 etc., după care se selectează și se utilizează marcajul de umplere.

C	D	E	F	G	H
1	1	2	4		a
2	3	4	0		b
3	5	6	-4		a
4	7	8	-8		b
5	9	10	-12		a
6	11	12	-16		b



Exemplul 3. Pentru a genera o serie de date calendaristice este suficient să se scrie prima valoare a acesteia, după care se generează seria. (Atenție la formatul de dată introdus! Acesta trebuie să coincidă cu formatul datei calendaristice a sistemului de operare.)

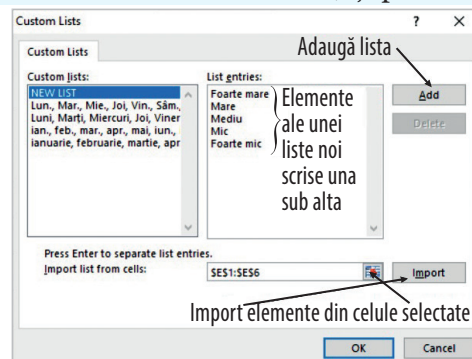
Meniul **AutoFill Options** permite modificarea elementelor care variază (ziua, luna, anul) și scrierea zilelor lucrătoare din săptămână (weekdays).



Aplicația **Microsoft Excel** are definite serii de date precum zilele săptămânii, lunile anului (dacă scriem într-o celulă textul *ianuarie* și generăm seria de date apar automat următoarele luni ale anului) și permite definirea de liste proprii, cu ajutorul

butonului **Edit Custom Lists** (**Editare Liste Particularizate**), din fereastra **Excel Options** (**Opțiuni Excel**). Aceasta se deschide din meniul **File** (**Fișier**).

În imaginea alăturată, este creată o listă cu 5 elemente, enumerate în caseta **List entries** (**Intrări listă**). După apăsarea butonului **Add** (**Adaugă**) lista va fi vizibilă în zona **Custom Lists** (**Liste Particularizate**) și va putea fi utilizată în foaia de calcul.





Fișa de lucru 7

Creați, în dosarul **Capitol1**, dosarul numit **Fisa7** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs (cele noi și cele cu care s-a lucrat în alte fișe).

1. (**) Deschideți registrul de calcul **rauri.xlsx** din folderul **Fisa6** și realizați următoarele cerințe:
 - a. În foaia de calcul **râuri_top10_lungime**, inserați în stânga tabelului o coloană goală. În celula A1, nou obținută, scrieți textul *Nr. Crt.*
 - b. Numerotați, automat în coloana A, cu valori de la 1 la 10, râurile din tabel.
2. (***) Deschideți registrul de calcul **exercitiu.xlsx** din folderul **Fisa5** și realizați următoarele cerințe:
 - a. Inserați două coloane noi în stânga coloanei A.
 - b. Începând cu celula B4 nou obținută, generați pe coloană, o serie de date care să aibă ca valori zilele lucrătoare dintr-o săptămână (Luni, Marți, ..., Vineri). Creați lista, în cazul în care aceasta nu există.
 - c. Creați o listă cu valorile: Euro, Dolar, Liră sterlină, Franc elvețian, Liră turcească și completați celulele C2, D2, ..., G2 cu valori ale acestei liste, începând cu Euro (în celula C2). Textele existente se vor înlocui.
 - d. Începând cu celula A4 creați o serie de date calendaristice, care va conține datele săptămânii în curs. (*Model în imagine*).
 - e. Completați cu date tot tabelul. Salvați registrul cu numele **kurs_valutar.xlsx**.

	A	B	C	D	E	F	G
1			CURS VALUTAR				
2			Euro	Dolar	Liră sterlină	Franc elvețian	Liră turcească
3			1 €	\$1	£1	1 CHF	1 TRY
4	12.09.2020	Luni	4,7794	4,2927	5,5717		
5	13.09.2020	Marti					
6	14.09.2020	Miercuri					
7	15.09.2020	Joi					
8	16.09.2020	Vineri					

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

Aplicațiile de calcul tabelar au fost realizate pentru a permite crearea de tabele și efectuarea de calcule cu datele din acestea. **Microsoft Excel** permite efectuarea de calcule simple și complexe cu valorile existente în celule și alte valori.



Formula de calcul poate fi scrisă în celula unde se dorește a fi afișat rezultatul sau în bara de formule, după ce s-a dat click pe celulă.



- Orice formulă de calcul începe cu semnul =, urmat de expresii de calcul.
- În funcție de datele utilizate (existente în celule) se pot obține rezultate de tip numeric, text, logic ș.a.



Operatorii utilizați în formulele de calcul pot fi:

- matematici (+ adunare, - scădere, * înmulțire, / - împărțire, % - procent, ^ ridicare la putere)
- relaționali (<, >, <=, >=, =, <> - diferit)
- de concatenare: & (lipește valori de tip text cu valori existente în celule)

Operanzii din formulele de calcul pot fi valori concrete (numere, texte – trecute între ghilimele) și adrese de celule, numite referințe la celule.

EX.

Exemple:

- = 4+A2 - adună valoarea 4 cu cea din celula A2
- = A3-A4 - scade valorile din cele două celule
- = A1*A2 - se înmulțesc valorile din A1 și A2
- = A1/A2 - se împart valorile din A1 și A2
- = 20%*A2 - se calculează 20% din valoarea existentă în A2
- = (A1+A2)/2 - calculează media aritmetică a valorilor din A1 și A2



- Ordinea de efectuare a operațiilor se poate schimba cu ajutorul parantezelor rotunde.
- Rezultatele calculelor în care sunt implicate referințe la celule se schimbă automat dacă valorile din celule sunt modificate.
- Există situații în care se obțin diverse erori de calcul precum **#DIV/0!** – în cazul în care se încearcă împărțire la 0, **#VALUE!** – în cazul în care valorile folosite în calcul nu sunt potrivite. (Ex: se adună valorile din celule care conțin text în loc de valori numerice), **#REF!** – când formulele de calcul implică adrese ale unor celule care au fost șterse.
- Formulele de calcul pot fi copiate în celulele alăturate cu ajutorul marcatului de umplere (**AutoFill**). În noile formule se vor modifica referințele la celule. Dacă se copiază formula în celula de pe rândul următor, se mărește cu 1 numărul rândului din adresa celulei, dacă se copiază formula în celule din coloana din dreapta, litera din adresa celulei implicată în calcul se schimbă cu următoarea. (Ex: = A1+B1 – copiată într-o celulă de pe rândul următor devine A2+B2, iar copiată într-o celulă de pe coloana din dreapta devine B1+C1). Pentru ca o adresă să nu se modifice, celula trebuie să fie **referită absolut**. Acest lucru se realizează dacă se folosește simbolul \$ în fața literei și în fața numărului din adresă (Ex: \$A\$4).



Fișa de lucru 8

Creați, în dosarul **Capitol1**, dosarul numit **Fisa8** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

(***) Deschideți aplicația **Microsoft Excel** și realizați următoarele cerințe:

- Salvați registrul cu numele **calcule.xlsx**.
- Completați celulele A1:C4 din foaia de calcul activă, cu valori numerice (naturale, întregi, raționale).
- În celula D1 scrieți o formulă de calcul pentru suma valorilor din celulele A1, B1, C1. Copiați formula, în jos, pe coloană, în celulele D2, D3, D4.
- În celulele de sub tabel realizați, pe baza valorilor din celule, următoarele calcule:

$$A1-A3 \quad B1*B2-C1*C2 \quad \frac{A3+A4-C1}{5} \quad \frac{5+A2*C4}{C4-A1}$$
- Într-o celulă goală din foaia de calcul calculați 20% din valoare existentă în celula A3.

Funcții specifice aplicației de calcul tabelar



Funcțiile sunt formule de calcul predefinite, utilizate pentru calcule complexe precum media aritmetică, valoarea minimă sau maximă, pentru a număra anumite valori ș.a.

Aplicația **Microsoft Excel** pune la dispoziția utilizatorilor o gamă vastă de funcții, împărțite în diverse categorii (matematice, statistice, financiare, logice ș.a.). Funcțiile pot fi scrise, la fel ca și calculele elementare, în celule sau în bara de formule sau pot fi inserate și particularizate cu ajutorul butonului **Insert Function (Inserare Funcție)** f_x situat în lateralul barei de formule.



Orice funcție începe cu semnul =, urmat de numele funcției și un set de paranteze rotunde, între care se specifică argumentele funcției.



▪ Argumentele funcției sunt datele cu care lucrează funcția și pot fi: valori concrete, adrese de celule și domenii.

▪ Un domeniu este o zonă de celule adiacente, identificată prin prima și ultima celulă, separate prin : (două puncte). (Ex: A1:A10 – identifică celulele A1, A2, ..., A10; A1:C1 – celulele A1, B1, C1; A1:B2 – celulele A1, A2, B1, B2)

▪ Există funcții fără argumente (Ex: NOW() – returnează data curentă), funcții cu un argument (Ex: SQRT (nr) – extrage rădăcina pătrată a numărului) sau funcții cu mai multe argumente (Ex: PRODUCT (A1;A2;A3) – înmulțește valorile din cele 3 celule). Argumentele sunt separate prin ; (punct și virgulă).

Funcția sumă – are sintaxa **SUM(argumente)** și se utilizează pentru a aduna argumentele.



Exemplu: **SUM(4;A1;A2)** - adună 4 cu valorile din celulele A1 și A2.

SUM(A1:A100) – adună valorile din toate cele 100 celule.

SUM(A1;A100) – aduna doar valorile din celulele A1 și A100.



Funcția SUM este mai utilă decât operatorul + dacă trebuie să adunăm mai multe valori.

Funcția minim – are sintaxa **MIN(argumente)** și returnează minimul dintre argumente.



Exemplu: **MIN(-4;-10; A2;A3)** - determină minimul dintre cele 4 argumente.

MIN(A1:A100) – returnează minimul valorilor numerice din cele 100 celule.

MIN(A1;A100) – returnează minimul valorilor numerice din celulele A1 și A100.

Funcția maxim – are sintaxa **MAX(argumente)** și returnează maximul dintre argumente.



Exemplu: **MAX(4;10;A2;A3)** - determină maximul dintre cele 4 argumente.

MAX(A1:A100) – returnează maximul valorilor numerice din cele 100 celule.

MAX(A1;A100) – returnează maximul valorilor numerice din celulele A1 și A100.

Funcția pentru media aritmetică – are sintaxa **AVERAGE(argumente)** și returnează media aritmetică a argumentelor.



Exemplu: **AVERAGE(4;10;A2;A3)**

AVERAGE(A1:A8) – returnează media aritmetică a valorilor numerice din cele 8 celule.

În cazul în care vreuna este goală, aceasta nu e luată în considerare în calcul.

AVERAGE (A1;A100) – returnează media aritmetică a valorilor din celulele A1 și A100.



Funcția pentru decizie – are sintaxa **IF(condiție; valoare_pt_adevarat; valoare_pt_fals)** și returnează **valoare_pt_adevarat** în cazul în care condiția dată ca prim argument este adevărată, **valoare_pt_fals** în caz contrar.



Exemplu: **IF(A1>=A2; “Valoarea din A1 este mai mare sau egala”; “Valoarea din A2 este mai mare”)** – în cazul acestei funcții se compară valorile din A1 și A2 și se returnează un mesaj corespunzător.

IF(A1*A2>0; “Valorile din celule au același semn”; “Valorile au semne contrare”)

IF(A1=”suma”; SUM(B1:B5); AVERAGE(B1:B5)) – dacă celula A1 conține textul “suma” se va calcula suma valorilor din B1:B5, în caz contrar se calculează media aritmetică.

IF(ABS(A1-A2)=1; “Celulele conțin numere consecutive”; “Celulele nu conțin numere consecutive”) – dacă valoarea absolută (modulul) diferenței celor 2 valori este 1, se afișează textul „Celulele conțin numere consecutive”.

ALTE FUNCȚII UTILE (CONȚINUT FACULTATIV)

Funcții de numărare

COUNTIF(domeniu; condiție) – returnează numărul de celule din domeniul specificat, care respectă condiția dată.



Exemplu: **COUNTIF(A1:A10;">0")** – numără valorile pozitive din zona indicată.

COUNTA(domeniu) – numără celulele din domeniu care nu sunt goale.

Funcții matematice

ABS(argument) – returnează valoarea absolută.



Exemplu: **ABS (A10)** – returnează modulul numărului din A10.

ROUND(valoare; nr_zecimale) – rotunjește valoarea la un număr cu **nr_zecimale**.



Exemplu: **ROUND (A5;3)** – rotunjește valoarea din A5 la un număr cu 3 zecimale.



Funcții logice – utile în implementarea condiției funcției decizionale IF

AND(condiție1; condiție2; condiție3;...) – intersectează valorile de adevăr ale condițiilor date ca argumente și returnează TRUE doar dacă toate condițiile sunt adevărate.



Exemplu: **AND(2<=3; 4=5)** – returnează FALSE deoarece a doua condiție este falsă.



- Pe măsură ce se scrie funcția, apar lângă cursor explicații cu privire la sintaxa acesteia.
- Adresele de celule apar colorate diferit, aceste culori fiind asociate și chenarului celulelor referite.

- Adresele de celule și domeniile pot fi scrise cu ajutorul tastaturii sau inserate cu click pe celulă, respectiv selectând domeniul.



Rețineți!

- 📖 Seriile de date sunt șiruri de valori generate în celule adiacente ale foii de calcul.
- 📖 Aplicația Excel permite efectuarea de calcule simple utilizând operatori precum +, *, -, /.
- 📖 Orice calcul sau formulă începe cu semnul =.
- 📖 În calcule și funcții pot fi utilizate valori concrete și referințe la celule (adrese de celule).



Fișa de lucru 9

Creați, în dosarul **Capitol1**, dosarul numit **Fisa9** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs (cele noi și cele cu care s-a lucrat în alte fișe).

1. () Deschideți fișierul *angajati.xlsx* din folderul *Fisa6* și realizați următoarele cerințe:**

a. În foaia de calcul **angajati**, scrieți una sub alta, în celulele A12, A13, A14 textele: *Salariul minim*, *Salariul maxim*, *Vârsta medie* și în celulele D12, D13 și D14 determinați valorile corespunzătoare, cu ajutorul funcțiilor studiate.

b. În celula H1 calculați suma salariilor tuturor angajaților.

c. Bazându-vă pe valoarea din celula D14, introduceți în celula H2 o funcție care va afișa textul *Angajați foarte tineri* în cazul în care vârsta medie a angajaților este sub 35 ani, iar în caz contrar textul *Angajați de vârstă medie*.

2. (**) Deschideți fișierul **rauri.xlsx** din folderul **Fisa7** și realizați următoarele cerințe:

- În prima foaie de calcul din registru, pe coloanele C, D și E, imediat sub tabel, calculați sumele valorilor de pe coloanele respective.
- În celula A20 calculați pe ce lungime, râurile enumerate în tabel nu curg pe teritoriul țării noastre.
- În celula A21 determinați lungimea medie a râurilor din tabel.
- În celula A22 inserați o funcție care determină dacă această lungime medie depășește 1500 și afișează textul *Lungime medie peste 1500 km*, iar în caz contrar *Lungime medie de cel mult 1500 km*.

3. (***) Deschideți fișierul **curs_valutar.xlsx** din folderul **Fisa7** și realizați următoarele cerințe:

- Determinați sub fiecare coloană, începând cu C, cursul valutar mediu al săptămânii, valoarea minimă și maximă atinsă.
- Inserați în celula A14 o funcție care afișează textul *Fluctuații mari*, în cazul în care diferența dintre valoarea maximă și medie a cursului monedei euro este mai mare decât diferența dintre valoarea medie și minimă a aceleiași monede și textul *Fluctuații mici* în caz contrar.
- (facultativ) Calculați, în celula A15, în câte zile cursul monedei dolar a depășit 4,78 lei.
- Inserați în registru o foaie de calcul nouă, numită **Conversie valutară** și scrieți în celula D1 valoarea 5,0512 (cursul valutar pentru moneda euro). În celula A1 Scrieți textul *Euro*, în B1 textul *Lei* și completați celulele A2:A10 cu valori numere naturale. Calculați, în celulele B2:B10 prețul în lei a valorilor din coloana A, utilizând ca și curs de schimb valoarea din celula D1. (Referiți absolut celula D1! --- \$D\$1)



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da*, *1 – nu*, *1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa9**.

	Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...			
1. cum se folosește instrumentul AutoFill.			
2. cum se folosesc operatorii aritmetici (+ , - , * , /).			
3. să utilizez funcția SUM.			
4. să utilizez funcțiile MIN și MAX.			
5. să utilizez funcția AVERAGE.			
6. să utilizez funcția IF.			

7. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 6. Reprezentarea grafică a datelor

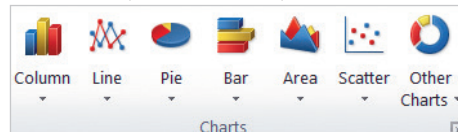
O foaie de calcul plină de numere, poate fi dificil de analizat, de înțeles. Pentru a interpreta diverse valori numerice, precum rezultatele elevilor la diverse teste sau examene, mediile elevilor dintr-o clasă ș.a., se recomandă reprezentarea grafică a datelor.



Graficele, numite și **diagrame (charts)** sunt obiecte utilizate în reprezentarea vizuală a datelor numerice din foaia de calcul și a relațiilor dintre ele.

Datele care stau la baza unei diagrame poartă numele de **sursă de date (Data Series)**.

Crearea de diagrame se face cu ajutorul instrumentelor existente în grupul de butoane **Charts (Diagrame)**, pe fila **Insert (Inserare)**.



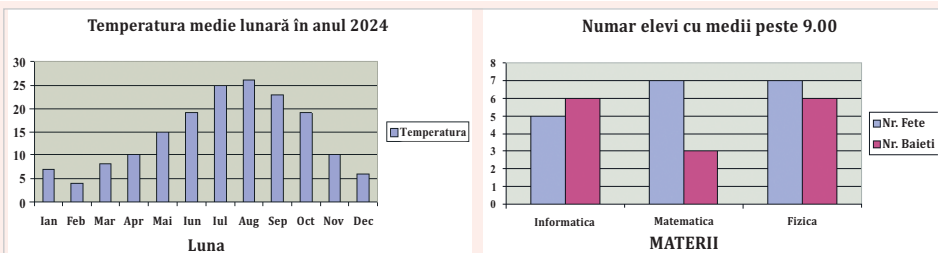
Graficele pot avea diverse forme, în funcție de ceea ce se dorește reprezentat. Cele mai des folosite sunt următoarele:

- **Grafice pe coloane – simple** – prezintă informații despre un singur element, **grupate** – compară informații despre mai multe elemente, **procentuale** – reprezintă procentual informații despre mai multe elemente ș.a.



Exemple: Tem-

peratura lunară medie în anul 2024 (grafic pe coloane simplu); Numărul de fete și băieți cu medii peste 9 la diverse materii (grafic pe coloane grupat), Numerele de absențe motivate și nemotivate ale elevului X, în anul școlar în curs (grafic pe coloane pe componente sau procentual).

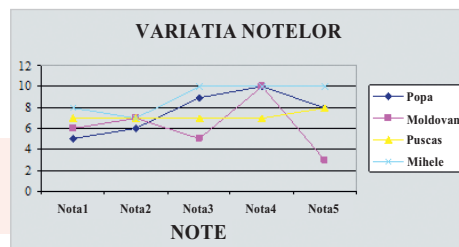


- **Grafice sub formă de bară** – la fel ca și graficele pe coloane se utilizează pentru a reprezenta comparativ informații despre unul sau mai multe elemente.

- **Grafice liniare** – se utilizează pentru a reprezenta variații ale unui element sau ale mai multor elemente.



Exemple: Variația notelor unor elevi la disciplina Matematică; Numărul de note de 4, 5, 6,..., 10 la un test.



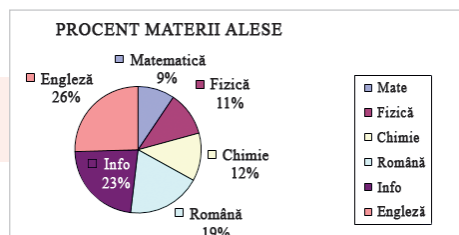
- **Grafice circulare** – sunt utilizate pentru a reprezenta procentual diverse elemente în raport cu un total.



Exemple: Rezultatul alegerii disciplinelor opționale într-o clasă; Cheltuielile pe luna ianuarie ale unei familii.



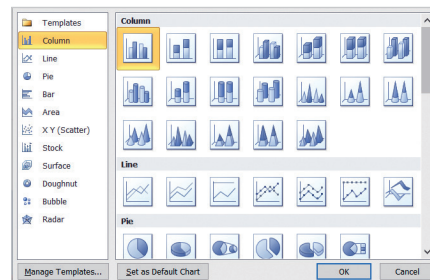
Fiecare categorie de grafice este divizată în mai multe subcategorii. Unele grafice permit o reprezentare a datelor 2D, altele 3D.





Crearea unei diagrame

Pentru a insera o diagramă, se selectează datele care stau la baza acesteia, după care se alege tipul de diagramă dorit (**Insert – Chart** – vezi imaginea alăturată). Aceasta va apărea automat în foaia de calcul.



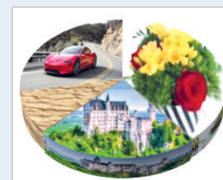
Elementele unei diagrame sunt:

- Peretele diagramei (*Chart area*), suprafața graficului (*Plot area*);
- Seriile de date (*Data Series*) – elementele prin care se reprezintă concret datele (bara, coloana, secțiune în cazul graficelor circulare ș.a.);
- Legenda – oferă informații despre elementele reprezentate grafic (poate lipsi);
- Titlul diagramei, titluri pentru axe.



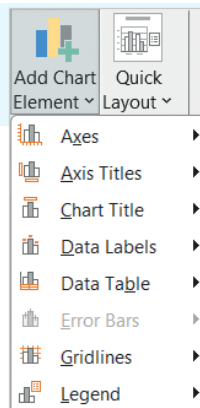
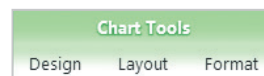
Știați că?

Fundalul elementelor unui grafic poate fi înlocuit de imagini reprezentative pentru informația la care face referire.



Formatarea diagramei

O diagramă poate fi modificată cu ajutorul instrumentelor de pe filele de lucru noi care apar când diagrama este selectată. În funcție de versiunea de program în care se lucrează, pot exista două sau trei asemenea file de lucru (**Design (Proiectare)**, **Layout (Aspect)** și/sau **Format (Formatare)**). Instrumentele cu ajutorul cărora se pot particulariza anumite elemente ale diagramei sunt accesibile și din meniurile lor contextuale.



În versiunile aplicației începând cu 2013, toate elementele necesare afișării elementelor pe grafic (legenda, titlul, date etc.) sunt grupate într-un meniu **Add Chart Element (Adaugă Element Diagramă)**.

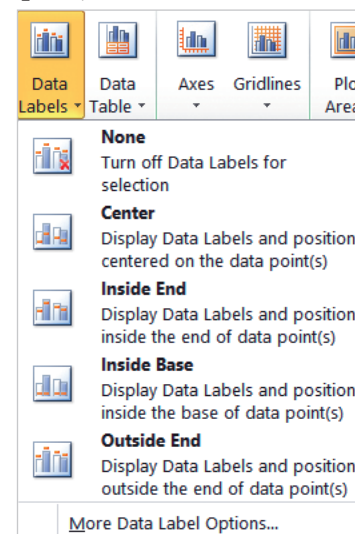
a. Formatarea legendei – presupune stabilirea poziției acesteia și a aspectului ei (culoare de fundal, linia de contur, aspect text). Textul din legendă se formatează cu ajutorul butoanelor de pe fila **Home (Pornire)**.

b. Formatarea titlurilor – Titlul diagramei poate fi poziționat central, deasupra graficului, deasupra diagramei sau poate lipsi. Diagrama poate conține titlurile pentru axa orizontală sau verticală. Toate aceste titluri se formatează cu ajutorul

butoanelor de pe fila **Home (Pornire)**, iar poziția lor poate fi stabilită și cu ajutorul mouse-ului, mișcând caseta asociată.

c. Afișarea etichetelor de date – Poziția etichetelor de date poate fi stabilită din meniul **Data Labels (Etichete de Date)** sau manual, cu ajutorul mouse-ului.

Modificarea tipului de diagramă se realizează cu ajutorul butonului **Change Chart Type (Modificare Tip Diagramă)** de pe fila **Design (Proiectare)**.





Fișa de lucru 10

Creați, în dosarul **Capitol1**, dosarul numit **Fisa10** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs (cele noi și cele cu care s-a lucrat în alte fișe).

(***) Deschideți registrul de calcul **curs_valutar.xlsx** din folderul **Fisa9** și realizați următoarele cerințe:

a. Inserați în foaia de calcul cinci diagrame, două de tip **3D Clustered Column** (pentru Euro și dolar) și 3 de tip **2D Line with markers** (pentru Lira sterlină, turcească și Franc elvețian) pentru a reprezenta fluctuațiile cursului valutar. (Selectați numele zilelor săptămânii și valorile numerice, pentru fiecare diagramă în parte.)

b. Colorați diferit suprafața fiecărei diagrame. Stabiliți pentru fiecare diagramă un titlu de forma **Curs valutar – monedă – în săptămâna ...**, plasat deasupra diagramei, formatat astfel încât să fie vizibil pe fundalul ales.

c. Ascundeți legenda unei diagrame și plasați legendele celorlalte 4, în părți diferite. Particularizați fundalul, linia de contur și aspectul textului unei legende dintre cele 4 astfel: fundal gri, linie de contur punctată, roșie, grosime 1pt, text – font *Cambria*, dimensiune 12, culoare roșie.

d. Colorați fundalul graficului și coloanele unei diagrame de tip **3D Clustered column**. Fiecare coloană va avea culoare diferită.

e. Afișați valorile numerice (*Data Labels*) în fiecare diagramă. Formatați valorile afișate.

f. Mutați trei dintre diagrame într-o foaie de calcul goală, numită **diagrame**, începând cu celulele A1, A15 și A30.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da*, *1 – nu*, *1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa10**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. să inserez un grafic în foaia de calcul, pornind de la date existente.			
2. să modific titlul graficului.			
3. să adaug legenda unui grafic.			
4. să modific culorile elementelor din grafic.			
5. să adaug datele pe elementele graficului.			
6. să mut un grafic în aceeași foaie dintr-o zonă în alta.			
7. să copiez/mut un grafic dintr-o foaie în alta.			

8. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:

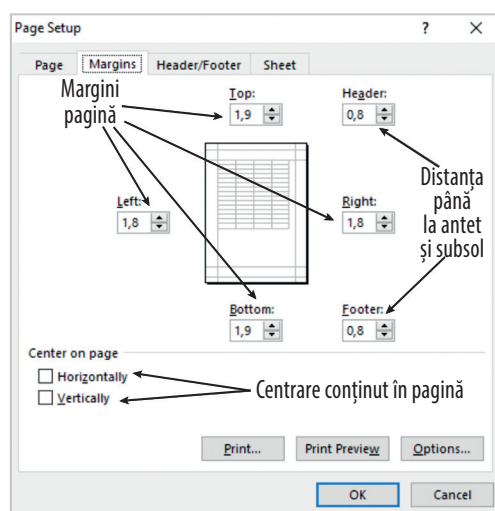
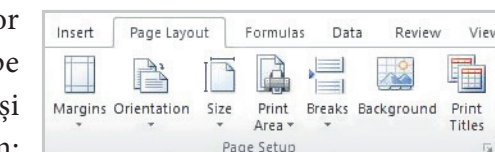
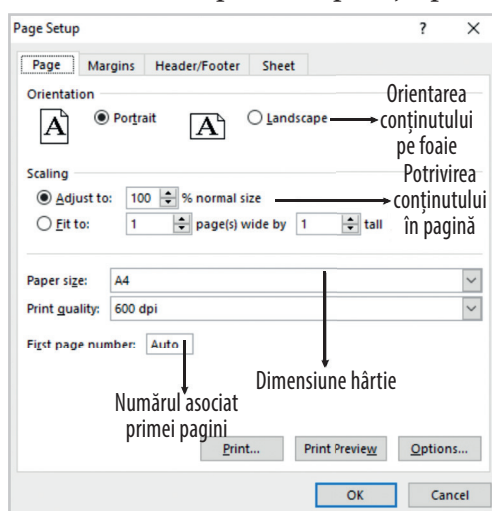


Lecția 7. Pregătirea registrului pentru imprimare

Spre deosebire de alte fișiere, fișierul **Excel** conține foarte multe date, răspândite pe suprafețe mari, iar conținutul unei foi de calcul poate fi tipărit pe mai multe pagini. Tipărirea unui asemenea fișier presupune o atenție sporită la formatarea paginilor din fiecare foaie de calcul și pregătirea datelor pentru tipărire.

i **Formatarea paginilor** se realizează cu ajutorul opțiunilor din grupul de butoane **Page Setup** (**Inițializare Pagină**), de pe fila **Page Layout** (**Aspect Pagină**) sau din fereastra cu același nume. Cele patru tab-uri ale ferestrei permit operații precum:

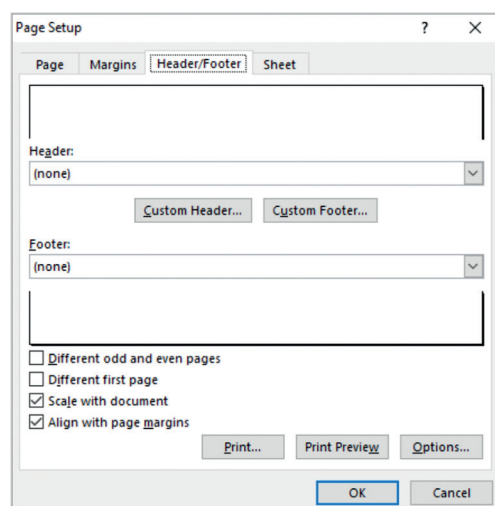
stabilirea orientării conținutului pe foaie, stabilirea marginilor, a distanței până la antet și subsol, centrarea pe orizontală și verticală a conținutului, scalarea conținutului astfel încât să încapă în pagină.

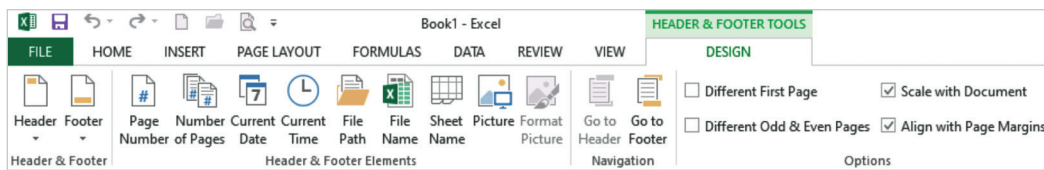


EX.

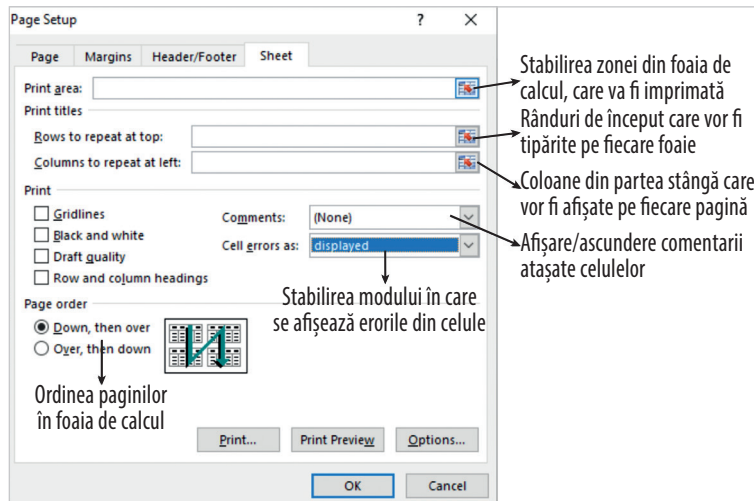
Exemplu: Dacă în secțiunea **Fit to:** se scrie **1 page(s) wide by 1 tall** conținutul va fi comprimat pentru a încapă pe o singură foaie. **2 pages wide by ____ tall** – va comprima conținutul astfel încât să încapă pe lățime pe 2 pagini, iar pe înălțime pe câte este nevoie.

Tab-ul **Header/Footer** (**Antet/Subsol**) permite particularizarea conținuturilor antetului și subsolului foilor. Antetul și subsolul pot fi diferite pe prima pagină, diferite pe pagini pare și impare sau identice pe toate paginile și au trei zone fiecare (stângă, centrală și dreaptă). Pentru a particulariza antetul și subsolul, se poate și trece în modul de vizualizare **Page Layout** (**Aspect Pagină Imprimată**). Astfel vor fi vizibile cele trei zone ale antetului, respectiv subsolului. La click pe oricare din aceste zone va apărea o filă de lucru (**Design**) cu instrumente care permit inserarea: numărului de pagină, a numărului total de pagini, a datei sau orei curente, a căii spre fișier, a numelui fișierului, a numelui foi de calcul sau a unei imagini. Formatarea imaginii se poate face cu ajutorul instrumentului **Format Picture** (**Formatare Imagine**).

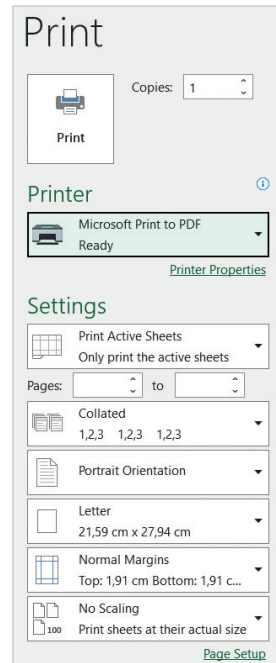




Textele din antet și subsol se formează cu ajutorul instrumentelor de pe fila **Home** (**Pornire**).



Instrumentele de pe tab-ul **Sheet** (**Foaie**) permit tipărirea liniilor de grilă (**Gridlines**), a titlurilor de rânduri și coloane (**Row and column headings**), alegerea zonei din foaie care va fi tipărită, a ordinii paginilor și tipărirea unor rânduri sau coloane pe fiecare pagină.



Tipărirea conținutului se realizează cu ajutorul opțiunii **Print** (**Imprimare**), existentă în meniul **File** (**Fișier**) sau pe bara de instrumente **Acces Rapid**. Combinația de taste pentru imprimare este **CTRL+P**.

Fereastra **Print** (**Imprimare**) permite stabilirea: imprimantei la care se va tipări (inclusiv tipărirea în fișier), a paginilor foii active sau a registrului de calcul, a dimensiunii paginilor (A4, Letter ș.a.), a orientării conținutului pe foaie etc.



Fișa de lucru 11

Creați, în dosarul **Capitol1**, dosarul numit **Fisa11** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

(**) Deschideți registrul de calcul **rauri.xlsx** din folderul **Fisa7** și realizați următoarele cerințe:

a. Formatați paginile din foaia de calcul **rauri_top10_lungime** astfel: dimensiune A4, orientare tip vedere (landscape), margini de 3 cm sus și jos, 2 cm în stânga și dreapta.

b. Centrați conținutul pe orizontală și realizați setările necesare astfel încât conținutul să fie tipărit pe o singură pagină.

c. Scrieți în antet, centrat, titlul *Cele mai lungi râuri din România*, formatat astfel: dimensiune 12pt, aldin, font *Times New Roman*, culoare albastră. Inerați în partea stângă din subsol numărul paginii, iar în partea dreaptă din subsol, numele foi de calcul pe primul rând, iar pe rândul următor data și ora, cu actualizare automată.

d. Realizați setările necesare astfel încât să fie afișate la tipărire liniile de grilă și titlurile de rânduri și coloane.

e. Formatați paginile din foaia de calcul **rauri_alfabetic** astfel: dimensiune A5, margini de 3 cm peste tot, 1 cm distanța până la antet și subsol, orientare tip portret (*portrait*).

(***)

f. Particularizați diferit antetul și subsolul paginilor pare și impare, astfel încât în antetul paginilor pare, în dreapta, să apară numele fișierului, centrat un titlu la alegere, iar în antetul paginilor impare, în stânga numele vostru și clasa. Numărul paginilor și numărul total de pagini vor apărea atât în subsolul paginilor pare cât și în subsolul paginilor impare, centrat.

g. Realizați setările necesare astfel încât să se repete rândul 1 la tipărire pe fiecare pagină și să fie tipărite 3 exemplare ale foi de calcul active, într-un fișier numit **rauri**, 3 exemplare ale foi de calcul active. Realizați o captură a ferestrei Print și salvați-o cu numele **tiparire.jpg**. Nu tipăriți fișierul.

h. Stabiliți zona A1:D6 ca zonă de tipărit.

i. Inerați în toate paginile, în subsol, o imagine cu fluviul Dunărea. Realizați setările necesare astfel încât imaginea să aibă dimensiunea de 1 cm × 1 cm.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa11**.

	Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...			
1. să pregătesc foaia de calcul pentru imprimare.			
2. să fac setările necesare pentru a imprima ceea ce doresc.			
3. să introduc date în antet și subsol.			
4. să particularizez diferit paginile pare și impare ale registrului.			

5. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:





Proiecte

Salvați toate fișierele pe care le creați, într-un dosar numit **Proiecte**, în dosarul **Capitol1**.

1. (**) Creați un registru de calcul cu numele **situatie_scolara.xlsx** și realizați următoarele cerințe:
 - a. Redenumiți foaia de calcul activă cu numele: **situatie_Info_TIC**.
 - b. În foaia de calcul **situatie_Info_TIC**, scrieți în celulele A1 și B1 textele *Nr. Crt.* și *Nume și prenume*.
 - c. În celulele C1:G1, scrieți textele: *Nota1*, *Nota2...*, *Nota5*, iar în celula H1, textul *Media*. (Consi-derăm că elevii primesc câte o notă în fiecare interval de cursuri.)
 - d. În celulele A2:A6 generați automat o serie numerică, având valori de la 1 la 5. În celulele B2:B6 scrieți numele și prenumele a cinci colegi, completați apoi notele acestora.
 - e. Calculați mediile colegilor în celulele H2:H6 și formatați valorile din celulă astfel încât să aibă 0 zecimale.
 - f. Conturați liniile de grilă (borders) ale tabelului, colorați fundalul celulelor din primul rând al tabelului, formatați textul la alegere și formatați coloanele astfel încât lățimile lor să fie potrivite automat la conținut.
 - g. (***) În celula I1 scrieți textul *Situație școlară* și determinați, folosind funcția IF, în cele cinci celule de sub aceasta, situația fiecărui elev în funcție de medie, afișând textului **promovat/corigent** după caz.
 - h. (****) În alte celule goale de sub tabel scrieți textele *Nr. Medii peste 8*, *Nr. note între 6 și 10* și determinați valorile corespunzătoare în celule alăturate.
 - i. Sortați datele din tabel descrescător după medii, iar în caz de medii identice, crescător după numele și prenumele elevilor.
 - j. Inserați o diagramă de tip **3D Clustered Column**, în care să fie vizibile *Numele și prenumele* și *Media*. Diagrama va avea titlul și valorile vizibile pentru fiecare coloană. Formatați diagrama la alegere. Mutați diagrama începând cu celula A15.
 - k. Stabiliți pentru foaia de calcul **situatie_Info_TIC** următoarele: dimensiune A4, orientare tip vedere, margini de 1 cm în stânga și dreapta, 2,5 cm sus și jos, 1 cm distanță până la antet și subsol. Centrați conținutul pe orizontală. Realizați setările necesare astfel încât datele să fie tipărite pe o singură pagină.
 - l. Scrieți în antet, centrat, textul *Situație școlară - Informatică și TIC, Clasa a VIII-a* și formatați-l în concordanță cu datele din tabel.

2. (***) Creați un registru de calcul nou, numit **calcule.xlsx** și realizați cerințele următoare:

a. Realizați setările necesare astfel încât registrul să conțină 4 foi de calcul, numite: **matematica**, **biologie**, **chimie**, **geografie**.

b. În foaia de calcul **matematica** realizați cele trei tabele din imaginile alăturate. Calculați, utilizând formule de calcul și funcții valorile lipsă din tabele. Pentru radical se folosește funcția SQRT. (Ex: SQRT(A1*A1+B1*B1); SQRT(3) – radical din 3)

Cateta1	Cateta2	Ipotenuza	Inaltime	Perimetru	Arie
3	4				
5		13			

Latura triunghi echilateral	Perimetru	Arie
4		
5		
2,5		

x	2x-3
2	
3	
-4	
-7	
10	

c. Sortați datele din tabelul în care s-a calculat valoarea expresiei $2x-3$, crescător după valorile lui x , apoi asociați-le o diagramă liniară.

(****) Stabiliți ca sursă de date pentru diagramă doar valorile din a doua coloană.

d. În foaia de calcul **biologie** scrieți în celulele A1 și B1 textele *Data* și *Înălțime*. Plantați un bob de fasole și începând cu prima zi din care răsare, măsurați-i înălțimea timp de 7 zile. Scrieți în tabel datele calendaristice și înălțimile măsurate.

e. În celulele D1 și D2 scrieți textele *Creștere medie* și *Înălțime* într-o lună, iar în celulele E1 și E2 calculați cât a crescut în medie, zilnic planta într-o săptămână și ce înălțime ar putea avea aceasta în 30 de zile.

f. În foaia de calcul **chimie** realizați tabelele din imaginile alăturate și calculați valorile lipsă din celule, pornind de la formulele specificate în antetul tabelor.

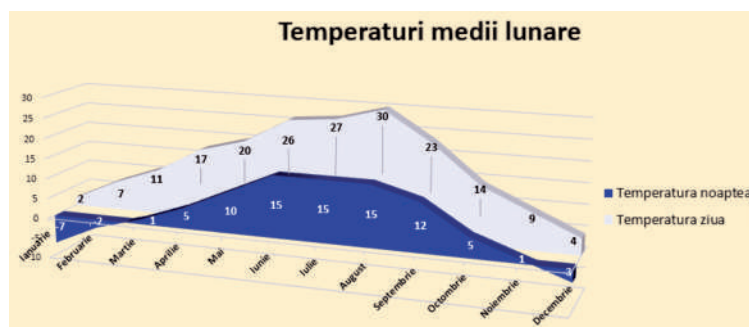
Masă subst. dizolvată - md	Masă soluție - ms	Concentrație procentuală $c = md * 100 / ms$
70	350	
	200	70%
50	200	

Masă subst. pură - mp	Masă subst. impurificată - mi	Puritate $p = mp * 100 / mi$
12	14	
	20	60%
18		75%

g. În foaia de calcul **geografie** realizați tabelul de mai jos și pornind de la valorile din acesta, creați o diagramă de tip *3D Area*, formatată precum cea din imaginea de mai jos.

	Ianuarie	Februarie	Martie	Aprilie	Mai	Iunie	Iulie	August	Septembrie	Octombrie	Noiembrie	Decembrie
Temperatura noaptea	-7	-2	1	5	10	15	15	15	12	5	1	-3
Temperatura ziua	2	7	11	17	20	26	27	30	23	14	9	4

h. Salvați registrul.





Verifică-ți cunoștințele!

Evaluare sumativă

(Punctaj maxim: 75 puncte)



Cerință: Creați, în dosarul **Capitol1**, un registru numit **gestiune.xlsx** și rezolvați următoarele cerințe:

a. Într-o foaie de calcul, numită **carti**, realizați tabelul alăturat, începând din celula A1. Respectați formatările. (Font *Arial Narrow*, 11pt)

Nr. Crt.	Denumire carte	Preț fără TVA	TVA	Preț cu TVA	Cantitate vândută	Data	Preț total
	Morometii	34			2	14.10.2021	
	Baltagul	29			3	14.10.2021	
	Ion	30			1	14.10.2021	
	Cei trei mușchetari	45			4	15.10.2021	
	Amintiri din copilărie	20			7	15.10.2021	
	Morometii	34			2	17.10.2021	
	Baltagul	29			4	17.10.2021	
	Amintiri din copilărie	20			5	19.10.2021	
	Ion	30			10	20.10.2021	
	Amintiri din copilărie	20			11	25.10.2021	

b. Completați cu valori de la 1 la 10, automat, celulele A2:A11.

c. Stabiliți pentru toate celulele care conțin prețuri, tipul de date Monedă (Currency) – lei. Realizați setările necesare astfel încât pe coloana *Cantitate vândută* să apară automat textul *buc* (bucăți) după fiecare număr.

d. Calculați valoarea TVA-ului, prețurile cu TVA și prețul total încasat pentru fiecare carte vândută.

e. Copiați tabelul într-o altă foaie de calcul, numită **calcule**, începând cu celula A1. Sortați datele crescător după denumirea cărții, iar în cazul cărților cu același nume, crescător după dată.

f. În celulele de sub tabel, calculați, folosind formule de calcul, cât s-a obținut în total din vânzarea fiecărui titlu. (De exemplu: Ion – 30 lei * 11 bucăți vândute în total = 330 lei)

g. Scrieți în celula A21 textul *Total încasări* și calculați în celula C21 cât s-a încasat în total din vânzarea cărților.

h. În celula A22 scrieți textul *Profit:*, iar în celula C22, utilizând funcția potrivită, afișați DA în cazul în care totalul încasărilor depășește 4000 lei, NU în caz contrar.

i. În foaia de calcul **carti**, inserați un rând liber, de fiecare dată când se schimbă data, pentru a putea calcula totalul încasărilor din fiecare zi. Colorați diferit fundalul celulelor acestor rânduri. Îmbinați celulele de pe coloanele de la A la G, scrieți textul **TOTAL ÎNCASĂRI** și calculați totalul încasat în ziua respectivă, în coloana H.

Nr. Crt.	Denumire carte	Preț fără TVA	TVA	Preț cu TVA	Cantitate vândută	Data	Preț total
	Morometii	34	6,46 lei	40,46 lei	2	14.10.2021	80,92 lei
	Baltagul	29	5,51 lei	34,51 lei	3	14.10.2021	103,53 lei
	Ion	30	5,70 lei	35,70 lei	1	14.10.2021	35,70 lei
TOTAL ÎNCASĂRI							220,15 lei

j. Determinați, în celulele A20, A21 și A22, suma totală maximă încasată în zilele din evidență, suma totală minimă încasată și o medie a sumelor încasate în cele 6 zile.

k. (****) Realizați, utilizând datele din celulele B1:B4 și H1:H4, o diagramă de tip *3D Clustered bar* și formatați-o la fel ca în imaginea alăturată. (font *Cambria*)



l. Mutați diagrama într-o foaie de calcul goală.

m. Scrieți în partea dreaptă a antetului foi de calcul **carti**, textul **Gestiune cărți**, inserați numărul de pagină, centrat în subsol și în partea stângă, în antet o imagine de dimensiuni 1,5 × 1,5 cm, reprezentând o carte.

Mult succes!

Barem de corectare posibil:

Cerințele **a-h**, 5 puncte fiecare (8 × 5 = 40p.); cerințele **i-m** – 7 puncte fiecare (5 × 7 = 35p.).



Investighează!

(Punctaj maxim: 15 puncte)

Una dintre aplicațiile de calcul tabelar cunoscute este cea oferită de firma Google. Pentru a descoperi asemănările și deosebirile dintre *Microsoft Excel* și *Google Sheets*, lucrați în echipe de câte 5 elevi și parcurgeți următorii pași:

1. Alegeți un responsabil care va accesa Google Drive (scriind în browser adresa drive.google.com) și va crea un fișier de tip Google Sheets, numit **proiecteCapitol1**, pentru a evalua proiectele realizate de cele 5 persoane din echipă. Responsabilul va partaja fișierul cu toți membrii echipei.

2. Realizați, individual, următoarele cerințe:

a. Inserați în fișier o foaie de calcul și dați-i numele vostru. b. Realizați în foaie, începând cu celula A1, tabelul din imaginea alăturată. c. Partajați, tot utilizând Google Drive, proiectele voastre cu colegii de echipă și evaluați proiectele colegilor. d. Stabiliți în echipă o ordine în care să completați punctajele pe care le-ați acordat colegilor din echipă și completați pentru fiecare coleg punctajele acordate. (În lipsa unor proiecte, acordați punctaje fictive!) e. Calculați în celula F3 media aritmetică a valorilor din celulele B3:E3. Copiați formula în celulele F4:F7. f. Introduceți în celulele A10 și A11 formule care să afișeze cel mai mic, respectiv cel mai mare punctaj acordat pentru toate proiectele. g. Sortați datele descrescător după coloana F (media).

A	B	C	D	E	F
	Evaluare proiecte				
	Punctaj 1	Punctaj 2	Punctaj 3	Punctaj 4	Media
Problema 1					
Problema 2 - matematică					
Problema 2 - biologie					
Problema 2 - geografie					
Problema 2 - chimie					

3. Descărcați fișierul în folderul **Capitol1**, păstrând numele **proiecteCapitol1.xlsx**.

4. Elaborați un document numit **investigheaza.docx** în care notați minimum: 5 asemănări, 3 deosebiri, 5 avantaje ale aplicației Google Sheets și 2 dezavantaje.

Modalitate de evaluare: Fișierul **proiecteCapitol1.xlsx** conține: • câte o foaie pentru fiecare membru din echipă - 3p • în fiecare foaie, tabelul specificat, completat cu valori/ punctaje - 2p • formulele corespunzătoare în coloana F, în fiecare foaie - 2p • formulele solicitate în celulele A10 și A11, în fiecare foaie - 2p, date sortate, în fiecare foaie - 2p (din cele 2 puncte se acordă doar 1 punct dacă cerințele au fost realizate doar în câteva foi – minim 1). Fișierul **investigheaza.docx** conține: • cinci asemănări - 1p • trei deosebiri - 1p • cinci avantaje - 1p • două dezavantaje - 1p.



Portofoliu

(Punctaj maxim: 10 puncte)

La finalul acestui capitol, portofoliul personal al elevului ar trebui să cuprindă în dosarul **Capitol1** următoarele materiale: Dosarul **Fisa1**, cu fișierele: **autoevaluare.docx**, **interfata.xls**, **interfata_grafica.xlsx**, **proprietati.jpg**, **raspunsuri.docx**, **rebus.docx** Dosarul **Fisa2**, cu fișierele: **date_elevi.xlsx**, **operatii_foi.xlsx**, **zoom.jpg** Dosarul **Fisa3**, cu fișierele: **autovaluare.docx**, **operatii_foi.xlsx** Dosarul **Fisa4**, cu fișierele: **angajati.xlsx**, **exercitiu.xlsx** Dosarul **Fisa5**, cu fișierele: **angajati.xlsx**, **autovaluare.docx**, **exercitiu.xlsx** Dosarul **Fisa6**, cu fișierele: **angajati.xlsx**, **autovaluare.docx**, **rauri.xlsx** Dosarul **Fisa7**, cu fișierele: **curs_valutar.xlsx**, **rauri.xlsx** Dosarul **Fisa8**, cu fișierul: **calcule.xlsx** Dosarul **Fisa9**, cu fișierele: **angajati.xlsx**, **autovaluare.docx**, **curs_valutar.xlsx**, **rauri.xlsx** Dosarul **Fisa10**, cu fișierele: **autovaluare.docx**, **curs_valutar.xlsx** Dosarul **Fisa11**, cu fișierele: **autovaluare.docx**, **rauri.xlsx**, **tiparire.jpg** Dosarul **Proiecte**, cu fișierele: **situatie_scolara.xlsx**, **calcule.xlsx** Fișierele **investigheaza.docx**, **gestiune.xlsx** și **proiecteCapitol1.xlsx**.

Modalitate de evaluare: Există dosarul **Capitol1** cu structura de directoare specificată - 2p. Există în dosare: ♦ peste 95% din fișiere - 8p ♦ între 75% și 94% - 6p ♦ între 50% și 74% - 4p ♦ între 25% și 49% - 2p ♦ sub 25% - 0p.

Recomandare! Nota finală a elevului se va stabili cumulând punctajele obținute la **Verifică-ți cunoștințele**, **Investighează** și **Portofoliu**.

Capitolul 2. Pagini web

Competențe: 1.2., 3.2.

În acest capitol vom învăța despre: ● Elementele de interfață ale unui editor de pagini web ● Instrumentele de bază ale editorului de pagini web ● Elementele de structură ale unei pagini web: antet, titlu, corp ● Operațiile de editare a elementelor de conținut (paragraf, imagini, tabele, liste, legături): inserare, ștergere, mutare, copiere ● Operațiile de formatare la nivel de text, paragraf, fundal ● Securitatea cibernetică

Lecția 8. Limbajul HTML. Editoare de pagini web

Limbajul HTML. Noțiuni generale



Pagina web este un fișier cu extensia *htm* sau *html*, scris cu ajutorul unui limbaj numit HTML (Hypertext Markup Language). Acest limbaj este interpretat cu ajutorul browserului, care arată utilizatorului conținutul paginii.



Limbajul **HTML**, folosit pentru crearea paginilor web (adică pentru scrierea codului sursă al paginii) este alcătuit din cuvinte speciale, numite **etichete** (tag-uri sau marcaje). Acestea sunt scrise între caracterele `< ... >` și marchează elementele paginii web. (Ex: `<HTML>`)



- Majoritatea etichetelor au etichete pereche, care conțin în plus caracterul `/` și marchează finalul acțiunilor etichetelor ale căror pereche sunt. (Ex: `</HTML>`)
- Unele etichete conțin cuvinte rezervate, numite **atribute**, cu ajutorul cărora se pot modifica proprietățile etichetei (Ex: `<FONT color="red">` - în eticheta FONT e utilizat atributul **color**).

Un site web este alcătuit din mai multe pagini web (una considerată principală) și alte fișiere. Pentru ca un site să poată fi accesat pe Internet, acesta trebuie încărcat pe un server web. Fiecare site are astfel o adresă URL unică, adresă asociată paginii principale. Din această adresă derivă adresele tuturor fișierelor care compun site-ul.



Știați că?

- 1989 – Tim Berners-Lee a inventat www
- 1991 – Tim Berners-Lee a inventat HTML
- 1995 – A fost lansat HTML 2.0
- 1997 – Consorțiul W3C recomandă HTML 3.2
- 1999 – Consorțiul W3C recomandă HTML 4.01
- 2000 – Consorțiul W3C recomandă XHTML 1.0
- 2014 – Consorțiul W3C recomandă HTML5
- 2017 – Consorțiul W3C recomandă HTML 5.2

Editoare de pagini web - instrumente de bază ale unui editor de pagini web



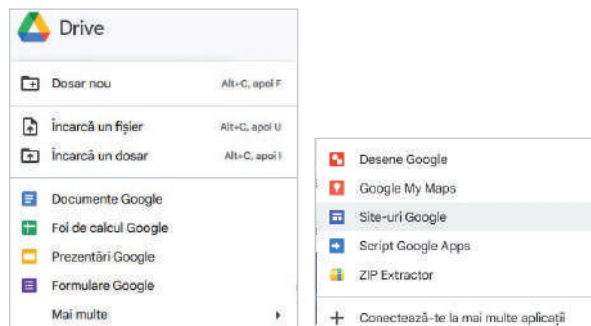
Un **editor de pagini web** este un program complex, specializat în elaborarea site-urilor. Există editoare care pot fi instalate în computer (Microsoft FrontPage, Notepad++, Dreamweaver, KompoZer) și editoare web online (Google Sites, WordPress, W3Schools, Joomla, Webnode).



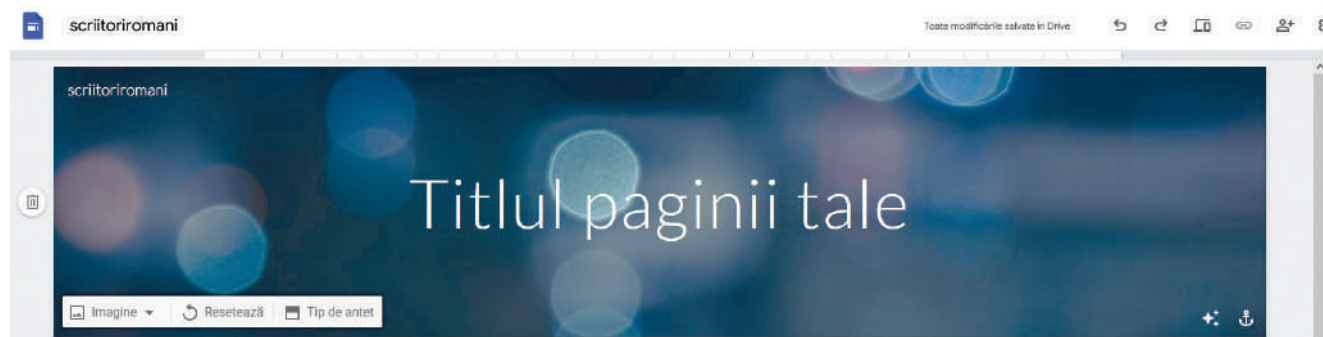
- O **pagină web** poate conține texte (titluri, paragrafe), imagini, liste, tabele și alte obiecte. Legătura între fișierele unui site se face prin elemente de tip hyperlink.
- Orice editor de pagini web trebuie să conțină instrumente de lucru cu ajutorul cărora se pot insera și modifica elementele de conținut ale paginii.

Editorul de pagini web Google Sites

Editorul online **Google Sites** poate fi accesat din contul personal **Google**, mai exact din aplicația **Google Drive**. Cu click pe **Nou** → **Mai multe** → **Site-uri Google** (**New** → **More** → **Google Sites**) se poate crea o pagină web cu un conținut minimalist, care poate fi particularizată cu ușurință.



Pagina web deschisă are în partea de sus, o **zonă de antet**, în care va fi inserat un banner la alegere, din galeria predefinită sau dintr-un fișier existent în calculator. Tot în zona de antet există o **casetă text** unde poate fi introdus un titlul în pagină, formatat după preferințe. Editarea și formatarea textului este ușor de realizat, deoarece există o bară de instrumente cu toate opțiunile necesare, bară care se activează când se execută click în caseta text.



Operații de editare a elementelor de conținut: inserare, ștergere, mutare, copiere

▪ **Inserarea** de elemente noi de conținut în pagină se poate face utilizând meniul **Inserază** (**Insert**), existent în partea laterală a ferestrei. În plus, cu ajutorul meniurilor **Pagini** (**Pages**) și **Teme** (**Themes**) pot fi inserate pagini noi și pot fi alese șabloane de formatare a elementelor paginii.

▪ **Ștergerea** unui element se face utilizând instrumentul **Elimină** (**Remove**), de pe bara de instrumente asociată obiectului (care apare la selecția acestuia) sau cu ajutorul tastei **Delete**.

▪ **Mutarea** unui obiect se face prin selectarea și glisarea acestuia în poziția dorită.

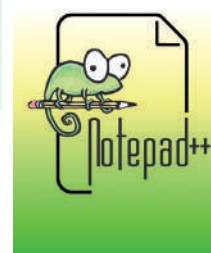
▪ **Copierea** unui obiect selectat se poate realiza folosind butonul **Duplică** (**Duplicate**) de pe bara de instrumente asociată obiectului sau folosind combinațiile de taste **CTRL+C**, respectiv **CTRL+V**.

Editorul de pagini web Notepad ++

Notepad++ este un editor gratuit, care permite editarea codului sursă specific unui număr mare de limbaje de programare, inclusiv editarea codului sursă al unei pagini web.

Pornirea aplicației Notepad++ se poate face astfel:

- Utilizând scurtătura din meniul **Start**;
- Căutând aplicația cu ajutorul instrumentului **Search** (**Căutare**) oferit de sistemul de operare.



Elementele de interfață ale aplicației Notepad++ sunt vizibile în imaginea de mai jos.

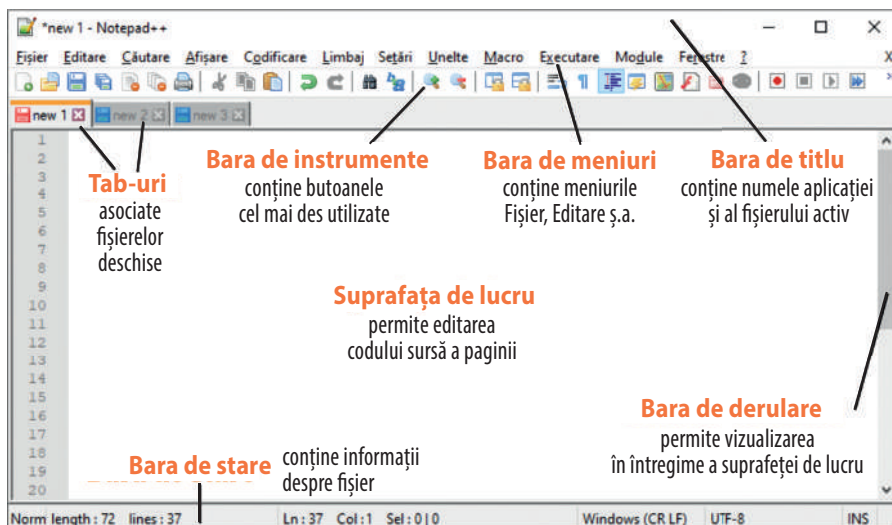
Principalele operații pentru gestionarea unei pagini web se pot realiza fie selectând din meniul **File** (*Fișier*) opțiunile corespunzătoare, fie acționând butoanele existente pe bara de butoane a aplicației. Acestea sunt:

a. Crearea unei pagini web – cu ajutorul opțiunii **New** (*Nou*) (**CTRL+N**).

b. Deschiderea unei pagini web – cu ajutorul opțiunii **Open** (*Deschidere*) (**CTRL+O**).

c. Salvarea unei pagini web – cu ajutorul opțiunii **Save** (*Salvare*) (**CTRL+S**), **Save all** (*Salvare toate*), **Save as** (*Salvare ca...*). Salvarea paginii web se realizează dându-se un nume paginii, urmat de caracterul ”.” și de extensia **htm** sau **html**.

d. Închiderea unei pagini web – cu ajutorul opțiunilor **Close** (*Închidere*) (**CTRL+W**) sau **Close All**.



Conținutul paginii web create cu editorul **Notepad++** se poate vedea cu ajutorul unui browser, dând dublu click pe fișierul htm/ html, în aplicația File Explorer.

Pagina web poate fi deschisă în editorul **Notepad++** și în browser în paralel. Orice modificare a codului sursă al paginii salvate poate fi văzută automat în browser prin apăsarea butonului **Refresh** (*Reîmprospătare*).



Fișa de lucru 12

Creați un dosar numit **Capitol2**. În acesta veți salva toate fișierele realizate pe măsură ce parcurgeți acest capitol. Pentru fiecare fișă de lucru parcursă creați, în dosarul **Capitol2**, un dosar cu numele **FisaNr**, unde Nr. va fi numărul de ordine al fișei de lucru. **Dosarul Capitol2 va face parte din portofoliul vostru personal.**



Cerință: Creați dosarul numit **Fisa12** și rezolvați exercițiile de mai jos salvând în acesta toate fișierele create.

1. (*) Identificați care sunt browserele instalate în calculatorul vostru și notați denumirea acestora într-un fișier text cu numele **1.txt**.
2. (**) Completați cuvintele lipsă din următoarele definiții, într-un fișier numit **2.txt**:
 - a. Pagina web este ..., scris cu ajutorul limbajului numit
 - b. Un ... este un program complex, specializat în elaborarea de pagini web.
 - c. Limbajul HTML este destinat realizării paginilor web și conține cuvinte speciale, numite
3. (**) Notați într-un fișier **3.txt** numele a patru editoare de pagini web.

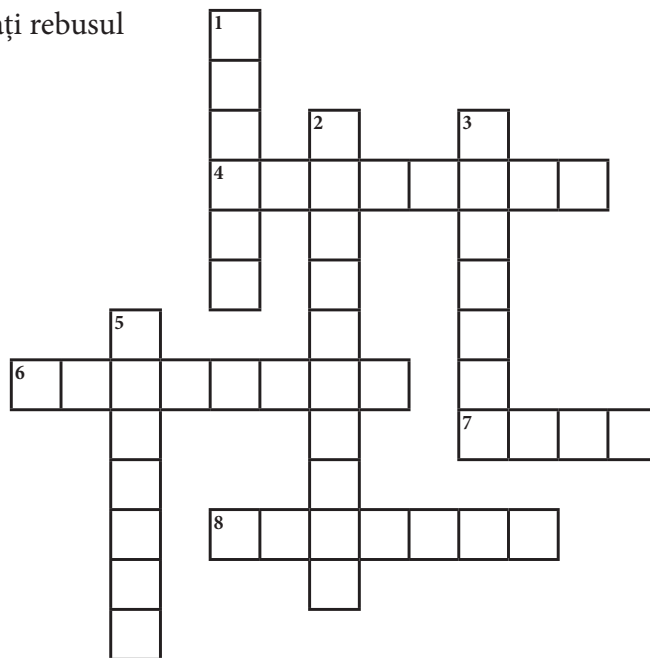
4. (***) Creați un document **rebus.docx** și rezolvați rebusul următor, identificând noțiunile următoare.

Vertical

1. Element asociat unui site web care identifică locația acestuia.
2. Texte care permit interconectarea fișierelor.
3. Buton accesibil în browser, care permite actualizarea modificărilor efectuate asupra paginii web.
5. Program cu care pot fi vizualizate paginile web. (denumirea în engleză).

Orizontal

4. Definesc cuvintele speciale în limbajul HTML.
6. Modifică proprietățile asociate unei etichete în limbajul HTML.
7. Limbaj folosit pentru realizarea paginilor web.
8. Numele unui editor gratuit de pagini web.



Autoevaluare

1. Verificați dacă ați răspuns corect la exercițiile din fișa de lucru, comparând răspunsurile voastre cu cele de mai jos:

2. a) un fișier cu extensia htm sau html; HTML b) editor de pagini web c) etichete; 4. 1) ADRESA 2) HIPERTEXTE 3) REFRESH 4) ETICHETE 5) BROWSER 6) ATRIBUȚIE 7) HTML 8) NOTEPAD

Punctaj: 1. – 1 punct, 2. – $0,5 \times 4 = 2$ puncte, 3. – $0,5 \times 4 = 2$ puncte, 4. – $0,5 \times 8 = 4$ puncte, 1 punct din oficiu

2. Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa12**.

	Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...			
1. ce este o pagină web.			
2. ce este limbajul HTML.			
3. ce este un editor de pagini web.			
4. care sunt elementele de conținut ale unei pagini web și cum se editează acestea.			
5. care sunt operațiile principale pentru gestionarea unei pagini web.			

6. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 9. Crearea unei pagini web

Elementele de structură ale unei pagini web: antet, titlu, corp



Începutul și sfârșitul codului sursă al unei pagini web este marcat prin etichetele `<html>` și `</html>`, între care se găsesc etichete pentru marcarea **antetului paginii (head)** și al **corpului acesteia (body)**. Antetul este definit cu ajutorul etichetelor `<head>` și `</head>` și oferă informații legate de titlul paginii, autorul ei, iar corpul paginii este definit cu ajutorul etichetelor `<body>` și `</body>`, între care se trec toate elementele de conținut.



Titlul paginii web este specificat între etichetele `<title>` și `</title>`, în antetul paginii și va apărea pe bara de titlu a browserului.



Exemplu: structura unei pagini web simple este prezentată în **figura 1**. În **figura 2** se poate observa cum este afișată pagina web în browser.

```

1 <!DOCTYPE html>
2 <html>
3 <head><title>Prima mea pagina web</title>
4 </head>
5 <body>
6 Acesta este prima mea pagina web!!!
7 </body>
8 </html>

```

Figura 1

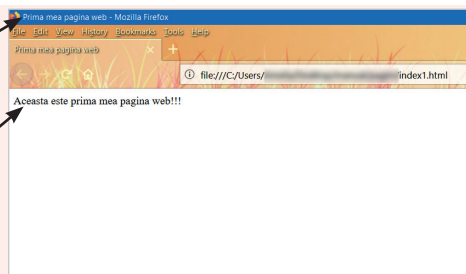


Figura 2



▪ Limbajul HTML nu face diferența între literele mari și literele mici, de aceea etichetele pot fi scrise cu majuscule sau cu litere mici.

▪ Declarația `<!DOCTYPE html>` apare înaintea etichetei `<html>` specificând versiunea de limbaj folosită și anume HTML 5.



▪ Pentru a insera comentarii care să fie ignorate de browser, acestea trebuie scrise între etichetele `<!--` și `-->`. Un comentariu oferă explicații legate de codul sursă al paginii web. (Ex: `<!-- Acesta este un comentariu -->`)



Fișa de lucru 13

Creați, în dosarul **Capitol2**, dosarul numit **Fisa13** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (*) Deschideți aplicația **Notepad++** și creați o pagină web cu numele **index1.html**. Scrieți, în fișierul creat, codul sursă de mai jos.

```

<!DOCTYPE html>
<html>
<head><title>Literatura română</title></head>
<body>
<h1>Caracteristicile perioadei interbelice în literatura română</h1>
<p>Perioada interbelică a literaturii române (1918-1939) a fost una extrem de bogată în creații literare. În această perioadă au reușit să coexiste și să „înflorească” două curente literare de orientări opuse: modernismul și tradiționalismul românesc.</p>
</body>
</html>

```

2. (*) Salvați pagina web și deschideți-o cu ajutorul unui browser.



3. (**) Identificați care este efectul etichetelor pereche `<h1>...</h1>`, respectiv `<p>...</p>` și adăugați un comentariu în pagina web, în care să explicați rolul acestora.

Fundalul unei pagini web

Fundalul unei pagini web poate fi particularizat cu o culoare sau o imagine.

a. Culori în fundal



Pentru a seta o culoare pentru fundalul paginii se folosește atributul **bgcolor**, al etichetei **body**, căruia i se atribuie fie numele culorii (ex: "red"), fie o valoare hexazecimală corespunzătoare culorii alese (ex: "#ff0000"), fie o valoare RGB (ex: "rgb(255,0,0)").



Exemplu:

```
<body bgcolor="#B6F6HA">...</body>
```



În versiunea HTML 5, atributul **bgcolor** poate fi înlocuit cu: `<body style="background-color: #B6F6HA">`. Acest mod de formatare este specific limbajului CSS (Cascade Style Sheets)¹.

b. Imagini în fundal



Pentru a plasa o imagine în fundalul paginii web se folosește atributul **background** al etichetei **body**, căruia i se atribuie numele și extensia fișierului de tip imagine, plasat în același dosar cu pagina.



Exemplu:

```
<body background="poza.jpg">...</body>
```



Atributul **background** nu mai este actual în versiunea HTML5, în locul lui folosindu-se sintaxa CSS: `<body style="background-image: url("poza.jpg")"> ... </body>`.



Fișa de lucru 14

Creați, în dosarul **Capitol2**, dosarul numit **Fisa14** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

- (*) Deschideți în **Notepad++**, pagina web **index1.html** creată în fișa anterioară.
- (**) Setati o culoare de fundal la alegere, prin una din metodele prezentate.
- (***) Salvați pagina cu numele **index2.html**. Căutați o imagine pe Internet care să aibă ca temă „Literatura română în perioada interbelică” și setati-o ca imagine de fundal pentru pagina nou creată.
- (***) Adăugați în pagina web următoarele curente literare: A. Tradiționalism, B. Modernism, C. Curente avangardiste. Căutați pe Internet informații despre aceste curente literare și adăugați câte un paragraf după fiecare curent literar enumerat. (Pentru elementele marcate cu A., B. și C., utilizați etichetele `<h1>...</h1>`, iar pentru paragrafe `<p>...</p>`)



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da*, *1 – nu*, *1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa14**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. care este structura unei pagini web.			
2. cum schimb culoarea de fundal a unei pagini web.			
3. cum setez o imagine ca fundal al unei pagini web.			

4. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



¹ CSS este un limbaj folosit pentru a descrie modul de prezentare a conținutului paginilor web.

Lecția 10. Titluri, paragrafe, texte în pagini web

Titluri



În limbajul HTML, un **titlu/subtitlu** este un text aldin, având o mărime dată de o valoare numerică. Pentru a insera titluri/subtitluri de diferite dimensiuni se folosesc marcate diferite: `<h1>`, `<h2>`, ..., `<h6>`.

Titlu cu dimensiunea 1

Titlu cu dimensiunea 2

Titlu cu dimensiunea 4



▪ Titlul marcat prin eticheta `<h1>` are dimensiunea cea mai mare, iar cel marcat prin eticheta `<h6>` are dimensiunea cea mai mică. Dimensiunea titlului de nivel 4 coincide cu cea a textului normal.

▪ Fiecare titlu are o dimensiune implicită, dar aceasta poate fi modificată cu ajutorul atributului **style**, folosind proprietatea CSS **font-size**.



Exemplu:

```
<h1 style="font-size:60px;"> Acesta este un titlu</h1>
```



Titlurile pot fi aliniate cu ajutorul atributului **align**. Valorile atributului **align** sunt: **left**, **center**, **justify** și **right**.



Exemplu:

```
<h1 align="center"> Acesta este un titlu aliniat centrat</h1>
```



Separarea conținuturilor unei pagini web se poate face cu o linie orizontală specificată prin eticheta `<hr>` care are atributele **width** (lățimea liniei exprimată în pixeli sau procent relativ la lățimea ecranului), **size** (grosimea liniei exprimată în pixeli), **align** și **color**.



Exemple: `<hr color="FF0000" width="500px" size="10px" align="center">`

`<hr color="AB00CC" width="100%" size="20px">`



Fișa de lucru 15

Creați, în dosarul **Capitol2**, dosarul numit **Fisa15** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (*) Deschideți cu **Notepad++** pagina **index2.html** din dosarul **Fisa14**. Salvați-o cu numele **index3.html**.

2. (**) Formatați titlul „Caracteristicile perioadei interbelice în literatura română”, astfel încât să aibă dimensiunea 1 și să fie aliniat în centrul paginii. Modificați textele corespunzătoare celor trei curente literare: „A. Tradiționalism, B. Modernism, C. Curente avangardiste” în titluri de dimensiunea 2, aliniate la stânga.

3. (***) Separați cele trei curente literare prin linii orizontale de culoare albastră, lățime identică cu cea a ferestrei, grosime 4px (vezi imaginea).

Caracteristicile perioadei interbelice în literatura română

Perioada interbelică a literaturii române (1918-1939) a fost una extrem de bogată în creații literare. În această perioadă au reușit să coexiste și să „înflorească” două curente literare de orientări opuse: modernismul și tradiționalismul românesc.

A. Tradiționalism

Reprezentanții curentului tradiționalist au mizat pe valorificarea tradițiilor românești și pe conservarea valorilor. Aceștia au decis să se îndepărteze de orice influențe moderne. Unul dintre cei mai importanți îndrumători ai scriitorilor tradiționaliști a fost Nichifor Crainic, director al revistei „Gândirea” din Cluj.

B. Modernism

La fel cum Nichifor Crainic a fost ideologul tradiționalismului, Eugen Lovinescu a jucat rolul de mentor al scriitorilor moderniști, operele acestora apărând în revista „Sburătorul” din București.

C. Curente avangardiste

Paragrafe



Un **paragraf** este un text cuprins între două alineate. Pentru a marca paragrafe în limbajul HTML se folosește perechea de etichete `<p>...</p>`.



Exemplu: `<p>Acesta este un paragraf</p>`



Pentru a face trecerea la rândul următor, în cadrul unui paragraf, se folosește eticheta `<br>`.



Exemplu: `<p>Acesta este un <br> paragraf</p>`

Perechea de etichete `<pre>...</pre>` definește un text preformatat, adică menține formătărilor, spațiile și întreruperile de rând din cadrul textului.



Paragrafele pot fi aliniate cu ajutorul atributului **align**, care poate lua valorile **left**, **center**, **right** și **justify**.



Exemplu: `<p align=justify> Acesta este un paragraf aliniat stânga-dreapta</p>`

- Eticheta `<div>` este folosită cu stilurile CSS, pentru a seta aspectul elementelor din interiorul său.



Exemplu: `<div style="background-color:pink"><p>Paragraf cu fundal colorat.</p></div>`

```

<!DOCTYPE html>
<html>
<head><title>Poeti</title></head>
<body>
<h1 align=left>Lucian Blaga</h1>
<h2> Eu nu strivesc corola de minuni a lumii
</h2>
<pre>
Eu nu strivesc corola de minuni a lumii
și nu ucid
cu mintea tainele, ce le-ntâlnesc
în calea mea
în flori, în ochi, pe buze ori morminte.
</pre>
</body>
</html>

```

Texte



Formatarea textului presupune stabilirea fontului, a dimensiunii, a stilului de scriere, a aspectului etc. Pentru a modifica proprietățile textului se folosește eticheta `<font>`, care are atributele **face**, **size** și **color**. Atributul **face** permite stabilirea fontului, atributul **size** ia valori de la 1 la 7 și stabilește dimensiunea textului și atributul **color** permite schimbarea culorii textului, prin atribuirea numelui culorii sau a unei valori hexazecimale corespunzătoare.



Exemplu: `<font face="arial" size=3 color="green">Acesta este un text formatat</font>`



Pentru formatarea textului cu CSS se folosește atributul **style**: `<style="proprietate:valoare">`.



Exemplu: `<p style="font-family:arial; font-size:11px; color:green; text-align:center">Acesta este un text formatat folosind CSS</p>`



Principalele etichete folosite pentru modificarea aspectului textului sunt:

- `<b>...</b>` - text îngroșat (aldin)
- `<i>...</i>` - text înclinat (cursiv)
- `<u>...</u>` - text subliniat
- `<sup>...</sup>` - text scris ca exponent
- `<sub>...</sub>` - text scris ca indice
- `<strike>...</strike>` - text tăiat cu o linie
- `<big>...</big>` - mărește textul cu o unitate
- `<small>...</small>` - micșorează textul cu o unitate
- `<mark>...</mark>` - text evidențiat

```

<!DOCTYPE html>
<html>
<head>
<title>Formatarea textului</title>
</head>
<body>
<p><b>Text scris ingrosat</b></p>
<p><i>Text scris cursiv</i></p>
<p><u>Text scris subliniat</u></p>
<p>Acesta este un <sub>indice</sub> si un <sup>
exponent</sup></p>
<p><strike> Text scris barat</strike></p>
<p><big>Text marit</big></p>
<p><small>Text micorat</small></p>
<p><mark>Text marcat</mark></p>
<p><b><i><u>Text scris ingrosat, cursiv si
subliniat</u></i></b></p>
</body>
</html>

```



Fișa de lucru 16

Creați, în dosarul **Capitol2**, dosarul numit **Fisa16** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (**) Utilizând **Notepad++**, creați o nouă pagină web cu numele **poet.html**. Stabiliți titlul paginii ca fiind „poeti” și adăugați textul din imaginea alăturată. Transformați textul „Lucian Blaga” în titlu de dimensiune 1, iar titlul poeziei în subtitlu de dimensiune 2.

2. (**) Căutați pe Internet întreaga poezie și inserați-o în pagină. Modificați fontul poeziei în **Verdana**, dimensiune de **14px**, culoarea textului - o nuanță de albastru, aspect cursiv (înclinat).

3. (*) Subliniați titlul poeziei și îngroșați cuvântul *flori* (toate aparițiile).

4. (***) Adăugați la finalul paginii, sub poezie, textul din imaginea de mai jos. Utilizați etichetele necesare astfel încât textul să fie formatat la fel ca în imagine (font – **Cambria**, dimensiune **12px**). Salvați pagina.

Lucian Blaga

Eu nu strivesc corola de minuni a lumii

Eu nu strivesc corola de minuni a lumii
și nu ucid
cu mintea tainele, ce le-ntâlnesc
în calea mea
în flori, în ochi, pe buze ori morminte.

Lucian Blaga a fost **filosof**, **poet**, **dramaturg** și traducator roman. **Personalitate impunatoare si polivalenta a culturii** romane interbelice, Lucian Blaga a marcat perioada respectiva prin **elemente de originalitate** compatibile cu inscrierea sa la **universitate**.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum **1 – da**, **1 – nu**, **1 – nu sunt sigur(ă)**, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa16**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. să formatez titluri/subtitluri (aliniere, dimensiune, separare).			
2. să formatez paragrafe (aliniere, preformatare, întrerupere, stiluri CSS).			
3. să formatez texte (tip font, dimensiune, culoare, stil de scriere).			

4. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 11. Liste în pagini web



O **listă** este o enumerare de elemente de același fel. În limbajul HTML există liste: ordonate, neordonate și de definiții.

▪ Liste ordonate



Lista ordonată este formată din elemente ordonate numeric/alfabetic. Aceasta se creează cu ajutorul etichetelor `<ol>...</ol>` (ordered list). Elementele listei se enumeră între etichetele `<li>...</li>` (list item).



Exemplu: *Figura 1* conține o listă ordonată, numerotată cu cifre arabe, iar *figura 2* codul sursă asociat.

O listă HTML ordonată

1. Lucian Blaga
2. Ion Barbu
3. Nichita Stănescu

Figura 1

```
<!DOCTYPE html>
<html>
<head><title>Exemplu</title></head>
<body>
<h2>O lista HTML ordonată</h2>
<ol>
  <li>Lucian Blaga</li>
  <li>Ion Barbu</li>
  <li>Nichita Stănescu</li>
</ol>
</body>
</html>
```

Figura 2



Numerotarea elementelor listei ordonate este automată și începe de la 1. Pentru a începe numerotarea de la o valoare diferită de aceasta se utilizează atributul **start=n** al etichetei `<ol>`, *n* fiind numărul dorit pentru primul element al listei.



Exemplu: `<ol start=4><li> Marin Sorescu</li></ol>`

Lista ordonată poate fi numerotată folosind litere sau cifre. Pentru a stabili modul de numerotare se utilizează atributul **type=s** al etichetei `<ol>`, *s* fiind unul din simbolurile: A, a, I, i, 1.



Exemplu: În cadrul sintaxei `<ol type=i start=3><li>Marin Sorescu</li><li>Ion Barbu</li></ol>` este definită o listă ordonată, numerotată cu cifre romane (i, ii etc.), având ca valoare de start „iii”.

▪ Liste neordonate



Lista neordonată este formată din elemente care nu sunt puse neapărat în ordine, elementele fiind marcate cu aceleași simboluri. Aceasta se creează folosind etichetele `<ul>...</ul>` (unordered list) și fiecare element al listei se enumeră între etichetele `<li>...</li>` (list item).



Exemplu: *Figura 1* conține o listă neordonată, marcată cu simbolul *disc*, iar *figura 2* codul sursă asociat.

O listă HTML neordonată

- Lucian Blaga
- Ion Barbu
- Nichita Stănescu

Figura 1

```
<!DOCTYPE html>
<html>
<head><title>Exemplu</title></head>
<body>
<h2>O lista HTML neordonată</h2>
<ul>
  <li>Lucian Blaga</li>
  <li>Ion Barbu</li>
  <li>Nichita Stănescu</li>
</ul>
</body>
</html>
```

Figura 2



Marcarea elementelor listei neordonate se face implicit cu simbolul *disc* ("●"). Aspectul marcatorului din fața fiecărei linii poate fi schimbat cu ajutorul atributului *type=v* al etichetei `<ul>`, unde *v* poate avea valorile *circle* ("○"), *disc* ("●"), *square* ("■") sau *none*.



EX.

Exemplu:

În cadrul sintaxelor de mai jos sunt definite două liste neordonate, marcate cu diverși marcatori.

```
<ul type=square><li>Tradiționalism</li><li>Modernism</li></ul>
```

```
<ul type=circle><li>Curente avangardiste</li></ul>
```



Știați că?

📖 Listele pot avea în interiorul lor alte liste. Acestea se numesc liste imbricate.

Exemplu: `<ol>`

```
<li>Poeți</li>
```

```
<ol type=a>
```

```
<li>Români</li>
```

```
<li>Străini</li>
```

```
</ol>
```

```
<li>Prozatori</li>
```

```
</ol>
```

▪ Liste de definiții



Lista de definiții este alcătuită din termeni și definițiile acestora și se creează cu etichetele `<dl>...</dl>` (definition list). Termenii listei se enumeră între etichetele `<dt>...</dt>`, iar definițiile termenilor între `<dd>...</dd>`.



EX.

Exemplu: *Figura 1* conține

o listă de definiții, iar *figura 2*

codul sursă asociat.

O listă HTML descriptivă

Pagina web

Colecție de hipertexte interconectate

Server web

Calculator care stochează pagini web

Figura 1

```
<!DOCTYPE html>
<html>
<head><title>Exemplu</title></head>
<body>
<h2>O lista HTML descriptivă</h2>
<dl>
  <dt>Pagina web</dt>
  <dd>Colecție de hipertexte interconectate</dd>
  <dt>Server web</dt>
  <dd>Calculator care stochează pagini web</dd>
</dl>
</body>
</html>
```

Figura 2



Știați că?

📖 Listele create în HTML pot fi afișate cu stiluri diferite, folosind CSS. De exemplu, afișând listele orizontal, se pot crea meniuri de navigare în pagina web.

Meniu de navigare

Proza Poezie Contact Despre



Fișa de lucru 17

Creați, în dosarul **Capitol2**, dosarul numit **Fisa17** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (*) Deschideți, cu ajutorul **Notepad++**, pagina web **index3.html** creată în directorul **Fisa15**. Salvați-o cu numele **index4.html**.

2. (**) După titlul paginii, creați un cuprins cu ajutorul listelor ordonate și neordonate. Formatați textul astfel: font *Arial*, culoare albastru. Prima listă din cuprins, va fi numerotată cu ajutorul literelor mari, iar cea de-a doua listă va fi neordonată, fără marcatori. Cuprinsul va arăta ca în imaginea alăturată.

Cuprins

Caracteristicile perioadei interbelice în literatura română

- A. Despre tradiționalism
- B. Despre modernism
- C. Despre curentele avangardiste

Scriitori reprezentanți

Poeți
Prozatori
Dramaturgi
Critici importanți/ Ideologi/ Mentori

3. (***) Adăugați sub paragraful care începe cu „Reprezentanții curentului tradiționalist...” și se termină cu „...al revistei «Gândirea» din Cluj”, din secțiunea **Tradiționalism**, conținutul din imaginea alăturată (subtitlul și lista neordonată).

Principii tradiționaliste

- crearea operelor literare bazate pe tradiții românești;
- un loc important în creațiile scriitorilor tradiționaliști le-au ocupat și ritualurile străvechi, miturile și valorile românești;
- idealizarea istoriei și a valorilor specifice lumii satului românesc;
- sublinierea rolului ortodoxiei, al credinței și al lui Dumnezeu;
- curentele literare promovate intens de către tradiționaliști au fost: gândirismul, sămănătorismul și poporanismul;
- apărarea valorilor naționale în fața „spiritului internaționalist”;
- promovarea autohtonismului;
- accentuarea dimensiunii spirituale a sufletului țărănesc;
- promovarea folclorului și a istoriei.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da*, *1 – nu*, *1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa17**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. să creez liste ordonate având diverse stiluri de numerotare.			
2. să creez liste neordonate având marcatori diverși.			
3. să creez liste de definiții.			
4. să creez o listă imbricată.			

5. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 12. Inserarea hiperlegăturilor

Introducerea unei legături în cadrul paginii web



Legătura (linkul) permite utilizatorului trecerea de la o pagină la alta. De obicei linkul este subliniat și acesta își schimbă forma la poziționarea cursorului deasupra lui. Linkul se numește **ancoră** și se inserează în pagină cu ajutorul etichetelor `<a>...</a>`.



▪ Eticheta `<a>` are următoarele atribute: **href** (indică adresa fișierului destinație), **target** (indică în ce fereastră se va deschide fișierul destinație), **title** (indică o descriere asociată legăturii și este afișată când cursorul se află deasupra legăturii).

Ex: `<a href="adresa_fisier" target="_blank" title="titlu">Textul care doriți să fie afișat în pagină</a>`



▪ Unele dintre valorile pe care le poate lua atributul **target** sunt: **_blank** (deschide documentul într-o fereastră nouă) și **_self** (deschide documentul în aceeași fereastră).



Exemplu: *Figura 1* conține un link spre motorul de căutare google.ro, iar *figura 2* codul sursă asociat. Atributul **href** specifică adresa de destinație a linkului (`https://www.google.ro/`). Această adresare este una absolută, prin specificarea întregii adrese.

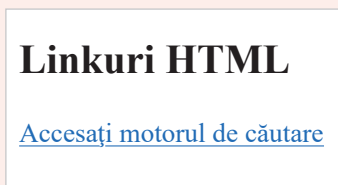


Figura 1

```
<!DOCTYPE html>
<html>
<body>
<h2>Linkuri HTML</h2>
<p><a href="https://www.google.ro">Accesați motorul
de căutare</a></p>
</body>
</html>
```

Figura 2

Tipuri de legături în paginile web



Știați că?

Există două tipuri de adrese URL: **absolute** (adresele web complete) sau **relative**.

Exemplu de adresă relativă: `<a href="nume_fisier.html">Text care asigură legătura </a>`



▪ Legăturile realizate prin **adresare relativă**, presupun referirea fișierelor pornind de la fișierul curent, respectiv dosarul în care se găsește acesta. Un fișier referit se poate afla în alt dosar decât fișierul curent sau în același dosar cu fișierul curent, caz în care e suficient să i se atribuie lui **href** numele și extensia fișierului referit.

▪ Pentru o mai bună organizare a unui site este indicată organizarea fișierelor în diverse directoare.

▪ **Legături către directoare care din punct de vedere ierarhic se află sub cel curent**



Pentru a face o legătură la un fișier aflat într-un director ierarhic situat sub cel curent, se trec toate directoarele, începând de la cel curent în jos, până la directorul părinte al fișierului, după care se trece numele fișierului și extensia.



Exemplu: Toate imaginile cu care se lucrează se află într-un director numit **imagini**, iar pagina în care se realizează legătura și directorul **imagini** sunt în același director părinte. Pentru a face legătura cu o imagine din acest director se scrie sintaxa:

`<a href="imagini/poza.jpg">Aceasta e o imagine</a>`.

▪ Legături către directoare care din punct de vedere ierarhic se află deasupra celui curent



Prin sintaxa `../` se transmite browserului să urce o treaptă în ierarhia de directoare. Dacă fișierul referit printr-un link se află într-un director deasupra celui curent (cel cu pagina) se folosește următoarea sintaxă:

```
<a href="../fișier.extensie">Text de legătură </a>
```



Exemplu: Figura 3 prezintă o ierarhie de directoare care conțin fișierele scrise în dreptul lor. În exemplele următoare se va face referire, pornind din paginile scrise în stânga simbolului →, la paginile scrise în dreapta acestui simbol.

index.html → flora.html

```
<a href="Pagini%20web/Componente/flora.html">Flora </a>
```

(%20 – trebuie să înlocuiască mereu spațiul din numele unui director sau fișier)

imagini.html → index.html

```
<a href="../../index.html">Inapoi la index</a>
```

imagini.html → romania.html

```
<a href="../../Pagini%20Web/romania.html">Romania</a>
```

romania.html → delta.jpg

```
<a href="../Fisiere/Imagini/delta.jpg">Imagine Delta Dunarii</a>
```

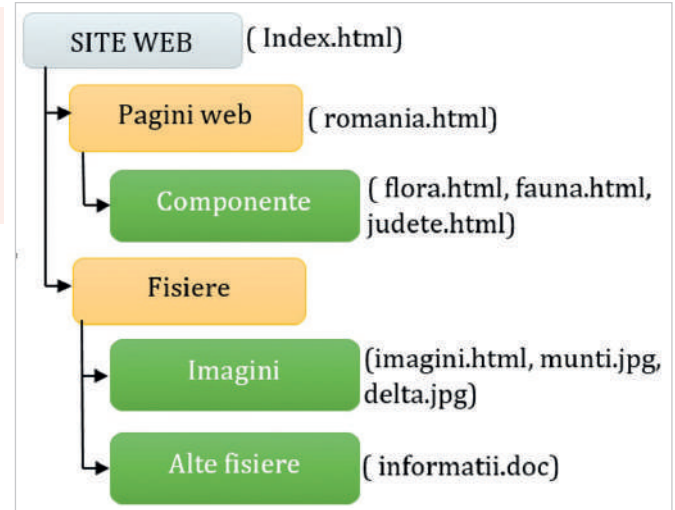


Fig. 3



Fișa de lucru 18

Creați în dosarul **Capitol2**, dosarul **Fisa18** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (*) Creați, cu **Notepad++**, pagina web **literatura.html**. Pagina va avea titlul „Perioade literare”. În pagină inserați textul „Epoci și perioade literare”, având dimensiunea 2. Realizați o listă numerotată cu cifre arabe care să conțină elementele din imaginea alăturată.

EPOCI ȘI PERIOADE LITERARE

1. Literatura română veche
2. Literatura română premodernă
3. Perioada pașoptistă
4. Epoca marilor clasici
5. Epoca de tranziție
6. Perioada interbelică
7. Perioada postbelică
8. Postmodernismul

2. (**) Copiați fișierul **index4.html** din dosarul **Fisa17**, în dosarul **Fisa18**. Realizați o legătură între elementul „Perioada interbelică” din lista creată la punctul anterior și pagina web **index4.html**. Codul sursă al paginii **literatura.html** va arăta ca în imaginea alăturată. Vizualizați pagina.

```
<!DOCTYPE html>
<html>
<head><title>Perioade literare</title></head>
<body>
<h2>EPOCI ȘI PERIOADE LITERARE</h2>
<ol>
  <li>Literatura română veche</li><br>
  <li>Literatura română premodernă</li><br>
  <li>Perioada pașoptistă</li><br>
  <li>Epoca marilor clasici</li><br>
  <li>Epoca de tranziție</li><br>
  <li><a href="index4.html">Perioada interbelică</a></li><br>
  <li>Perioada postbelică</li><br>
  <li>Postmodernismul</li><br>
</ol>
</body>
</html>
```

3. (**) Creați în directorul **Fisa18**, un subdirector **poeti**. Copiați în acesta pagina **poet.html** din **Fisa16**. Modificați pagina **index4.html** astfel încât după paragraful de după titlul „B. Modernism” să apară o listă neordonată precedată de textul „Poeți moderniști”, marcată cu simbolul „square”. Această listă va conține numele poezilor: George Bacovia, Lucian Blaga, Ion Barbu, Tudor Arghezi. Transformați numele poetului Lucian Blaga, într-un link care să ducă la pagina **poet.html** pe care ați copiat-o în subdirectorul **poeti**. Vizualizați pagina.

4. (**) În directorul **poeti** creați un subdirector **imagini**. Salvați în acest subdirector imagini ale poezilor moderni menționați la punctul anterior, imagini descărcate de pe Internet. Deschideți pagina **poet.html** în aplicația **Notepad++**. Sub poezia „Eu nu strivesc corola de minuni a lumii” introduceți textul **Poza autor** și transformați-l într-un link către poza autorului Lucian Blaga, din directorul **imagini**.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da*, *1 – nu*, *1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa18**.

		Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...				
1.	să creez legături către site-uri web.			
2.	să creez legături către fișiere situate în același dosar cu pagina.			
3.	să creez legături către fișiere situate în alte dosare decât cel în care e pagina.			
4.	să fac diferența între adresarea absolută și relativă.			

5. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 13. Inserarea și formatarea imaginilor



Știați că?

Fișierele HTML conțin adrese ale imaginilor care vor apărea în site. Imaginile trebuie încărcate pe serverul care va stoca paginile web. Este indicat lucrul cu imagini în formatul GIF sau JPEG, care să nu aibă dimensiuni mari.



Inserarea unei imagini într-o pagină web se realizează folosind eticheta `<img>`. Dacă nu este specificată nicio dimensiune pentru imagine, aceasta se va afișa la dimensiunea ei inițială.



Exemplu: ``



Eticheta `<img>` are următoarele atribute:

- **src** – precizează calea spre imaginea care va fi inserată. **Src** poate fi urmat de adresa URL a unei imagini sau de calea spre un fișier stocat în dispozitivul de memorie care conține și pagina web.



Exemplu: ``

- **align** – precizează modul de aliniere al imaginii față de textul sau obiectele care o înconjoară în cadrul paginii web. Atributul **align** are valorile: **left**, **right**, **top**, **bottom**, **middle**.



Exemplu: `<p>Acesta este un paragraf  Aici avem un text</p>`
Se pot folosi pentru alinierea imaginii și stilurile CSS.



Exemplu: `<p>Acesta este un paragraf  Acesta este un text</p>`



- **alt** – precizează un text alternativ care va fi afișat în cazul în care imaginea nu poate fi vizualizată.

Exemplu: ``



- **width** și **height** – ajustează lățimea, respectiv înălțimea imaginii, exprimată în pixeli:

Exemplu: ``

Pentru setarea dimensiunilor imaginii se pot folosi și stilurile CSS.



Exemplu: ``



- **border** – permite adăugarea unui chenar în jurul imaginii, având grosimea specificată în pixeli.

Exemplu: ``

Pentru modificarea chenarului, în HTML5 se pot folosi și stilurile CSS.



Exemplu: `<img style="border:5px solid black">`



- **hspace** și **vspace** – permit setarea distanțelor minime care separă imaginea de celelalte obiecte din pagină pe orizontală și verticală.



Exemplu: `<p> Acesta este un paragraf.</p>`



În versiunea HTML5 se folosesc stilurile CSS în locul atributelor **hspace** și **vspace**.

Exemplu: `<img style="margin: 0px 20px">`



Imaginile pot fi folosite și ca legături către alte pagini, fiind transformate în linkuri.



Exemplu: `<a href="https://www.google.ro" title="aceasta e o imagine legata"></a>`



Fișa de lucru 19

Creați, în dosarul **Capitol2**, dosarul **Fisa19** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (*) Copiați din dosarul **Fisa18**, fișierele **literatura.html**, **index4.html** și subdirectorul **poeti**, cu fișierele și subdirectoarele corespunzătoare lui. Deschideți fișierul **poet.html**, cu **Notepad++**. Căutați pe Internet imaginea unei flori și salvați-o în subdirectorul **imagini** din directorul **poeti**.

2. (**) Inserați imaginea salvată la punctul anterior în pagina **poet.html**, astfel încât să aibă următoarele proprietăți: dimensiunea de **100px** lățime, **100px** înălțime, alinierea imaginii față de text la stânga, bordura imaginii de **2px**. Setați un text alternativ care să fie afișat în cazul în care imaginea nu se încarcă în browser. Stabiliți distanța textului față de imagine la **10px** pe orizontală, respectiv **5px** pe verticală. Pagina va arăta ca în imaginea alăturată (Figura 1).

3. (**) Creați o pagină nouă cu titlul **Bacovia** și salvați-o cu numele **bacovia.html**. Pagina va conține pe primul rând titlul **George Bacovia** scris cu dimensiunea 1, iar pe rândul următor textul **Plumb**, scris cu dimensiunea 2. Copiați în pagină, de pe Internet, textul poeziei **Plumb**. Formatați textul astfel: font **Verdana**, dimensiune **14px**, culoare navy, aspect cursiv (italic). Pagina web va arăta ca în imaginea alăturată (Figura 2).

4. (***) În pagina **poet.html**, sub linkul cu numele autorului **Lucian Blaga**, inserați imaginile poezilor **George Bacovia**, **Ion Barbu** și **Tudor Arghezi**, imagini care se găsesc în subdirectorul **imagini**. Imaginile vor avea aceleași formătări ca imaginea de la punctul 2 al fișei de lucru curente, exceptând bordurile care vor avea valoarea 0. Inserați o legătură între imaginea poetului **George Bacovia** și pagina **bacovia.html**. Pagina web va arăta ca în imaginea alăturată (Figura 3).

Lucian Blaga

Eu nu strivesc corola de minuni a lumii



Eu nu strivesc corola de minuni a lumii
și nu ucid
cu mintea tainele, ce le-nvălesc
în calea mea
în flori, în ochi, pe buze ori morminte.
Lumina altora
sugrumă vraja nepătrunsului ascuns

în adâncimi de întineric,
dar eu,
eu cu lumina mea sporesc a lumii taină -
și-ntocmai cum cu razele ei albe luna
nu micșorează, ci tremurătoare
mărește și mai tare taina nopții,
așa imbogățesc și eu întunecata zare
cu largi flori de sfânt mister
și tot ce-i neînțeles
se schimbă-n neînțelesuri și mai mari
sub ochii mei-
căci eu iubesc
și flori și ochi și buze și morminte.

Figura 1

George Bacovia

Plumb

Dormea adânc sicriile de plumb,
și flori de plumb și funerar vestmânt -
Stam singur în cavou... și era vânt...
și scârțâiau coronele de plumb.

Dormea întors amorul meu de plumb
Pe flori de plumb, și-am început să-l strig -
Stam singur lângă mort... și era frig...
Și-i atârnav aripile de plumb..

Figura 2

Eu nu strivesc corola de minuni a lumii



Eu nu strivesc corola de minuni a lumii
și nu ucid
cu mintea tainele, ce le-nvălesc
în calea mea
în flori, în ochi, pe buze ori morminte.
Lumina altora
sugrumă vraja nepătrunsului ascuns

în adâncimi de întineric,
dar eu,
eu cu lumina mea sporesc a lumii taină -
și-ntocmai cum cu razele ei albe luna
nu micșorează, ci tremurătoare
mărește și mai tare taina nopții,
așa imbogățesc și eu întunecata zare
cu largi flori de sfânt mister
și tot ce-i neînțeles
se schimbă-n neînțelesuri și mai mari
sub ochii mei-
căci eu iubesc
și flori și ochi și buze și morminte.

Lucian Blaga



Figura 3



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum **1 - da**, **1 - nu**, **1 - nu sunt sigur(ă)**, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa19**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. să inserez imagini în pagini web.			
2. să formatez imaginile inserate în pagini.			
3. să folosesc sintaxe CSS pentru formatare.			

4. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 14. Inserarea și formatarea tabelelor



Paginile web pot conține tabele sau pot fi structurate sub formă de tabele. Etichetele folosite pentru a insera un tabel în pagină sunt `<table>...</table>`. O linie a tabelului este marcată de etichetele `<tr>...</tr>`. Antetul tabelului (primul rând) este definit prin etichetele `<th>...</th>` și au ca efect îngroșarea și centrarea textului din acel rând. O celulă a tabelului este definită prin `<td>...</td>`.



- Un tabel poate să conțină elemente diverse precum: texte, imagini, liste, alte tabele etc.
- Dacă un rând conține mai multe celule față de restul rândurilor, automat toate rândurile vor avea același număr, maxim, de celule.



Exemplu: *Figura 1* conține un tabel, cu patru rânduri și trei coloane (liniile de grilă nu se văd). Primul rând este antetul tabelului. *Figura 2* prezintă codul sursă al paginii web care conține tabelul din *figura 1*. Observați cum sunt organizate rândurile și coloanele.

Exemplul unui tabel

Nume	Prenume	Vârstă
Popescu	Ana	30
Marian	Andrada	13
Ionescu	Maria	45

Figura 1

```
<html>
<body>
<h2>Exemplul unui tabel</h2>
<table style="width:45%">
  <tr>
    <th>Nume</th>
    <th>Prenume</th>
    <th>Vârstă</th>
  </tr>
  <tr>
    <td>Popescu</td>
    <td>Ana</td>
    <td>30</td>
  </tr>
  <tr>
    <td>Marian</td>
    <td>Andrada</td>
    <td>13</td>
  </tr>
  <tr>
    <td>Ionescu</td>
    <td>Maria</td>
    <td>45</td>
  </tr>
</table>
</body>
</html>
```

Figura 2



Unui tabel i se poate atașa un titlu, specificat cu etichetele `<caption>...</caption>` care trebuie poziționate imediat după eticheta `<table>`. Atributul acestei etichete este *align* și definește locul în care va fi afișat titlul relativ la tabel. Poate lua valorile: *top*, *bottom*, *left*, *right*. *Figura 3* conține un tabel care are atașat un titlu, iar *figura 4* conține codul sursă al paginii web care conține tabelul din *figura 3*.

Exemplul unui titlu al tabelului

Tabel persoane		
Nume	Prenume	Vârstă
Popescu	Ana	30

Figura 3

```
<html>
<body>
<h2>Exemplul unui titlu al tabelului</h2>
<table style="width:45%">
  <caption>Tabel persoane</caption>
  <tr>
    <th>Nume</th>
    <th>Prenume</th>
    <th>Vârstă</th>
  </tr>
  <tr>
    <td>Popescu</td>
    <td>Ana</td>
    <td>30</td>
  </tr>
</table>
</body>
</html>
```

Figura 4



Eticheta `<table>` are următoarele atribute:

▪ **border** – definește grosimea liniilor care înconjoară tabelul, exprimată în pixeli.



Exemplu: `<table border=1>`

În versiunea HTML5 se recomandă folosirea stilurilor CSS:



Ex.: `<table style="border:1px solid black;">`

▪ **width/height** – definesc lățimea/înălțimea tabelului. Acestea pot fi exprimate în pixeli sau procente, relativ la dimensiunile ferestrei browserului.



Exemplu: `<table width=500 height=300>`
sau `<table width=50% height=50%>`

În versiunea HTML5 se recomandă folosirea stilurilor CSS:

EX. Ex.: `<table style="width:500px">`

▪ **align** – aliniaza tabelul în pagină și poate lua valorile: *left*, *right*, *center*. Dacă, pe lângă tabel, există și text, acesta se poziționează față de tabel la fel cum se poziționează față de imagini:

EX. Ex.: `<table border=1 align="center">`

▪ **cellspacing** – determină spațiul dintre celulele tabelului și este exprimat în pixeli.

EX. Ex.: `<table cellspacing=10>`

În HTML5 nu există acest atribut, în schimb se poate folosi atributul **border-spacing**:

EX. Ex.: `<table style="border-spacing: 10px">`

▪ **cellpadding** – determină spațiul dintre conținutul celulei și marginile acesteia, exprimat în pixeli.

EX. Ex.: `<table cellpadding=10>`

În HTML5 se recomandă folosirea CSS:

EX. Ex.: `<table style="padding: 10px">`

▪ **bgcolor** – determină culoarea de fundal a tabelului.

EX. Ex.: `<table bgcolor=#00FF00>`

În versiunea HTML5 se recomandă folosirea stilurilor CSS:

EX. Ex.: `<table style="background-color: blue">`

```
<html>
<body>
<h2 align="center">Situatie medii elevi</h2>
<table width=900 height=100 border=1 align="center"
bgcolor=EEFFDD cellspacing=4 cellpadding=14>
  <tr>
    <th>Nume si prenume elev</th>
    <th>Romana</th>
    <th>Matematica</th>
    <th>Fizica</th>
    <th>Chimie</th>
    <th>Info</th>
  </tr>
  <tr bgcolor="DD11AA">
    <td>Marian Andrada</td>
    <td align="center">9</td>
    <td align="center">8</td>
    <td align="center">7</td>
    <td align="center">9</td>
    <td align="center">8</td>
  </tr>
  <tr>
    <td>Marin Andreea</td>
    <td align="center">10</td>
    <td align="center">8</td>
    <td align="center">8</td>
    <td align="center">10</td>
    <td align="center">10</td>
  </tr>
</table>
</body>
</html>
```

Figura 5

Figura 6 conține imaginea unui tabel cu trei rânduri și cinci coloane, formatat conform specificațiilor din codul sursă prezentat în figura 5.

Situatie medii elevi					
Nume si prenume elev	Romana	Matematica	Fizica	Chimie	Info
Marian Andrada	9	8	7	9	8
Marin Andreea	10	8	8	10	10

Figura 6

? Știați că?

Etichetele `<tr>` și `<td>` au atributele: **align** și **valign** care aliniaza pe orizontală și verticală conținutul celulelor; **bgcolor** care stabilește culoarea de fundal a rândului/celulei.

Eticheta `<td>` are, în plus, atributele: **width** și **height** care stabilesc lățimea/ înălțimea celulei; **colspan="n"** - extinde orizontal celula peste *n* celule din dreapta ei; **rowspan="n"** - extinde vertical celula peste *n* celule de dedesubtul ei.



Fișa de lucru 20

Creați, în dosarul **Capitol2**, dosarul numit **Fisa20** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

1. (*) Deschideți **Notepad++** și creați o pagină web numită **orar.html**. Pagina va avea titlul „**Orarul clasei**” și va conține pe primul rând, scris cu dimensiunea 1, textul **ORARUL CLASEI** *numele_clasei*. Creați un tabel cu 6 coloane și 8 rânduri, care va arăta ca în imaginea alăturată. Inserați în antetul tabelului zilele săptămânii, iar în prima coloană intervalele orare. Stabiliți pentru tabel lățimea de 60%, bordură de 1px, culoarea bordurii – negru, pentru coloane lățimea de 100px, exceptând prima coloană care va fi lată de 50px.

ORARUL CLASEI A VIII-A B

	Luni	Marti	Miercuri	Joi	Vineri
8-9					
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					

2. (**) Personalizați celulele tabelului folosind culori de fundal la alegere și texte care să indice disciplina de studiu. Textul va fi formatat astfel: dimensiune 14 pixeli, aliniere la centru, font *Helvetica*.

ORARUL CLASEI A VIII-A B

	Luni	Marti	Miercuri
8-9	INFO	ROMANA	
9-10	INFO	FRANCEZA	
10-11	INFO	SPORT	
11-12	FIZICA	INFO	
12-13	CHIMIE	INFO	
13-14	ENGLEZA	INFO	
14-15	Orele de informatică se desfășoară pe grupe		

3. (***) Pe ultimul rând al tabelului, uniți celulele care corespund coloanelor *Luni* și *Marti*, folosind atributul **colspan**. Adăugați în celula obținută textul „*Orele de informatică se desfășoară pe grupe*” scris îngroșat, roșu și aliniat la centru.



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum *1 – da*, *1 – nu*, *1 – nu sunt sigur(ă)*, în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa20**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. să inserez tabele în pagini web.			
2. să formatez tabelele inserate într-o pagină web.			
3. să folosesc sintaxe CSS.			

4. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 15. Securitate cibernetică



Să ne reamintim!

Securitatea datelor este un element important atunci când este creat și gestionat un site, deoarece există anumite riscuri care pot fi prevenite. Un site web protejat este unul în cadrul căruia accesul se face restrictiv, prin nume de utilizator și parolă. Riscurile care pot să apară în crearea și gestionarea site-urilor sunt:

- ✓ În cazul site-urilor dinamice care permit conectarea utilizatorului la site, principalul risc este *furtul datelor personale*, date pe care utilizatorul este nevoit să le ofere creatorului site-ului, în momentul în care își creează un cont de acces;
- ✓ Tot în cazul site-urilor dinamice, care au în construcția lor implementate baze de date, există riscul accesului nepermis la aceste baze de date și a coruperii datelor stocate în ele;
- ✓ Blocarea accesului la propriul site;
- ✓ Umplerea site-ului cu comentarii de tip spam (reclame nedorite, repetitive și agresive).



Pentru a preveni problemele de securitate care pot să apară, se recomandă:

- În cazul site-urilor dinamice, în care utilizatorul accesează site-ul cu ajutorul unui cont de utilizator și a unei parole, parola pe care o definește trebuie să fie puternică, cu mai mult de 8 caractere, litere mari, litere mici, cifre și simboluri.
- Dacă site-ul este creat cu ajutorul unor platforme specializate, care permit conectarea proiectantului site-ului printr-un nume de utilizator și parolă, este recomandat ca acest nume de utilizator să nu fie „admin”, deoarece acesta este foarte ușor de descoperit.
- Platforma utilizată pentru crearea site-ului trebuie să fie mereu actualizată, precum și modulele¹ instalate pe site. Când se instalează module și alte aplicații pe site, trebuie instalate din surse sigure, iar cele nefolosite trebuie șterse.
- Folosirea unui certificat de securitate SSL (protocol folosit pentru securitatea pe Internet) este utilă celor care au afaceri/magazine online și colectează informații despre cumpărători precum: nume, număr de telefon, adresă, CNP etc. Cu ajutorul acestui certificat digital, o terță parte nu poate avea acces la datele utilizatorilor site-ului.
- Realizarea unor back-up-uri frecvente, în vederea salvării datelor.
- Alegerea unui server foarte bun pentru găzduirea site-ului (hosting), deoarece atacurile cele mai dese au loc din cauza breșelor de securitate la furnizorul de hosting.



Pentru a verifica dacă un site poate fi accesat în siguranță, se pot consulta simbolurile care apar în dreptul adresei paginii web și care se pot vedea în imaginea alăturată.

-  securizat,
-  informații sau nesecurizat,
-  nesecurizat sau periculos

¹ Modulele sunt aplicații software care aduc funcționalități suplimentare site-ului.

Alte elemente care țin de securitatea unei pagini web



Știați că?

Protocolul **https** (HyperText Transfer Protocol Secure) este un protocol de comunicație care este destinat transferului de informație criptată prin intermediul Internetului, dar este și o metodă de autentificare a serverului web care îl folosește, prin intermediul certificatelor digitale.

Cookie-urile sunt informații păstrate pe hard-diskul calculatorului, referitoare la opțiunile utilizatorului, astfel încât la o nouă vizitare a site-ului respectiv, particularitățile să fie încărcate automat. Cookie-urile sunt folosite pentru autentificare și pentru reținerea preferințelor utilizatorului.

O fereastră de tip **pop-up** este o fereastră mică care apare în partea de sus a site-ului vizitat și care conține de obicei reclame. Aceste ferestre pot fi blocate cu ajutorul navigatorului web.



Fișa de lucru 21

Creați, în dosarul **Capitol2**, dosarul numit **Fisa21** și rezolvați exercițiile de mai jos salvând în dosar toate fișierele realizate pe parcurs.

Creați un document numit **raspunsuri.docx** și rezolvați exercițiile următoare:

1. Asociați corect informațiile de pe cele două coloane. (Folosiți notații precum 1-a.)

1.	Site web securizat	a.
2.	Site web nesecurizat sau periculos	b.
3.	Informații	c.

2. Scrieți care sunt denumirile pentru următoarele acronime: **https**, **ssl**.

3. Enumerați cel puțin trei metode prin care pot fi prevenite problemele de securitate care pot să apară în **administrarea** paginilor web. Identificați alte trei pericole care pot să apară în momentul **accesării** unei pagini web. Argumentați răspunsurile!



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa21**.

		Da 	Nu 	Nu sunt sigur(ă)
	La finalul lecției, știu...			
1.	să identific riscurile care pot să apară în crearea și gestionarea paginilor web.			
2.	să respect recomandările necesare pentru a preveni eventualele probleme de securitate.			

3. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:





Investighează!

(Punctaj maxim: 10 puncte)

Deoarece există numeroase editoare online de pagini web, descoperiți asemenea aplicații, parcurgând pașii:

1. Alegeți un editor online de pagini web.
2. Descoperiți cum se folosește și ce facilități oferă.
3. Realizați un site pe una dintre următoarele teme: „Frumusețile României”, „Parcuri naturale din țara noastră”, „Personalități istorice din România”. Site-ul trebuie să conțină cel puțin 3 pagini web și imagini, texte, liste, legături, un tabel etc.
4. Salvați toate fișierele necesare site-ului într-un dosar numit **Investighează**, în dosarul **Capitol2**.

Sugestii: Lucrați în echipe de 3-4 elevi. La final prezentați și evaluați materialele create de fiecare echipă, după următoarele criterii:

Modalitate de evaluare:

- Dosarul **Investighează** conține cele trei pagini web care respectă tema aleasă – **3p**
- Paginile web conțin elementele indicate: imagini, texte, liste, legături, tabel – **5p**
- Designul paginilor web – **1p**
- Originalitate/creativitate – **1p**

Evaluare sumativă



Verifică-ți cunoștințele!

(Punctaj maxim: 75 puncte)



Cerință: Rezolvați exercițiile 1 și 2 într-un fișier **răspunsuri.docx**. Salvați toate fișierele pe care le creați, într-un dosar **Verificare**, în dosarul **Capitol2**.

1. Completați următoarele enunțuri: (2,5p × 4 = 10p)
 - a) Limbajul HTML este alcătuit din cuvinte speciale, numite.....
 - b) O pagină web este un fișier cu extensia
 - c) este un editor de pagini web.
 - d) O pagină web poate conține mai multe elemente interconectate prin elemente de tip
2. Stabiliți valoarea de adevăr a următoarelor afirmații: (2,5p × 6 = 15p)
 - a) Antetul unei pagini web este definit cu ajutorul etichetelor **<title>** și **</title>**!
 - b) Culoarele pot fi exprimate în limbajul HTML doar cu ajutorul valorilor hexazecimale!
 - c) În limbajul HTML există trei tipuri de liste!
 - d) Există două tipuri de adrese URL: absolute și relative!
 - e) Imaginile nu pot fi inserate într-o pagină web la dimensiunea lor inițială!
 - f) Paginile web nu pot fi structurate sub formă de tabele!
3. Creați o pagină web nouă, numită **testare.html** și rezolvați următoarele cerințe: (5p × 10 = 50p)
 - a) Stabiliți pentru pagină titlul *Proiect realizat de (nume elev)*.
 - b) Modificați în albastru deschis culoarea de fundal a paginii.
 - c) Introduceți textul **FLORA ȘI FAUNA ROMÂNIEI**, aliniat la centru, de culoare verde deschis, font Lucida Handwriting, dimensiune H1.

- d) Introduceți în pagină, pe rânduri diferite, următoarele texte: *Ce este flora?*, *Ce este fauna?*, *Flora României*, *Fauna României*.
- e) Inserați lângă fiecare text de la cerința d) o imagine sugestivă, preluată de pe Internet.
- f) Creați o pagină web nouă numită **flora.html**. Realizați legătura între textul *Ce este flora?* din pagina **testare.html** (cerința d) și pagina nou creată.
- g) În pagina **flora.html** introduceți textul *Ce este flora?* scris cursiv, îngroșat, culoare verde deschis, font *Lucida Handwriting*.
- h) Introduceți în pagina **flora.html** o definiție a florei.
- i) Realizați în aceeași pagină o listă neordonată cu tipurile de floră.
- j) Inserați un tabel cu 2 linii și 2 coloane și în fiecare celulă a acestuia câte o imagine cu diverse tipuri de flori. Imaginile vor fi preluate de pe Internet.



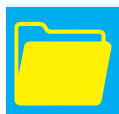
Proiect

(Punctaj maxim: 15 puncte)

Pentru că vă pasionează matematica, creați un site web, într-un dosar numit **proiect_num_elev**, în dosarul **Capitol2**, care să conțină informații despre corpurile geometrice în spațiu, rezolvând cerințele următoare:

- a) Creați o pagină web numită **matematica.html**, care va conține trei texte: piramide, cub/paralelipiped și prisme, texte formate la alegere.
- b) Creați trei pagini web, cu denumiri sugestive, în care vor fi prezentate corpurile geometrice, după următoarea schemă: definiție, clasificare, tabel cu imagini și formule de calcul.
- c) Inserați sub fiecare dintre cele trei texte din pagina **matematica.html** câte o imagine sugestivă și asociați imaginilor legături spre paginile web create la subpunctul b).

(Evaluare: 1p/fiecare pagină, 7p/elemente de conținut, design – 2p, creativitate – 2p)



Portofoliu

(Punctaj maxim: 10 puncte)

La finalul acestui capitol, portofoliul personal al elevului ar trebui să cuprindă în dosarul **Capitol2** următoarele materiale: Dosarul **Fisa12**, cu fișierele: **autoevaluare.docx**, **rebus.docx**, **1.txt**, **2.txt**, **3.txt** Dosarul **Fisa13**, cu fișierul: **index1.html** Dosarul **Fisa14**, cu fișierele: **autoevaluare.docx**, **index2.html** Dosarul **Fisa15**, cu fișierul: **index3.html** Dosarul **Fisa16**, cu fișierele: **autoevaluare.docx**, **poet.html** Dosarul **Fisa17**, cu fișierele: **autoevaluare.docx**, **index4.html** Dosarul **Fisa18**, cu fișierele: **autoevaluare.docx**, **index4.html**, **literatura.html**, **poet.html** Dosarul **Fisa19**, cu fișierele: **literatura.html**, **index4.html**, subdirectorul **poeti**, **autoevaluare.docx**, **bacovia.html**, **poet.html** Dosarul **Fisa20**, cu fișierele: **autoevaluare.docx**, **orar.html** Dosarul **Fisa21**, cu fișierele: **autoevaluare.docx**, **raspunsuri.docx** Dosarul **Investigheaza**, cu materialele aferente Dosarul **proiect_num_elevi**, cu materialele aferente Dosarul **Verificare**, fișierele **flora.html**, **raspunsuri.docx**, **testare.html**.

Modalitate de evaluare: Există dosarul **Capitol2** cu structura de directoare specificată – 2p. Există în dosare: ♦ peste 95% din fișiere – 8p ♦ între 75% și 94% – 6p ♦ între 50% și 74% – 4p ♦ între 25% și 49% – 2p ♦ sub 25% – 0p.

Recomandare! Nota finală a elevului se va stabili cumulând punctajele obținute la **Verifică-ți cunoștințele**, **Proiect**, **Portofoliu**.

Capitolul 3. Algoritmi

Competențe: 2.1., 2.2., 3.3.

În acest capitol vom învăța despre: • Șir de valori (noțiuni introductive) • Operații cu șiruri de valori: citire, afișare, parcurgere • Algoritmi de bază pentru șiruri de valori: numărare, suma / produs, minim / maxim, verificare a unei proprietăți

Lecția 16. Limbaje de programare – noțiuni recapitulative



Să ne reamintim!

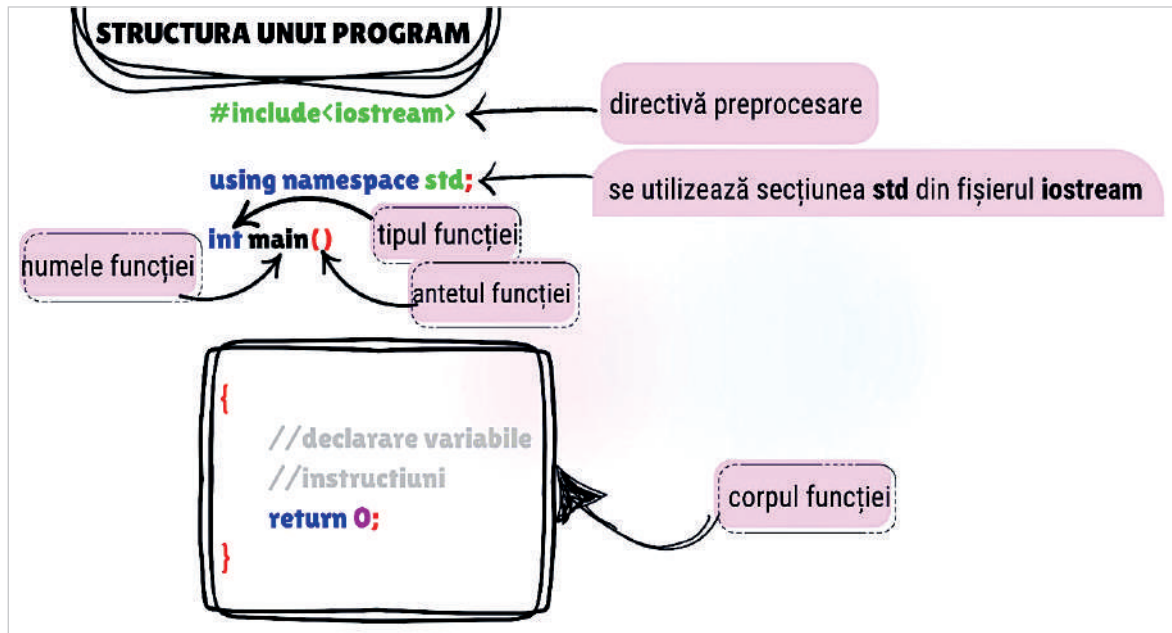


Limbajul de programare reprezintă mijlocul de comunicare dintre programator și calculator și este format dintr-un **vocabular** și un **set de reguli sintactice**, necesare pentru scrierea unui algoritm.



Programul este un **algoritm** scris utilizând un **limbaj de programare**.

Forma generală a unui program C++ se vede în imaginea de mai jos.



Funcția **main** este funcția principală, execuția programului începând întotdeauna cu această funcție.

Declararea variabilelor și a constantelor este obligatorie, pentru ca acestora să li se aloce spațiu în memorie și pentru a fi recunoscute în program.

```
//declararea variabilelor
tip nume_variabilă; //formă generală
tip nume_variabilă=valoare_inițială; //formă generală
/*Unei variabile i se poate da o valoare inițială deja la
declarare.*//

//declararea constantelor
#define nume valoare //formă generală
const tip nume=valoare; //formă generală
```

Tipurile de date elementare, utilizate în limbajul C++, sunt:



- **Numerice întregi** (int, unsigned int, short int, long/long int, unsigned long, long long, unsigned long long);
- **Numerice reale** (float, double, long double);
- **Șiruri de caractere** (char, unsigned char);
- **Logice** (bool).

Numele variabilelor pot conține litere (sunt *case sensitive*¹) mari, mici, cifre, linia de subliniere și nu pot conține următoarele caractere: **spații albe**, %, &, #, *, / ș.a. Primul caracter este, obligatoriu, literă sau _.

În proiectarea unui algoritm se utilizează următoarele structuri de bază:

a. Structura liniară – compusă din:

- *instrucțiunea de citire*;

Sintaxa: `cin>>variabila;`

sau `cin>>var1>>var2>>var3>>...>>varn;`

- *instrucțiunea de scriere*;

Sintaxa: `cout<<variabila;`

sau `cout<<var1<<var2<<var3<<...<<varn;` sau `cout<<"Text";` sau `cout<<variabila<<"Text";`

- *instrucțiunea de tip expresie* – utilizată pentru calcule diverse;

Operatorii utilizați în expresii sunt:

- *Aritmetici*: +, -, *, /, %
- *Relaționali*: <, <=, >, >=, == (egal), != (diferit)
- *Incrementare/Decrementare*: ++, -- (Ex: ++a, a++, --a, a--)
- *Logici*: &&, ||, !
- *Atribuire*: =, +=, -=, *=, /=, %=
- *De conversie explicită*: (tip) operand

b. Structura decizională/alternativă – având una din următoarele forme generale:

```
if (condiție)
{
    set_instrucțiuni_1;
}
else
{
    set_instrucțiuni_2;
}
```

```
if (condiție)
{
    set_instrucțiuni;
}
```

*forma simplificată
(unde lipsește ramura else)*

c. Structura repetitivă – utilizată pentru repetarea unei/unor instrucțiuni:

- *cu număr necunoscut de pași*:

- *repetitivă cu test inițial (condiționată anterior)* –

vezi **figura 1**.

- *repetitivă cu test final (condiționată posterior)* –

vezi **figura 2**.

- *cu număr cunoscut de pași*:

- *repetitivă cu număr*

cunoscut de pași (cu con-
tor) – vezi **figura 3**.

```
while (condiție)
{
    set_instrucțiuni;
}
```

Figura 1

```
do
{
    set_instrucțiuni;
} while (condiție);
```

Figura 2

```
for (contor = valoare_inițială; condiție_contor; pas_actualizare_contor)
{
    set_instrucțiuni;
}
```

Figura 3



Perechile de acolade pot lipsi doar în cazul în care în interiorul structurilor există o singură instrucțiune!

¹ Ex. S, s, SUMA, suma - sunt percepute de limbaj ca fiind variabile diferite.



Fișa de lucru 22

Creați un dosar numit **Capitol3**. În acesta veți salva toate fișierele realizate pe măsură ce parcurgeți acest capitol. Pentru fiecare fișă de lucru parcursă creați în dosarul **Capitol3** un dosar cu numele **FisaNr**, unde Nr. va fi numărul de ordine al fișei de lucru. **Dosarul Capitol3 va face parte din portofoliul vostru personal.**



Cerință: Creați dosarul numit **Fisa22** și rezolvați exercițiile de mai jos într-un fișier **raspunsuri.docx**. Salvați fișierul în dosarul creat.

Alegeți varianta corectă de răspuns:

1. Care dintre următoarele elemente nu fac parte, neapărat, dintr-un program scris în limbajul C++:

- | | |
|----------------------------------|----------------------------|
| a. Directive de preprocesare; | b. Antet; |
| c. Declarații de tipuri de date; | d. Definiții de constante. |

2. Care dintre următoarele declarații de date este corectă:

- | | | | |
|-------|-------|-------|-------|
| a. 1; | b. 2; | c. 3; | d. 4. |
|-------|-------|-------|-------|

```
1) int t, i, c;
2) int t, 2i, c;
3) int t i c;
4) int t, i, c
```

3. Stabiliți ce se va afișa în urma execuției secvenței următoare:

```
int a = 20;
cout<<"Valoarea variabilei a este " <<a;
```

- | |
|-----------------------------------|
| a. Valoarea variabilei a este 20 |
| b. Valoarea variabilei 20 este 20 |
| c. Valoarea variabilei a este 20 |
| d. Valoarea variabilei 20 este 20 |

4. Stabiliți valoarea de adevăr a expresiei $(a > b \ \&\& \ c == d)$, în urma execuției instrucțiunilor alăturate:

- | | |
|-------|-------|
| a. A; | b. F. |
|-------|-------|

```
int a=8, b=4, c=6, d;
a++;
d=--b;
d=c++;
```

5. Ce valori se afișează în urma executării următoarei secvențe:

- | | |
|----------------|---------------|
| a. 10, 7, 4, 1 | b. 10 7 4 1 |
| c. 9 6 3 0 | d. 9, 6, 3, 0 |
| e. 9630 | |

```
int x = 10;
do
{
    cout<<x-1<<" ";
    x = x - 3;
}while (x > 0);
```

6. Parcurgeți următoarea secvență și stabiliți ce se va afișa:

```
int i;
for (i = 10; i >= 2; i -= 3)
    cout<<i<<" ";
```

- | | |
|---------------------|---------------|
| a. 10 7 4 1 | b. 10 8 6 4 2 |
| c. 10 9 8 7 6 5 4 3 | d. 10 7 4 |

7. Parcurgeți algoritmul alăturat, scris în limbajul C++, și stabiliți ce se va afișa dacă se introduce valoarea 98765:

- | | | | |
|-------|-------|--------|-------|
| a. 48 | b. 14 | c. 360 | d. 21 |
|-------|-------|--------|-------|

```
#include<iostream>
using namespace std;
int main()
{
    unsigned int a, p;
    cin>>a;
    p = 1;
    while (a!= 0)
    {
        if (a%10%2 == 0)
            p *= a%10;
        a /= 10;
    }
    cout<<p;
    return 0;
}
```





Aplicații suplimentare

Scrieți în limbajul C++, un algoritm de rezolvare pentru următoarele probleme:

1. (*) Hobby-ul lui Robert este confecționarea zmeiilor. Recent el a confecționat un zmeu cu diferite stele pe cele două părți ale sale. Dacă numărul de stele de pe fiecare parte a zmeului este **palindrom** (un număr este **palindrom** dacă el este egal cu oglinditul său; de ex. 11, 12321), atunci Robert va putea ieși la joacă cu zmeul, iar în caz contrar el va dona zmeul surioarei mai mici, Ana. Ajutați-l pe Robert să afle ce va face cu zmeul, știind că pe fiecare parte a zmeului numărul de stele este cuprins între 2 și 99. (Exemplul 1: $p_1=33$, $p_2=7$ – se va afișa mesajul "Robert poate ieși la joacă", Exemplul 2: $p_1=25$, $p_2=5$ – se va afișa mesajul "Robert îi oferă zmeul surioarei mai mici, Ana")



2. (***) O firmă specializată în configurarea sistemelor de alarmă realizează soft-uri care determi-

nă coduri de alarmă perfect securizate. Pentru a avea un cod **perfect securizat**, cifrele din acesta trebuie să îndeplinească următoarele condiții: **cifra minimă** să fie **pară**, **cifra maximă** să fie **divizibilă cu 3** și **numărul de cifre** din cod să fie **impar**. Se introduc de la tastatură mai multe coduri. Verificați dacă există unul perfect securizat pentru alarmă, afișând codul securizat, iar dacă nu există un astfel de cod afișați mesajul **Nu există!**. Introducerea de coduri se oprește la introducerea valorii 0. (Exemplul 1: 234567, **8822397**, 2223, 0 – se va afișa

8822397, deoarece cifra **minimă** 2 este **pară**, cifra **maximă** 9 este **divizibilă cu 3**, iar **numărul de cifre** este 7; Exemplul 2: 11111, 2222, 33333, 0 – se va afișa mesajul **Nu există!**)



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa22**.

	Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...			
1. ce este un limbaj de programare și un program.			
2. care este forma generală a unui program C++.			
3. cum se declară variabilele și constantele.			
4. care sunt structurile utilizate în proiectarea unui algoritm.			
5. care sunt tipurile de operatori utilizați în expresii.			
6. Am întâmpinat dificultăți în: ...			

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 17. Șir de valori – noțiuni introductive. Operații cu șiruri de valori



Un **șir de valori**, numit și **tablou unidimensional**, este o colecție de elemente de același tip, așezate continuu în memorie. Elementele șirului pot fi accesate individual și sunt identificate prin poziția lor în șir, numită și **indice**.



Știați că?

Există tablouri unidimensionale, bidimensionale și multidimensionale.



Exemplu:



Declararea unui șir de valori se face ținând cont de următoarele elemente: **tipul** valorilor din șir, **numele** șirului și **dimensiunea** maximă a acestuia.

Forma generală de declarare a unui șir este: **tip nume_șir[dimensiune];**



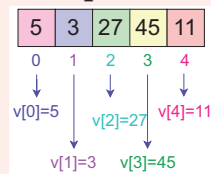
Exemplu:

```
int v[25]; // un șir care poate reține maximum 25 de numere întregi
double x[10]; // un șir care poate reține maximum 10 de numere reale
char s[30]; // un șir care poate reține maximum 30 de caractere
```

Accesarea unui element dintr-un șir de valori se face pornind de la **numele** șirului, urmat între paranteze drepte de **indicele (poziția)** pe care îl (o) ocupă elementul în șir.



Exemplu:



Observați că în exemplul alăturat șirul v are 5 elemente, iar indicii iau valori de la 0 la 4. Un element al șirului se poate utiliza în calcule ca orice altă variabilă: $v[0]$ – va referi primul element din șir, $v[1]$ al doilea element ș.a.m.d.

Un număr de **n elemente** se pot reține într-un șir pe poziții de la **0** la **$n-1$** sau de la **1** la **n** , cu condiția ca, în cazul din urmă, șirul să aibă dimensiunea minimum **$n+1$** . (Convenim că reținem elemente începând cu poziția 0).

Parcurgerea unui șir de valori



Pentru a putea citi, afișa, respectiv a prelucra elementele unui șir e necesară parcurgerea șirului de valori. Parcurgerea se face, în general, de la **primul** element până la **ultimul**, folosind un indice **i** care ia valori de la **0** la **$n-1$** (n – fiind **dimensiunea** șirului, **i** – **poziția** elementului, **$v[i]$** – **valoarea** elementului de pe poziția **i**).

Există, însă, situații când parcurgerea se realizează și altfel: de la ultimul element al șirului la primul sau până la un anumit element al șirului.



Exemplu:

```
for (i = 0; i < n; i++) // de la primul la ultimul element al șirului
    // v[i] se citește, afișează sau prelucrează
for (i = n - 1; i >= 0; i--) // de la ultimul la primul element al șirului
    // v[i] se citește, afișează sau prelucrează
```




Dacă trebuie prelucrate doar anumite elemente (Ex.: cele de pe poziții pare/impare ș.a.) **i** poate lua valori din 2 în 2, 3 în 3 etc.



Exemplu în care se poate utiliza un șir de valori

Pentru a putea determina cea mai mare notă a unui elev la o disciplină, se folosește un șir de valori care reține toate notele acestuia și, parcurgând șirul, se poate determina nota maximă obținută de elev.

note[5]	5	3	9	7	8
	0	1	2	3	4
			↓		
			nota		
			maximă		



Rețineți!

- 📖 Șirul de valori este o colecție de elemente de același tip.
- 📖 NU se pot efectua operații doar asupra întregului șir de valori deodată, ele se efectuează asupra fiecărui element în parte.
- 📖 Parcurgerea șirului se face utilizând o structură repetitivă.



Aplicație!

Dați 3 exemple de situații în care puteți utiliza șiruri de valori.



Fișa de lucru 23

Creați dosarul numit **Fisa23** și rezolvați exercițiile de mai jos într-un fișier **raspunsuri.docx**. Salvați fișierul în dosarul creat.

Alegeți varianta corectă de răspuns:

1. Care dintre următoarele declarații este corectă?

- a. double 25v[];
- b. float v4[int x];
- c. char [45]t;
- d. int t[45];

2. Stabiliți valoarea ultimului element al șirului, în urma executării secvenței alăturate:

- a. 1;
- b. 2;
- c. 6;
- d. 4.

```
for (i = 0; i < 5; i++)
    v[i] = i + 2;
```

3. Stabiliți valoarea primului element al șirului, în urma executării secvenței alăturate:

- a. 17;
- b. 7;
- c. 16;
- d. 11.

```
v[5] = 2;
for (i = 5; i > 0; i--)
    v[i-1] = v[i] + i;
```

4. Care dintre următoarele declarații definește un șir de valori reale cu maximum 15 elemente?

- a. double 15v[];
- b. int v15[float];
- c. double v[15];
- d. int v[15];

Operații cu șiruri de valori



1. Citirea unui șir de valori cu număr cunoscut de elemente presupune *citirea numărului de elemente* și *citirea valorii fiecărui element* din șir, de la primul până la ultimul.



Exemplu:

```
int v[25], i, n;
cout<<" Dați dimensiunea șirului ";
cin>>n; // n<=25
for (i = 0; i < n; i++)
{
    cout<<"v["<<i<<"]="";
    cin>> v[i]; // citirea fiecărui element din șir
}
```



▪ Dimensiunea șirului nu poate depăși valoarea precizată la declararea șirului ($n \leq 25$).

▪ Valoarea elementului din șir de pe poziția i este $v[i]$.

▪ Se citesc pe rând valorile elementelor din șirul de valori.

2. Citirea unui șir de valori cu număr necunoscut de elemente – În anumite situații, numărul de valori care urmează a fi citite nu este cunoscut, deoarece citirea se face ținând cont de anumite condiții/reguli (de ex. citirea numerelor până se întâlnește valoarea 0). În aceste situații, citirea se realizează parcurgând următorii pași: *inițializarea cu 0 a indicelui folosit pentru adăugare în șir, citirea fiecărei valori până la întâlnirea uneia care nu respectă condiția, adăugarea valorii în șir și modificarea indicelui/poziției de adăugare, pentru următorul element care va fi adăugat, iar la final stabilirea dimensiunii întreg șirului.*



Exemplu: Citirea unor numere până la întâlnirea valorii 0.

```
int v[100], i, n, x;
i = 0;
do
{
    cout<<"Dați un element ";
    cin>> x;
    v[i] = x; // introducerea elementului în șir
    i++; // modificarea indicelui și implicit a
        dimensiunii șirului
} while (x != 0); // condiția de oprire a citirii
n = i-1; // dimensiunea finală a șirului de
        valori fără ultima poziție
```



▪ Valorile se citesc și se adaugă în șirul de valori, până la întâlnirea valorii 0.

▪ După introducerea unui element în șir, se modifică valoarea indicelui și, implicit, și dimensiunea șirului, care inițial este 0.

▪ La final, după ce au fost introduse toate valorile, se reține în n dimensiunea șirului.



3. Afișarea unui șir de valori se realizează parcurgând elementele de la *primul* până la *ultimul* și afișând valorile, rând pe rând. Afișarea se poate face și de la ultimul element (cel de pe poziția $n-1$) la primul (de pe poziția 0). În funcție de cerințe se pot afișa doar anumite elemente ale șirului.



Exemple:

```
for (i = 0; i < n; i++) // de la primul la ultimul element
    cout<< v[i] <<" "; // afișarea fiecărui element din șir
for (i = n - 1; i >= 0; i--) // de la ultimul la primul element
    cout<< v[i] <<" "; // afișarea fiecărui element din șir
```



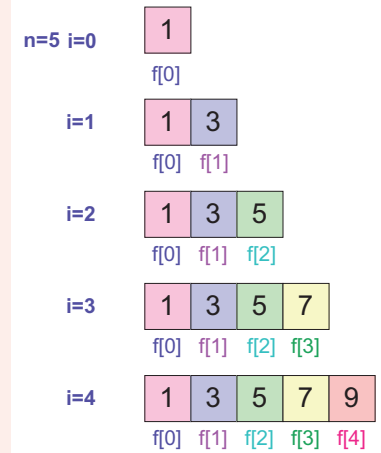
4. Generarea unui șir de valori – În anumite situații, șirul de valori nu este cunoscut și trebuie creat, element cu element, ținând cont de anumite condiții/reguli. Există șiruri ale căror elemente se generează începând cu primul, în funcție de diverse reguli (de ex. generarea șirului cu primele 50 numere pare/impare etc.), dar există șiruri ale căror elemente se generează utilizând alte elemente din șir (de ex. un element este dublul precedentului, un element este suma vecinilor ș.a.).



EX.

Exemplu: Generarea unui șir care conține primele 5 numere impare:

```
int f[30], i, n, x;
i = 0;
n = 5; // numărul de numere impare
x = 1; // primul număr impar
do
{
    if (x%2 != 0) // se verifică condiția cerută
    {
        f[i] = x; // adăugăm elementul găsit în șir
        i++; // trecem la următoarea poziție în șir
    }
    x++; // se trece la căutarea următorului element
} while (i < n);
```

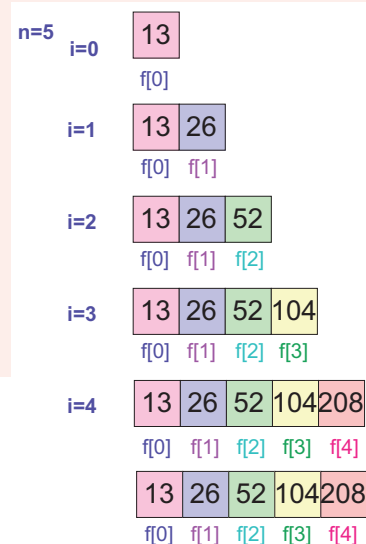


- Se inițializează cu 0 poziția *i*, în care se adaugă elementul;
- Se pornește de la primul număr impar *x=1*;
- Se caută următorul număr impar existent, mărinnd cu 1 mereu valoarea lui *x*;
- Elementul impar găsit este adăugat în șirul de valori pe poziția *i*, după care se mărește valoarea lui *i* cu 1, pentru ca următorul element găsit să se adauge în următoarea poziție în șir;
- Generarea se termină când *i* devine 5, deci s-au adăugat 5 numere în șir, în pozițiile de la 0 la 4.

EX.

Exemplu: Generarea unui șir cu *n* elemente, cu primul element cunoscut, iar următoarele elemente egale cu dublul elementului precedent:

```
int f[30], i, n;
n = 5; // numărul de elemente ale șirului
f[0] = 13; // primul număr cunoscut
i = 1;
while(i < n)
{
    f[i] = f[i-1]*2; // adăugăm elementul determinat în șir
    i++; // trecem la următoarea poziție în șir
}
```



Rețineți!

Operațiile care se pot efectua cu șirurile de valori sunt:

- 📖 Citirea unui șir cu număr cunoscut/necunoscut de elemente.
- 📖 Afișarea elementelor.
- 📖 Generarea elementelor unui șir.
- 📖 Parcurgerea și prelucrarea elementelor șirului.



Fișa de lucru 24

Creați dosarul numit **Fisa24** și scrieți, în limbajul C++, un program pentru rezolvarea următoarelor probleme. Salvați fișierele realizate pe parcurs în dosarul creat.



1. Dominik este un regizor foarte ordonat, care notează într-o agendă numărul de cadre pentru fiecare filmare realizată de el. Cunoscând numărul de cadre pentru n filmări, ajutați-l pe Dominik să vizualizeze, în ordine inversă, fiecare număr de cadre notat în agendă. (Ex.: $n=7$ 224, 102, 56, 345, 213, 34, 652 – se va afișa 652 34 213 345 56 102 224)

2. Albinuța Bzzz zboară pe un câmp cu n flori. La fiecare floare pe care se așează, aceasta petrece un anumit timp, exprimat în minute. Cunoscând timpul pe care îl poate petrece la fiecare floare și faptul că albinuța se oprește tot la a 5-a floare, începând cu prima, afișați pe ecran, separați prin spații, timpii pe care îi va petrece la fiecare floare la care s-a oprit. (Ex.: $n=17$ 4, 5, 7, 3, 11, 5, 8, 4, 3, 1, 2, 7, 4, 7, 3, 4, 5 – se va afișa 4 5 2 4)



3. Se consideră următorul șir, în care n este un număr natural nenul:

$$f_n = \begin{cases} 0, & \text{dacă } n = 1 \\ 5, & \text{dacă } n = 2 \\ 2 * f_{n-1} - f_{n-2} + 5, & \text{dacă } n > 2 \end{cases}$$

Generați și afișați, separate prin spații, primele n valori ale acestui șir. (Ex.: $n=5$ – se va afișa 0 5 15 30 50)

4. Generați și afișați un șir cu n valori, în ordine crescătoare, cu condiția ca elementele să fie divizibile cu 4. Se cunoaște primul element ca fiind x , o valoare multiplu de 4, citită de la tastatură. (Ex.: $n=7, x=8$ – se va afișa 8 12 16 20 24 28 32)



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa24**.

	Da 	Nu 	Nu sunt sigur(ă) 
La finalul lecției, știu...			
1. ce este un șir de valori.			
2. cum să declar, să accesez și să parcurg un șir de valori.			
3. cum se citește un șir cu număr cunoscut/necunoscut de elemente.			
4. cum se afișează elementele unui șir de valori.			
5. cum se generează un șir de valori.			

6. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 18. Algoritmi de bază pentru șiruri de valori – numărare, sumă/produs și maxim/minim

i Obținerea unei sume/produs utilizând elementele unui șir de valori

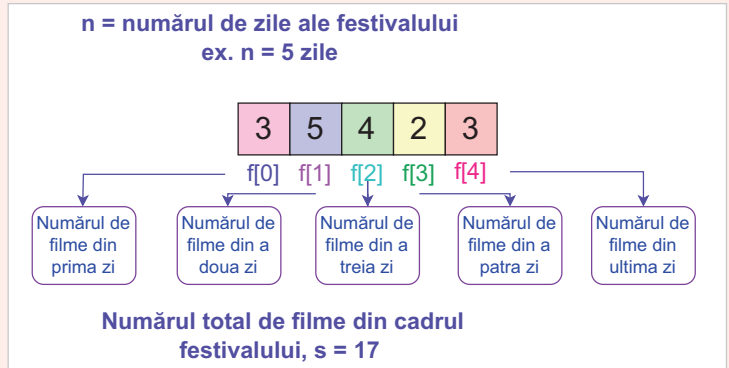
Algoritmul presupune parcurgerea elementelor de la primul la ultimul și calculul sumei, respectiv al produsului, utilizând o variabilă care e inițializată cu **0**, în cazul sumei, respectiv cu **1** în cazul unui produs.

EX. **Exemplu:** Ema și Freya participă la festivalul internațional de filme **CineWorld**, care se desfășoară pe parcursul a n zile. Cunoscând numărul de filme care rulează în fiecare zi, afișați numărul total de filme rulate pe parcursul festivalului.



```
#include <iostream>
using namespace std;
int main()
{
    unsigned int f[25], i, n, s = 0;
    cout<<" Câte zile durează festivalul? ";
    cin>>n;
    for (i = 0; i < n; i++) // citirea fiecărui element din șir
    {
        cout<<"f["<<i<<"]="";
        cin>> f[i];
        s = s + f[i]; // la s se adună elementul curent
    }
    cout<<" Numărul total de filme este " << s;
    return 0;
}
```

EX. **Exemplu de valori** pentru algoritmul de mai sus: dacă numărul de zile ale festivalului este 5, șirul de valori va avea 5 poziții. Pentru cele 5 zile se memorează numărul de filme care au rulat zilnic. Odată cu citirea șirului, urmează și adunarea elementelor pentru a determina valoarea lui s , care reprezintă numărul total de filme din cadrul festivalului.



i Determinarea valorii minime, respectiv maxime a unui șir

Algoritmul constă în parcurgerea elementelor de la primul la ultimul și determinarea minimului, respectiv a maximumului, utilizând o variabilă care se inițializează cu prima valoare din șir (presupunem că această valoare e cea mai mică, respectiv cea mai mare, și o modificăm dacă întâlnim în șir o valoare mai mică, respectiv mai mare).

EX.

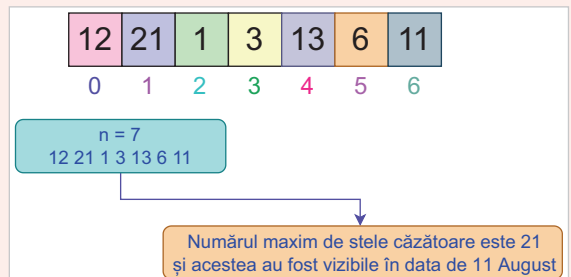
Exemplu: Anual, luna august e luna în care se poate vedea cel mai cunoscut curent de meteori, și anume *Perseidele*. Crina și Anda, pasionate fiind, urmăresc *Perseidele* n seri la rând și vor ca apoi să precizeze care a fost seara cu cele mai multe stele căzătoare văzute, precum și numărul acestora, cunoscând faptul că prima seară e 10 august.



```
#include <iostream>
using namespace std;
int main()
{
    unsigned int perseide[21], i, n, maxi, pmax;
    cout<<" Specificați numărul de nopți în care se urmăresc Perseidele ";
    cin>>n; // n<=21
    for (i = 0; i < n; i++) // citirea fiecărui element din șir
    {
        cout<<"perseide["<<i<<"]="";
        cin>>perseide[i];
    }
    maxi = perseide[0];
    pmax = 0;
    for (i = 1; i < n; i++) // parcurgerea șirului
        if (perseide[i] > maxi) // se întâlnește o valoare mai mare
        {
            maxi = perseide[i]; // maxi reține noua valoare
            pmax = i; // se reține poziția maximului
        }
    cout<<" Numărul maxim de stele căzătoare este " << maxi;
    cout<<" și acestea au fost vizibile în data de " << 10 + pmax<<" August";
    return 0;
}
```

EX.

Exemplu de valori pentru algoritmul anterior: dacă numărul de seri este 7, șirul de valori are 7 poziții pentru care se rețin numerele de stele căzătoare văzute în fiecare seară. Variabila **maxi** reține prima valoare din șir și e comparată apoi cu celelalte elemente ale șirului. La întâlnirea unei noi valori maxime se schimbă și poziția acesteia **pmax** pentru a putea calcula data. Ziua se calculează adunând numărul 10 la poziția în care se află valoarea maximă.



Probleme de numărare/frecvență (problemele de frecvență – conținut opțional)

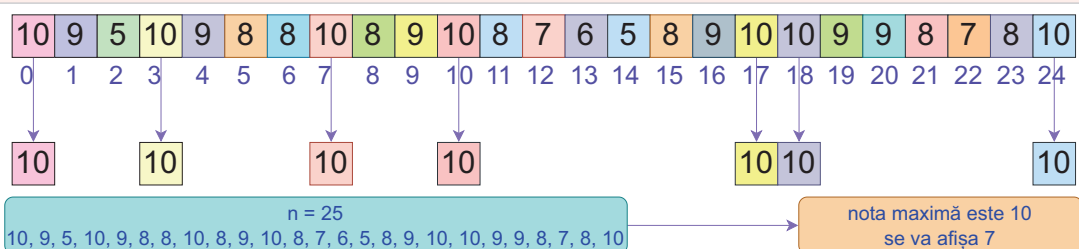
Pentru a număra elementele care îndeplinesc o condiție cerută (de ex. numărul de valori pozitive dintr-un șir) se utilizează o variabilă cu **rol de contor**, care are inițial valoarea 0.



EX. **Exemplu:** La finalul unei săptămâni, profesorul de sport face o statistică a ultimelor note date la o clasă cu n elevi, la disciplina „Educație fizică și sport”. Acesta dorește să știe câți copii au obținut nota cea mai mare. (Ex. $n=25$ 10, 9, 5, 10, 9, 8, 8, 10, 8, 9, 10, 8, 7, 6, 5, 8, 9, 10, 10, 9, 9, 8, 7, 8, 10 – se va afișa 7)

```
#include <iostream>
using namespace std;
int main()
{
    unsigned int el[35], i, n, max, nr = 0;
    cout<<" Specificați numărul de elevi ";
    cin>>n;
    for (i = 0; i < n; i++) // citirea fiecărui element din șir
    {
        cout<<"el["<<i<<"]="";
        cin>> el[i];
    }
    max = el[0]; // considerăm prima notă ca fiind cea mai mare și compăram max cu celalalte note
    for (i = 1; i < n; i++)
        if (el[i] > max) // se întâlnește o notă mai mare
            max = el[i]; // max reține noua valoare
    for (i = 0; i < n; i++)
        if (el[i] == max) // se numără câți elevi au nota maximă
            nr ++;
    cout<<" Numărul elevilor cu nota maximă este " << nr;
    return 0;
}
```

EX. **Exemplu de valori pentru algoritmul anterior:** dacă numărul de elevi din clasă este 25, șirul de valori are 25 poziții în care se rețin notele fiecărui elev. Cu ajutorul variabilei **max** se determină nota maximă, iar **nr** este variabila contor în care se reține câte note maxime sunt în șir.



CONȚINUT FACULTATIV

Există situații în care nu se poate realiza numărarea cu ajutorul unei singure variabile (de ex. numărul de apariții ale cifrelor într-un număr). În astfel de situații se utilizează un șir de valori care poartă numele de **șir de frecvență**.

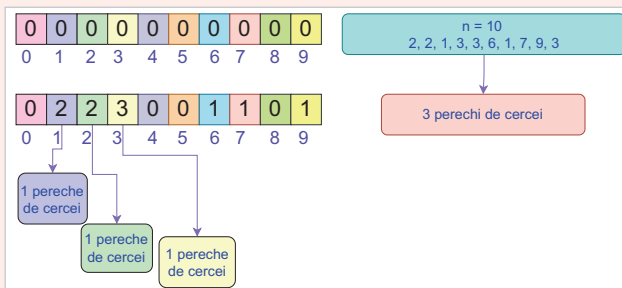
În acest șir indicii reprezintă valorile pentru care se vor stabili numărul de apariții, iar elementele șirului sunt inițializate cu 0 în momentul declarării șirului, utilizând atribuirea **șir[dimensiune]={0}**. Treptat, în momentul în care întâlnim un element **i**, valoarea de pe poziția **i** crește cu o unitate.

EX. **Exemplu:** Arian lucrează la un magazin universal, la departamentul de bijuterii. Fiecare tip de cercei din magazin are asociat o **cifră**, reprezentând un cod pentru a identifica modelul din care face parte acesta. Având un șir cu **n** astfel de coduri, stabiliți câte perechi de cercei se pot forma.

```
#include <iostream>
using namespace std;
int main()
{
    unsigned int c[10]={0}, i, n, cod, perechi = 0;
    cout<<"Specificați numărul de cercei ";
    cin>>n;
    for (i = 0; i < n; i++)
    {
        cout<<"Introduceți codul cerceiului ";
        cin>> cod; // se citește codul cerceiului i
        c[cod] ++; // se contorizează codul
    }
    for (i = 0; i < 10; i++)
        perechi = perechi + c[i]/ 2; // determinăm numărul de perechi
    cout<<" Numărul perechilor de cercei este " << perechi;
    return 0;
}
```



EX. **Exemplu de valori** pentru algoritmul anterior: numărul de cercei este 10. Cunoscând faptul că cerceii au coduri de o cifră, se utilizează un șir de frecvență cu indici de la 0 la 9, pentru posibilele coduri. Șirul este inițializat cu valoarea 0 și își modifică valorile în funcție de codul fiecărui cercei. În variabila **perechi** se calculează numărul de perechi care se pot forma, împărțind la 2 fiecare element din șir pentru care se rețin numerele de cercei cu un anumit cod (pentru a forma o pereche sunt necesari 2 cercei cu același cod).

**Rețineți!**

- 📖 Suma se inițializează cu 0, iar produsul cu 1.
- 📖 Maximul/minimul se inițializează cu primul element din șirul de valori.
- 📖 Numărarea presupune utilizarea unei variabile **contor**, care se inițializează cu valoarea 0.



Fișa de lucru 25

Creați dosarul numit **Fisa25** și scrieți, în limbajul C++, un program pentru rezolvarea următoarelor probleme. Salvați fișierele în dosarul creat.

1. Aurelian este pasionat de anotimpuri, în special de primăvară. El măsoară temperaturile în mai multe zile, la prânz, până până întâlnește o temperatură negativă. În urma acestor măsurători, el va preciza care este temperatura medie a valorilor pozitive din aceste zile. (Ex.: 10, 8, 11, 11, 4, -1 – se va afișa 8.8)



2. La festivalul lălelelor, Laura, pasionată de flori, se plimbă pe la fiecare stand, din cele n existente, pentru a determina care este numărul **minim**, respectiv **maxim** de lălele care pot fi găsite la standuri (minim 2 lălele/stand). În plus, ea dorește să rețină și numerele standurilor unde se găsesc cele mai multe, respectiv cele mai puține lălele, știind că numerotarea standurilor începe de la 1. (Ex.: $n=6$ - 4, 7, 13, 3, 5, 16 – se va afișa: **standul 4** – nr. lălele: 3 și **standul 6** – nr. lălele: 16)

3. Ilinca merge la cumpărături pentru sezonul de vară. Mama i-a dat o sumă de bani pentru a cumpăra n piese vestimentare. Cunoscând prețul fiecărei piese vestimentare, determinați numărul de piese cu prețul cel mult egal cu 15 lei. (Ex.: $n=5$ - 10, 12, 10, 45, 20 – se va afișa 3 piese)

4. Pornind de la secvența alăturată de algoritm, realizați modificările necesare pentru a afișa minimul și pozițiile pe care acesta se află.

```
maxi = x[1];
for (i = 2; i <= n; i++)
    if (x[i] > maxi)
    {
        maxi = x[i];
        pmax = i;
    }
cout<<" Numărul maxim este " << maxi;
cout<<" și se află pe poziția " << pmax;
```



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa25**.

	Da	Nu	Nu sunt sigur(ă)
<i>La finalul lecției, știu...</i>			
1. cum să calculez suma/produsul utilizând elementele unui șir.			
2. cum să determin minimul/maximul într-un șir de valori.			
3. cum să număr elementele dintr-un șir care au anumite proprietăți.			

4. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 19. Algoritmi de bază pentru șiruri de valori – verificare a unei proprietăți

Verificarea unei proprietăți îndeplinită de elementele unui șir de valori



Pentru a verifica dacă elementele unui șir de valori respectă o anumită proprietate (de ex. numere pare/impare, divizibile cu o valoare, numere prime ș.a.), acesta se parcurge, se verifică proprietatea pentru fiecare element, după care, elementul se contorizează, se afișează sau se efectuează diverse operații cu el.

EX.

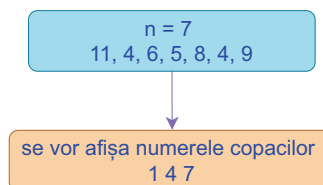
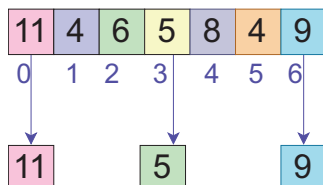
Exemplu: Paula și Șerban se plimbau prin parc într-o seară liniștită, trecând pe lângă n copaci. În fiecare copac exista un număr de bufnițe. Ei vor să cunoască numerele copacilor care au un număr impar de bufnițe, știind că numerotarea copacilor începe de la 1. Afișați numerele copacilor în care stau un număr impar de bufnițe, separate printr-un spațiu. (Ex.: $n=7$ – cu elementele: 11, 4, 6, 5, 8, 4, 9 – se va afișa 1 4 7)



```
#include <iostream>
using namespace std;
int main()
{
    unsigned int buf[25], i, n;
    cout<<" Specificați numărul de copaci! ";
    cin>>n;
    for (i = 0; i < n; i++) // citim elementele din șir
    {
        cout<<"buf["<<i<<"]=";
        cin>> buf[i];
    }
    for (i = 0; i < n; i++) // parcurgem elementele din șir
        if (buf[i] % 2 != 0) // verificăm proprietatea cerută
            cout<<i+1 <<" ";
    return 0;
}
```

EX.

Exemplu de valori posibile pentru algoritmul de mai sus: dacă numărul de copaci este 7, șirul de valori are 7 poziții în care se rețin numerele de bufnițe din fiecare copac. Se parcurge șirul de valori și se afișează numărul copacilor care conțin valori impare de bufnițe.



Verificarea existenței unor valori într-un șir

Pentru a verifica dacă într-un șir există valori care îndeplinesc o anumită condiție, s-ar putea număra elementele din șir care respectă proprietatea, iar în cazul în care acest număr este **nenul**, putem spune că **există elemente în șirul de valori care respectă proprietatea/condiția**.



Secvență parțială C++

```
int nr = 0;
for (i = 0; i < n; i++)
    if (v[i] – respectă proprietatea)
        nr ++;
if (nr != 0)
    Există
else
    Nu există
```



O altă metodă ar fi utilizarea unei variabile logice (de tip **bool**) și pașii ar fi următorii:

- ♦ inițializarea variabilei cu **false** (**0**) (presupunând că nu există vreun element cu proprietatea dată)
- ♦ parcurgerea șirului de valori, element cu element
 - dacă elementul **respectă** proprietatea, variabila logică devine **true** (**1**)
- ♦ în secvența din dreapta, parcurgerea șirului de valori se oprește datorită instrucțiunii **break** în momentul în care se descoperă un element conform.

Secvență parțială C++

```
bool ok = false;
for (i = 0; i < n; i++)
    if (v[i] – respectă proprietatea)
        ok = true;
if (ok == true)
    Există
else
    Nu există
```

Secvență parțială C++

```
bool ok = false;
for (i = 0; i < n; i++)
    if (v[i] – respectă proprietatea)
    {
        ok = true;
        break; // iese din structura
                repetitivă for
    }
if (ok == true)
    Există
else
    Nu există
```

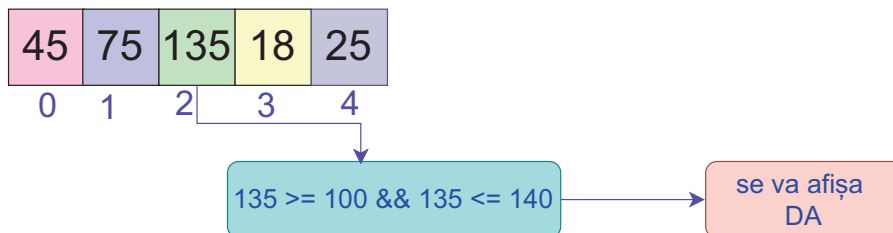


Exemplu: Pe o planșă sunt reprezentate diferite corpuri geometrice: piramide triunghiulare regulate, piramide patrulatere, cuburi, prisme triunghiulare ș.a. Cunoscând aria fiecărui corp, exprimată în m^2 , și faptul că ultimul corp are aria egală cu $25 m^2$, verificați dacă pe planșă există corpuri cu arii având valori cuprinse între $100 - 400 m^2$. (Ex. 45, 75, 135, 18, 25 – se va afișa DA, deoarece există aria $135 \geq 100 \ \&\& \ 135 \leq 400$)



```
#include <iostream>
using namespace std;
int main()
{
    unsigned int c[100], i = 0, n, arie;
    bool ok = false; // se presupune că nu există valori
    do
    {
        cout<<"Introdu aria unui corp geometric!";
        cin>> arie; // se citește fiecare valoare
        c[i] = arie; // se adaugă valoarea în șir
        i++;
    } while (arie != 25);
    n = i - 1;
    for (i = 0; i < n; i++) // se parcurg valorile până se ajunge la final
        if (c[i] >= 100 && c[i] <= 400) // se verifică dacă elementul respectă proprietatea
            ok = true;
    if (ok == true)
        cout<<" DA ";
    else
        cout<<" NU ";
    return 0;
}
```

EX. **Exemplu de valori** pentru algoritmul anterior: se presupune că *nu există* nicio arie cu valori între 100-400 m² (*ok=false*). Șirul de valori conține ariile din imaginea de mai jos. Se parcurge șirul de valori și în cazul în care există o arie cuprinsă între 100 și 400 presupunerea făcută se modifică (*ok=true*). După parcurgerea întregului șir de valori se verifică dacă s-a găsit cel puțin un element cu aria între 100 și 400 și se afișează un mesaj corespunzător.



Verificarea unei proprietăți comune pentru toate elementele dintr-un șir de valori

Pentru a determina dacă toate elementele unui șir îndeplinesc o anumită condiție/proprietate, o posibilă rezolvare ar putea fi numărarea elementelor din șir care respectă proprietatea, iar în cazul în care numărul de elemente găsit este **egal** cu numărul total de elemente **atunci toate elementele din șirul de valori respectă proprietatea**.



Secvență parțială C++

```
int nr = 0;
for (i = 0; i < n; i++)
    if (v[i] – respectă proprietatea)
        nr ++;
if (nr == n)
    Proprietatea este comună
    elementelor
else
    Proprietatea NU este comună
    tuturor elementelor
```



O altă metodă ar fi utilizarea unei variabile logice (de tip **bool**) și pașii ar fi următorii:

- ♦ inițializarea variabilei cu **true** (1) (presupunând că toate elementele respectă proprietatea dată)
- ♦ parcurgerea șirului de valori, element cu element
 - dacă elementul **nu respectă** proprietatea variabila logică devine **false** (0)
- ♦ în secvența din **dreapta** parcurgerea șirului se oprește datorită instrucțiunii **break** (care întrerupe repetarea/programul iese din structura repetitivă), în momentul în care s-a găsit un element neconform.

Secvență parțială C++

```
bool ok = true;
for (i = 0; i < n; i++)
    if (v[i] – nu respectă proprietatea)
        ok = false;
if (ok == true)
    Proprietatea este comună
    elementelor
else
    Proprietatea NU este comună
    tuturor elementelor
```

Secvență parțială C++

```
bool ok = true;
for (i = 0; i < n; i++)
    if (v[i] – nu respectă proprietatea)
    {
        ok = false;
        break;
    }
if (ok == true)
    Proprietatea este comună
    elementelor
else
    Proprietatea NU este comună
    tuturor elementelor
```

EX.

Exemplu: Pe postul de radio CloudyFM, DJ Victor pregătește zilnic melodiile pentru mixaj înainte de a fi difuzate. El are n piese cu numărul **bpm** (beats per minute) specificat pentru fiecare piesă. Știind că piesele se pot mixa doar dacă au toate același număr de **bpm**, specificați, printr-un mesaj, dacă toate piesele se pot mixa sau nu. (Ex.: $n=7$ 108, 108, 108, 108, 108, 108, 108 – se afișează DA; Ex. $n=5$ 108, 108, 128, 128, 108 – se afișează NU)



```

#include <iostream>
using namespace std;
int main()
{
    unsigned int m[100], i, n;
    bool ok = true; // se presupune că toate valorile îndeplinesc condiția cerută
    cout<<" Specificați numărul de piese! ";
    cin>>n;
    for (i = 0; i < n; i++) // se citesc valorile din șir
    {
        cout<<"Introdu bpm ";
        cin>> m[i];
    }
    for (i = 1; i < n; i++)
        if (m[i - 1] != m[i]) // se verifică dacă există valori care NU îndeplinesc condiția cerută
            ok = false; // presupunerea făcută devine falsă
    if (ok == true)
        cout<<" DA ";
    else
        cout<<" NU ";
    return 0;
}

```

EX. **Exemplu:** pentru algoritmul de mai sus: se presupune că toate piesele au același număr de **bpm** (**ok=true**). Se parcurge șirul de valori și în cazul în care există două numere diferite presupunerea făcută devine falsă (**ok=false**). După parcurgerea întregului șir de valori se verifică dacă toate piesele au același număr de **bpm** și se afișează un mesaj corespunzător.

108	108	108	108	108	108	108
0	1	2	3	4	5	6

n = 7
108, 108, 108, 108, 108, 108, 108

se afișează
DA



Rețineți!

📖 Pentru a verifica dacă **într-un șir există anumite valori**, acestea se pot număra sau se poate folosi o variabilă logică presupunând că, inițial, nu există asemenea valori. Variabila își va schimba valoarea de adevăr dacă se întâlnesc valorile căutate.

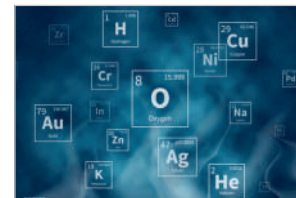
📖 Pentru a verifica dacă **toate elementele unui șir respectă o anumită condiție** se pot număra toate elementele șirului care respectă condiția, după care se compară numărul găsit cu numărul de elemente din șir. O altă variantă este folosirea unei variabile logice, presupunând că toate elementele respectă condiția. Variabila își va schimba valoarea când se găsește un element care nu respectă condiția.



Fișa de lucru 26

Creați dosarul numit **Fisa26** și scrieți, în limbajul C++, un program pentru rezolvarea următoarelor probleme. Salvați fișierele în dosarul creat.

(*) Ioana și Iulia, pasionate de chimie, notează într-un șir de valori masele atomice a n elemente din tabelul periodic. Ajutați cele două fete să verifice **dacă există elemente cu masa atomică număr întreg**. **Indicație:** Pentru a verifica dacă un număr este întreg se utilizează operatorul de conversie la tip de valoare întreagă: **(int)x==x**. (Ex.: $n=5$ - 58.69, 69.72, 101.1, 197, 107.9 – se va afișa DA)



2. (***) Cu ocazia zilei internaționale a prieteniei, copiii dintr-o clasă, s-au jucat următorul joc: clasa a fost împărțită în n echipe, a câte doi copii fiecare. Fiecare echipă a primit o foaie pe care cei doi copii au scris fiecare câte un număr. Se face o analiză a numerelor scrise pe foi și se verifică dacă numerele scrise de elevi pe fiecare foaie pot fi **prietene**. Două numere se numesc **prietene** dacă fiecare număr în parte este suma divizorilor (cei proprii și 1) celuilalt număr. La finalul jocului specificați **dacă a existat** o pereche de astfel de numere. (Ex.: $n=12$ perechile – 13, 2; 134, 4; 220, 284; 7, 9; 25, 13; 118, 121; 5, 77; 10, 55; 234, 99; 11, 10; 90, 432; 258, 40 – se va afișa DA, deoarece numerele din perechea (220 și 284) sunt numere prietene – 220 ($1+2+4+5+10+11+20+22+44+55+110=284$) și 284 ($1+2+4+71+142=220$))

3. Livia realizează un grafic pentru a determina dinamica populației, reprezentând valori din bilanțul natural și bilanțul migratoriu din mai multe zone. Având în grafic n valori, stabiliți **dacă șirul conține doar elemente pozitive**. (Ex.: $n = 4$, -10, -4, 22, 39 – se va afișa NU)



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa26**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. să verific o proprietate a unui element dintr-un șir.			
2. să determin existența unor valori cu anumite proprietăți într-un șir de valori.			
3. să verific o proprietate comună pentru toate elementele dintr-un șir de valori.			

4. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Lecția 20. Codebot – controlul unui robot didactic virtual

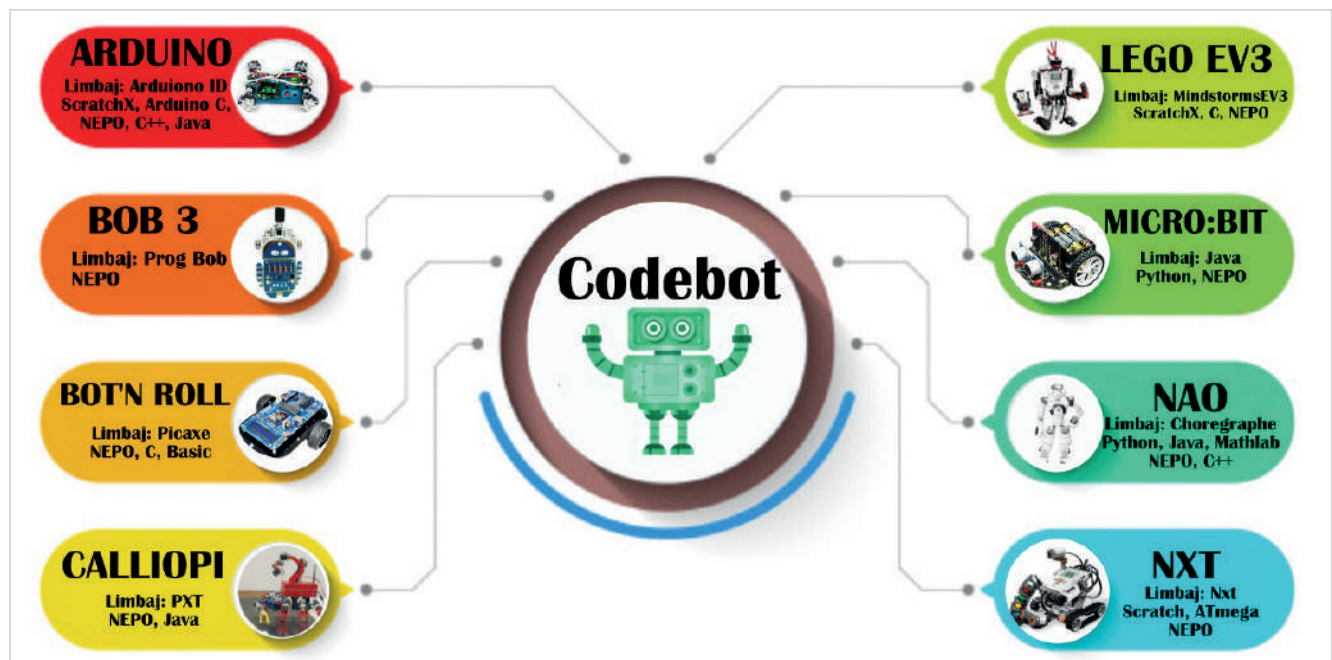


Să ne reamintim!



Un robot este un sistem compus din mai multe elemente mecanice, senzori, actuatori și un mecanism de direcționare. Elementele mecanice arată înfățișarea robotului, precum și mișcările posibile ale acestuia. Senzorii și actuatorii sunt folosiți la interacțiunea cu mediul sistemului, iar mecanismul de direcționare reglează motoarele și pregătește mișcările care urmează a fi efectuate.

O parte din roboții virtuali care pot fi programați folosind limbaje de programare sub formă de blocuri sau linie de cod sunt prezentați mai jos:



Programarea acestor roboți se poate face cu ajutorul mai multor limbaje de programare.



Open Roberta Lab este o platformă care permite programarea online de roboți reali sau virtuali, se poate accesa la adresa <https://lab.open-roberta.org>, este disponibilă în 21 limbi și acceptă conectarea următoarelor componente hardware:

- | | | | |
|----------------------|-----------------------|------------------------|---|
| ▪ Bionic Flower | ▪ EV3 dev | ▪ Nepo4Arduino Uno | ▪ Robotino |
| ▪ Bionics Kit | ▪ mBot | ▪ Nepo4Nano33BLE | ▪ senseBox |
| ▪ BOB 3 | ▪ mBot 2 | ▪ NXT | ▪ Spike Prime / Robot Inventor / Lego |
| ▪ Bot'n Roll | ▪ micro:bit | ▪ Open Roberta Sim | ▪ Spike Prime / Robot Inventor / Pybricks |
| ▪ Calliope mini | ▪ micro:bit Joy – car | ▪ EV3 leJOS 0.9.1 | ▪ Thymio |
| ▪ Calliope mini 2016 | ▪ micro:bit V2 | ▪ Open Roberta xNN | ▪ TxT 4.0 Controller |
| ▪ Calliope mini v3 | ▪ NAO | ▪ RCJ RescueOnline-Sim | ▪ WeDo |
| ▪ Edison V2 | ▪ Nepo4Arduino Mega | ▪ ROB3RTA | |
| ▪ Edison V3 | ▪ Nepo4Arduino Nano | | |

i Interfața platformei cuprinde un meniu principal pentru **Open Roberta Lab** și un meniu pentru simulatorul **Open Roberta SIM**. Meniul principal pentru **Open Roberta Lab** permite:

- Activarea robotului, realizarea principalelor operații asupra unui proiect (creare, salvare etc.);
- Selectarea robotului și vizualizarea informațiilor despre acesta;
- Obținerea de asistență în utilizarea platformei;
- Vizualizarea de tutoriale și programe dintr-o bibliotecă educațională.

Meniul **Open Roberta Sim** permite vizualizarea acțiunilor întreprinse de robot, fără ca acesta să fie neapărat conectat.

▶ Limbajul **NEPO**, utilizat pentru programare pe platforma **Open Roberta Lab**, este un limbaj care folosește simulatorul integrat pentru roboții virtuali și kit-urile pentru roboții reali.

Simulatorul integrat permite utilizarea unui robot virtual, astfel încât e posibilă programarea acestuia doar cu ajutorul calculatorului și a unei conexiuni la Internet, nefiind necesară achiziționarea unui robot.

▶ Limbajul de programare **NEPO (New Easy Programming Online)** este un limbaj simplu, care utilizează blocuri din biblioteca **Blockly**, la care se adaugă anumite elemente din **Scratch**.

PROGRAM NEPOprog ROBOT CONFIGURATION XNNbasis NEURAL NETWORK Define NEURAL NETWORK Learn

- Action**: Conține blocuri care permit controlul direct al robotului, oferindu-i instrucțiuni specifice de mișcare, sunet și lumină.
- Sensors**: Conține blocuri pentru senzorii robotului: de atingere, ultrasonic, de culoare, de infraroșu, butoane, giroscopie, de rotație și timer.
- Control**: Conține blocuri de control, cum ar fi: structura alternativă (IF), repetitivă (REPEAT) și de așteptare (WAIT).
- Logic**: Conține blocuri pentru crearea expresiilor logice folosind operatori relaționali, de egalitate și logici (AND, OR, T, F).
- Math**: Conține blocuri pentru crearea expresiilor folosind operatori aritmetici, funcții (sin, square root etc.) și numere.
- Neural Network**: Conține blocuri care permit utilizatorilor să creeze și să antreneze rețele neuronale artificiale simple.
- Text**: Conține blocuri pentru scrierea unor mesaje pe ecranul robotului și adăugarea unor comentarii explicative.
- Lists**: Conține blocuri pentru crearea unor liste: numerice, boolean, șiruri de caractere, culori și conexiuni.
- Colours**: Conține blocuri de culoare (gri, negru, albastru, verde, galben, roșu, alb și maro) utilizate pentru senzori.
- Variables**: Conține blocuri pentru declararea de variabile.
- Functions**: Conține blocuri pentru crearea unor funcții cu returnarea sau nu a unor valori (parametri).
- Messages**: Conține blocuri de conectare la alt robot, de trimitere și primire de mesaje prin Bluetooth.

Fiecare program începe implicit cu blocul **start**.

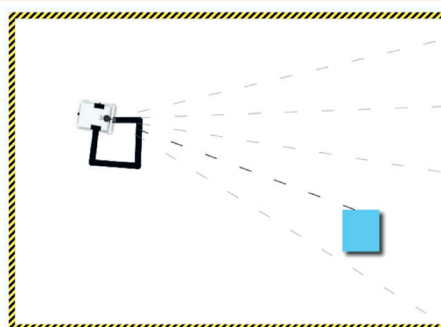
+ start **show sensor data**

EX. **Exemplu:** Deplasarea unui robot **NXT** pe o suprafață, trasarea unui pătrat în timpul deplasării acestuia precum și redarea unor note muzicale la pornire și la oprire.

```

+ start
  play whole note c
  repeat 4 times
    do
      drive forwards speed % 60 distance cm 25
      turn right speed % 50 degree 90
      play whole note g
  play whole note g

```



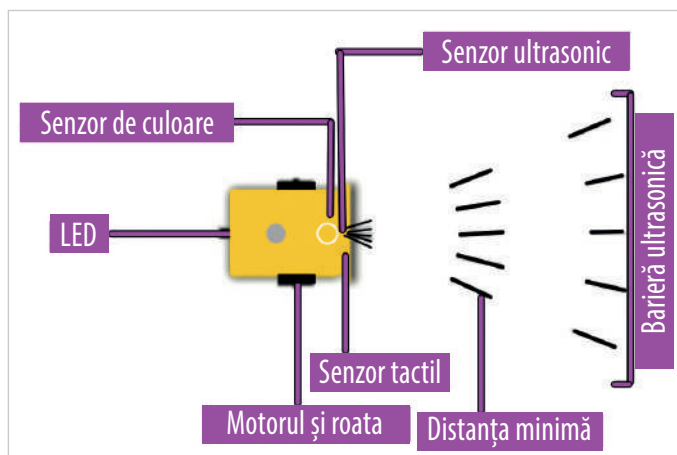


În exemplu s-a utilizat blocul **play** din grupul **Action** care a redat notele muzicale **Do**, respectiv **Sol**.

Deoarece la trasarea unui pătrat se repetă deplasarea și rotirea spre dreapta cu 90°, s-a utilizat un bloc **repeat** din grupul **Control** – o structură repetitivă cu contor (se repetă de 4 ori).

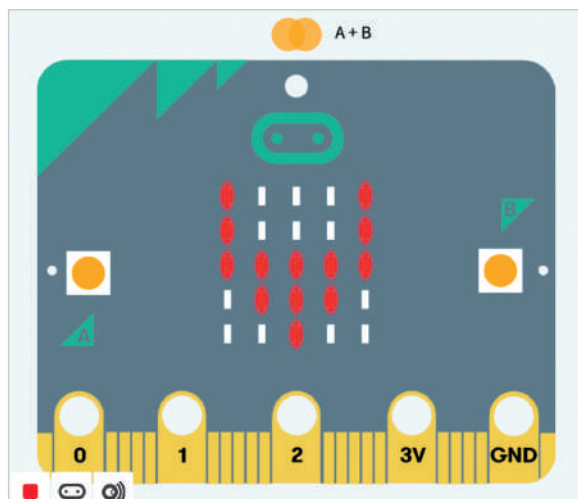
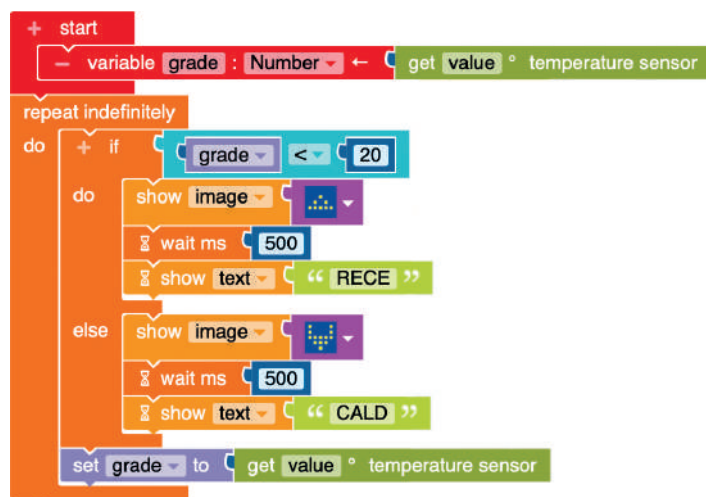
Pătratul albastru din imagine reprezintă un posibil obstacol pentru robot. Dacă robotul s-ar apropia de obstacol trasarea nu ar putea avea loc, fără comenzi suplimentare.

Există mai multe tipuri de **senzori**, în funcție de tipul de robot: **de culoare**, **tactil/de atingere**, **ultrasonic**, **de rotație**, **giroscopic**, **infraroșu**, **de lumină** ș.a. (vezi imaginea alăturată)



Exemplu: Utilizarea senzorului de temperatură

Preluarea temperaturii din încăperea cu ajutorul unui robot **micro:bit** și afișarea unui mesaj corespunzător, în funcție de valoarea acesteia:

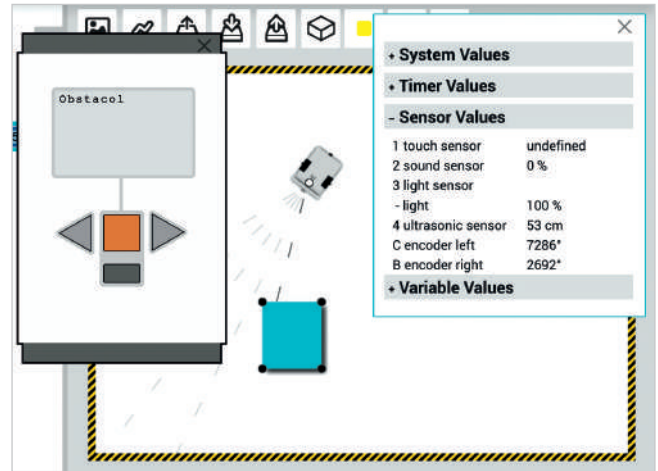
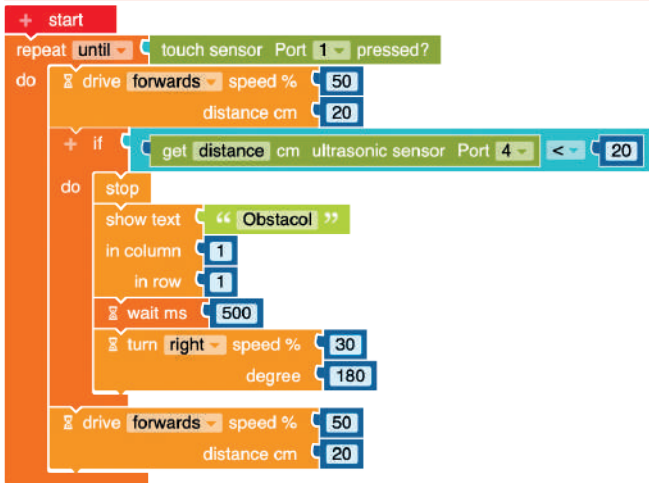


În exemplu s-a utilizat o variabilă **grade**, căreia i s-a atribuit temperatura, obținută cu ajutorul blocului **get value temperature sensor** din grupul **Sensors**.

S-a utilizat un bloc de repetiție, pentru a putea prelua temperatura în mod continuu, bloc din grupul **Control**.

În interiorul structurii repetitive se verifică în mod constant dacă temperatura este mai mică de 20, iar atunci pe placă se va afișa o imagine reprezentativă și textul **RECE**, iar în caz contrar o altă imagine și textul **CALD**.

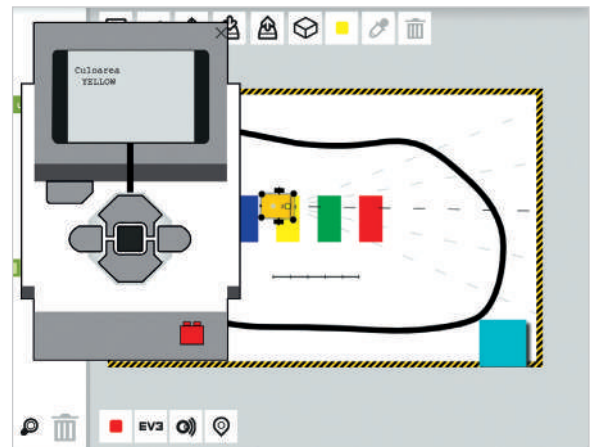
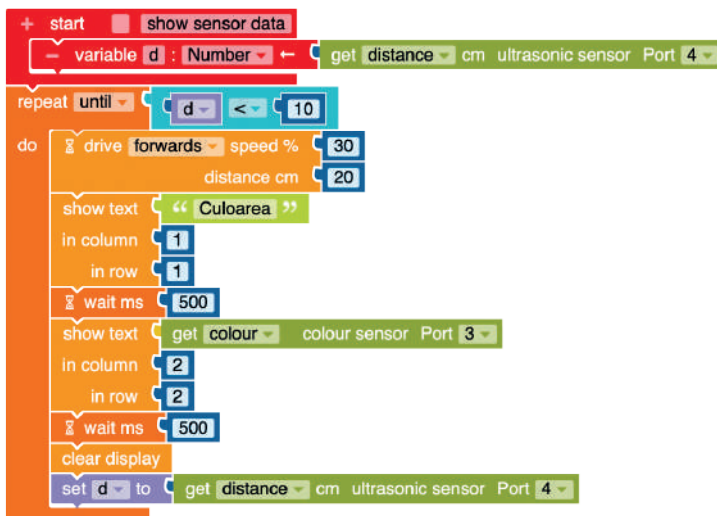
EX. Exemplu: **Evitarea obstacolelor** cu ajutorul unui robot *NXT*, prin efectuarea următoarelor acțiuni: oprirea motorului, întoarcerea cu 180° și continuarea deplasării în scenă.



■ În exemplu s-a utilizat un bloc de repetiție, pentru deplasarea robotului în mod continuu, bloc din grupul **Control**, repetiție care se oprește atunci când senzorul tactil este apăsat.

■ În interiorul structurii repetitive se verifică, cu ajutorul **senzorului ultrasonic**, dacă există vreun obstacol la o distanță mai mică de 20 cm. În caz afirmativ se afișează mesajul **Obstacol**, se oprește motorul, robotul se întoarce la dreapta cu 180° și își continuă deplasarea înainte.

EX. Exemplu: **Detectare culoare** cu ajutorul senzorilor de culoare a unui robot *EV3*, prin efectuarea următoarelor acțiuni: preluarea culorii în timpul deplasării, afișarea pe ecran a acesteia și continuarea deplasării până la apropierea de un obstacol la o distanță mai mică de 10 cm.



■ În exemplu s-a utilizat un bloc de repetiție, pentru deplasarea robotului

în mod continuu, bloc din grupul **Control**, repetiție care se oprește atunci când distanța față de un obstacol este mai mică de 10 cm. În interiorul structurii repetitive, se preia culoarea cu ajutorul **senzorului de culoare**, se afișează culoarea detectată pe ecran, iar apoi se atribuie în variabila **d**, cu ajutorul **senzorului ultrasonic**, noua distanță față de obstacole.



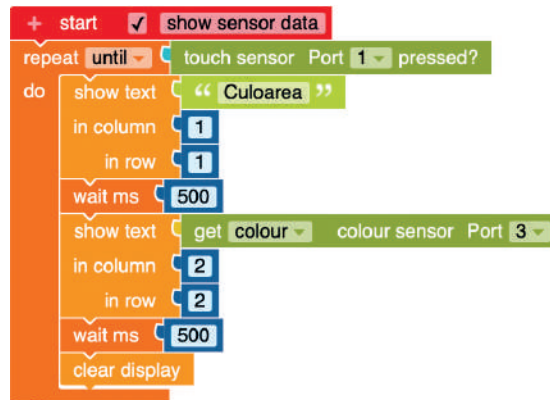
Rețineți! *Open Roberta Lab* este o platformă gratuită care permite programarea roboților utilizând limbajul de programare **NEPO** (New Easy Programming Online).



Fișa de lucru 27

Creați dosarul numit **Fisa27**, realizați programul următor utilizând platforma Open Roberta Lab și descărcați în dosar proiectul, cu numele **programare_robot.xml**:

Pornind de la algoritmul alăturat, care afișează culoarea recunoscută de senzorul de culoare, realizați modificările necesare pentru ca robotul **EV3** să se întoarcă cu 120° la întâlnirea unui obstacol la o distanță mai mică de 45 cm.



Aplicație suplimentară

Împărțiți elevii în grupe de 3-4, alegeți un robot de pe platformă și notele muzicale a 3-4 piese. Fiecare elev își alege o combinație de note muzicale și o redă cu ajutorul robotului ales. La final, fiecare grupă prezintă piesa aleasă.

În imaginea alăturată se poate vedea un model pentru **robotul Calliope mini**:



Autoevaluare

Alegeți pentru fiecare element din tabel răspunsul potrivit și completați enunțul final. Scrieți răspunsurile într-un document numit **autoevaluare.docx**, folosind notații precum 1 – da, 1 – nu, 1 – nu sunt sigur(ă), în funcție de răspunsurile alese. Salvați fișierul în dosarul **Fisa27**.

	Da	Nu	Nu sunt sigur(ă)
La finalul lecției, știu...			
1. ce reprezintă robotica.			
2. ce tipuri de roboți programabili există.			
3. cum să utilizez platforma Open Roberta Lab.			
4. cum să elaborez codul sursă pentru controlul robotului didactic virtual prin utilizarea și interpretarea datelor primite de senzorii acestuia.			

5. Am întâmpinat dificultăți în: ...

Nu uitați să vă evaluați comportamentul și activitatea din timpul lecției!
(cu ajutorul fișei de observare a comportamentului și a activității, de la pagina 99)

La finalul lecției, m-am simțit:



Evaluare sumativă



Verifică-ți cunoștințele!

(Punctaj maxim: 70 puncte)



Cerință: Plecând de la următoarea secvență de algoritm, pentru citirea unui șir de valori cu un număr necunoscut de elemente, până la întâlnirea valorii 0, rezolvați cerințele următoare:

1. Declarați variabilele utilizate și modificați secvența de citire, astfel încât șirul să preia doar valorile care au ultima cifră pară.
2. Parcurgeți noul șir de valori și afișați, separate printr-un spațiu, elementele de pe poziții impare, începând cu poziția 1.
3. Afișați, pe o linie nouă, numărul total de divizori ai numerelor din șir și verificați dacă șirul conține numere perfecte, afișând un mesaj corespunzător. Un număr se consideră **perfect** dacă acesta este egal cu suma divizorilor săi mai puțin el însuși (Ex.: $6 = 1 + 2 + 3$).

4. Salvați fișierul cu numele **evaluare.cpp**.

Timp de lucru: 50 de minute.

Mult succes!

Barem de corectare posibil:

1. **5p – 3p** – declarare variabile, **2p** – condiție corectă pentru reținerea valorilor în șir;
2. **15p – 3p** – parcurgere corectă, **10p** – condiția corectă pentru poziție, **2p** – afișarea conform cerințelor;
3. **50p – 5p** – trecerea la linie nouă, **15p** – determinarea numărului de divizori pentru fiecare element, **10p** – afișarea corectă a numărului total de divizori, **15p** – verificarea unui număr dacă este perfect, **5p** – verificarea existenței numerelor perfecte și afișarea unui mesaj corespunzător.

```
i = 0;
do
{
    cout<<"Dați un element";
    cin>>x;
    v[i] = x;
    i++;
} while (x != 0);
n = i - 1;
```



Investighează!

(Punctaj maxim: 10 puncte)

În multe situații elementele unui șir de valori trebuie sortate crescător sau descrescător. Pentru a descoperi modalitățile de sortare a unui șir, lucrați în echipe de 2-3 elevi și parcurgeți următorii pași:

1. Alegeți o metodă de sortare pentru care determinați principiul de funcționare, avantajele și dezavantajele.
2. Scrieți un program în limbajul C++, în care se utilizează metoda aleasă, folosind comentarii utile pe parcurs.
3. Realizați un document **investighează.docx**, în dosarul **Capitol3**, în care prezentați metoda de sortare aleasă conform cerințelor de la pct.1, inserați programul realizat în C++ și concluziile voastre.

Modalitate de evaluare:

- Documentul conține elementele indicate: principiu de funcționare – **2p**, avantaje – **1p**, dezavantaje – **1p**, programul C++ corespunzător pct.2 – **3p**, concluzii – **1p**
- Originalitate/creativitate – **2p**



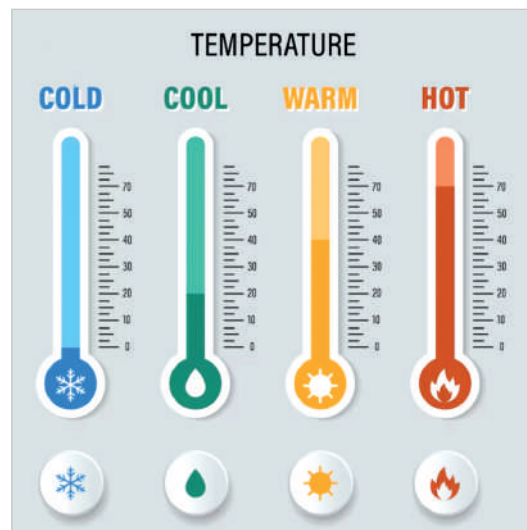
Proiect

(Punctaj maxim: 10 puncte)

Analiza și prelucrarea datelor meteorologice

Scrieți un program C++ cu ajutorul căruia să analizați temperaturile înregistrate la amiază într-o anumită perioadă (de exemplu, temperaturile zilnice dintr-o săptămână) și parcurgeți următorii pași:

1. Rețineți valorile înregistrate, cu ajutorului unui șir de valori;
2. Determinați temperatura medie, temperatura maximă/minimă, numărul de zile în care temperaturile îndeplinesc anumite condiții (de ex. temperaturi peste/sub o anumită valoare);
3. Salvați programul cu numele **temperatura.cpp** în dosarul **Capitol3**.



(Evaluare proiect: 1p/reținere date, 7p/elemente de conținut, 2p/creativitate)



Portofoliu

(Punctaj maxim: 10 puncte)

La finalul acestui capitol, portofoliul personal al elevului ar trebui să cuprindă în dosarul **Capitol3** următoarele materiale:

- 📁 Dosarul **Fisa22**, cu fișierele: **autoevaluare.docx**, **raspunsuri.docx**;
- 📁 Dosarul **Fisa23**, cu fișierele: **autoevaluare.docx**, **raspunsuri.docx**;
- 📁 Dosarul **Fisa24**, cu fișierele: **aplicatie1.cpp**, **aplicatie2.cpp**, **aplicatie3.cpp**, **aplicatie4.cpp**, **autoevaluare.docx**;
- 📁 Dosarul **Fisa25**, cu fișierele: **aplicatie1.cpp**, **aplicatie2.cpp**, **aplicatie3.cpp**, **aplicatie4.cpp**, **autoevaluare.docx**;
- 📁 Dosarul **Fisa26**, cu fișierele: **aplicatie1.cpp**, **aplicatie2.cpp**, **aplicatie3.cpp**, **autoevaluare.docx**;
- 📁 Dosarul **Fisa27**, cu fișierele: **autoevaluare.docx**, **programare_robot.xml**;
- 📁 Fișierele: **evaluare.cpp**, **investigheaza.docx**, **temperatura.cpp**.

Modalitate de evaluare:

Există dosarul **Capitol3** cu structura de directoare specificată – 2p.

Există în dosare: ♦ peste 95% din fișiere – 8p ♦ între 75% și 94% – 6p ♦ între 50% și 74% – 4p ♦ între 25% și 49% – 2p ♦ sub 25% – 0p.

Recomandare! Nota finală a elevului se va stabili cumulând punctajele obținute la **Verifică-ți cunoștințele!**, **Investighează**, **Proiect** și **Portofoliu**.



Aplicații suplimentare

Scrieți în limbajul C++, un algoritm de rezolvare pentru următoarele:

Citirea unui șir de valori. Afișarea unor elemente ale șirului

1. (*) Se consideră un șir cu n valori, numere întregi. Afișați elementele impare de pe pozițiile divizibile cu 3 (Ex.: $n=7$; 3 4 8 1 -2 5 13 – se va afișa 3 1 13 – elementele de pe pozițiile 0, 3 și 6)

2. (*) Se citește un șir de numere naturale, până la întâlnirea valorii 0. Afișați numerele de pe pozițiile impare ale șirului. (Ex.: 2, 3, 5, 1, 25, 0 – se va afișa: 3 1)

3. (**) Se consideră un șir de valori reale până la întâlnirea valorii 0. Determinați acele valori care sunt numere întregi. (Ex.: 3.44 3.76 7 4 8.345 0 – se va afișa 7 4)

4. (**) Se consideră un șir de x valori numere reale, reprezentând prețurile unor produse dintr-un magazin comercial. Parcurgeți șirul, reduceți prețurile cu $k\%$, k fiind un număr cunoscut ($1 \leq k \leq 100$) și afișați noile prețuri. (Ex.: $x = 4$; 12.50, 50, 45.50, 125 $k = 10$ – se va afișa 11.25 45 40.95 112.5)



5. (***) La un supermarket cartofii dulci sunt la ofertă promoțională. Cunoscând faptul că în stoc sunt c kg de cartofi și câte kilograme cumpără fiecare client, determinați numărul de clienți, și afișați numărul de kilograme cumpărate de fiecare client în parte până la epuizarea stocului de cartofi. (Ex.: $c=35$; 5, 3, 10, 2, 5, 5, 2, 3 – se va afișa 8 și apoi șirul 5 3 10 2 5 5 2 3)

Parcurerea unui șir de valori

6. (**) Se citește un șir cu n valori, numere naturale. Să se afișeze suma cifrelor fiecărui număr din șir. (Ex.: $n=5$; 123, 454, 121, 1000, 25 – se va afișa 6 13 4 1 7)

7. (***) Claudia este pasionată de flori și a amenajat un loc special pentru ele. Pe raft sunt aranjate n ghivece de flori. În funcție de pozițiile pe care le ocupă pe raft, pe pozițiile pare sunt ghivece cu orhidee, iar pe cele impare ghivece cu tradescantia. Pentru fiecare ghiveci ea notează numărul de flori ale plantei respective. Parcurgeți valorile reținute și afișați în ordinea ghivecelor de pe raft numărul de flori al fiecărei orhidee și apoi pe următorul rând afișați în ordinea inversă a ghivecelor numărul de flori al fiecărei tradescantia. (Ex.: $n=5$; 13, 3, 4, 7, 8 – se va afișa pe prima linie 13 4 8, iar pe a doua linie 7 3)



8. (****) Se dă un șir cu n valori, numere naturale. Afișați elementele după următoarea regulă: primul element, ultimul, al doilea element, penultimul etc. (Ex.: $n=7$; 1 4 2 7 5 3 9 – se va afișa 1 9 4 3 2 5 7)

Indicație: se utilizează două variabile pentru parcurgerea indicilor, una de la începutul șirului și una de la sfârșit.

Generarea unui șir de valori

9. (***) Construiți un șir cu n valori, în care, pe pozițiile pare valoarea este egală cu dublul indicelui (poziției), iar pe poziții impare valoarea este egală cu 2^i , unde i reprezintă indicele/ poziția din șir. (Ex.: $n=5$ – se va afișa 0 2 4 8 8, deoarece $0*2=0$, $21=2$, $2*2=4$, $23=8$, $4*2=8$)

10. (***) Construiți un șir de valori cu primele n numere care au ultimele două cifre egale, pornind de la un număr x ($x>10$) cunoscut. (Ex.: $n=4$ $x=1477$ – se va afișa 1477 1488 1499 1500)

11. (***) Determinați primele n numere pare, nenule, divizibile cu x , unde x este o valoare citită de la tastatură. (Ex.: $n=4$ $x=5$ – se va afișa 10 20 30 40)

12. (***) Considerăm următorul șir, în care n este un număr natural nenul:

$$f_n = \begin{cases} 1, & \text{dacă } n = 1 \\ 3, & \text{dacă } n = 2 \\ 2 * f_{n-1} + f_{n-2} - 1, & \text{dacă } n > 2 \end{cases}$$

Generați și afișați, separate printr-un spațiu, primele n valori ale acestui șir. (Ex.: $n=5$ – se va afișa 1 3 6 14 33)

13. (***) Considerăm următorul șir, în care n este un număr natural nenul:

$$f_n = \begin{cases} 2, & \text{dacă } n = 1 \\ (-1)^{1+n} * f_{n-1} - 1, & \text{dacă } n = \text{par} \\ (-1)^{2+n} * f_{n-1} + 3, & \text{dacă } n = \text{impar} \end{cases}$$

Generați și afișați, separate printr-un spațiu, primele n valori ale acestui șir. (Ex.: $n=5$ – se va afișa 2 -3 6 -7 10)

14. (****) Generați și afișați primii n termeni ai șirului Fibonacci, știind că primii doi termeni sunt 1, iar fiecare termen care urmează este egal cu suma dintre numărul precedent și cel anteprecedent al său. (Ex.: $n=6$ – se va afișa 1 1 2 3 5 8)

15. (****) A fost odată ca niciodată, un balaur **Caporyi** care ataca în fiecare zi castelul din regat. El avea 5 capete și la fiecare atac soldații reușeau să îi taie unul, dar peste noapte îi creșteau înapoi altele 5. Stabiliți câte capete va avea balaurul după n zile de atac asupra castelului și creați un șir cu numărul de capete pe care îl are în fiecare zi și noapte. (Ex.: $n=3$ – se va afișa 12 și șirul 5 4 9 8 13 12, unde valorile subliniate reprezintă numărul de capete pe care le are în fiecare zi, iar celelalte sunt după fiecare noapte)



16. (**)** Generați un șir cu un număr impar n de elemente care va conține pe pozițiile pare o valoare de trei ori mai mare decât poziția, iar pe pozițiile impare o valoare egală cu suma valorilor vecine. (Ex.: $n=5$ se va afișa 0 6 6 18 12, unde valorile subliniate sunt sume ale elementelor vecine)

Sumă/Produs/Medie

17. (*) Se consideră un șir cu n valori, numere naturale. Determinați suma ultimelor cifre ale celor n numere. (Ex.: $n=4$; 1234, 210, 2224, 1000 – se va afișa 8)

18. ()** Se consideră un șir cu n valori, numere întregi. Determinați valoarea absolută a diferenței dintre suma elementelor pare și suma elementelor impare. (Ex.: $n=7$; 3 4 8 1 -2 5 3 – se va afișa 2, deoarece $S_{\text{pare}} = 10$, iar $S_{\text{impare}} = 12$)

19. ()** Cunoscând temperatura medie a fiecărei luni dintr-un an calendaristic, determinați media anuală a temperaturilor pozitive, respectiv a celor negative și afișați cele două medii pe două linii diferite. (Ex.: -11, -3, 8, 5, 17, 24, 26, 32, 23, 18, 9, -7 – se va afișa pe prima linie 18, iar pe a doua linie -7)

20. (*)** Se dă un șir cu n valori, numere naturale. Afișați produsul elementelor prime din șir. (Ex.: $n=7$; 4, 5, 3, 12, 18, 2, 99 – se va afișa 30)

Minim/Maxim

21. ()** La un concurs au participat x concurenți. În urma primei etape fiecare concurent a obținut un punctaj. Determinați și afișați punctajul cel mai mare, precum și punctajul cel mai mic. (Ex.: $x=5$; 18, 25, 11, 20, 8 – se afișează 25 și 8)

22. (*)** Cu ocazia reducerilor de vară, magazinele s-au decis să aplice reduceri la x produse. Cunoscând prețurile inițiale și prețurile după reducere, determinați și afișați care a fost cea mai mare reducere aplicată, în procente.

Ex.: $x=4$; 15 25 80 100 – preț inițial

12 22.5 48 50 – preț cu reducere

Se va afișa 50%, deoarece reducerile aplicate sunt 20%, 10%, 40%, 50%.

Indicație: se vor utiliza 3 șiruri de valori, câte un șir pentru prețurile inițiale, prețurile cu reducere și unul pentru reduceri.



Numărare

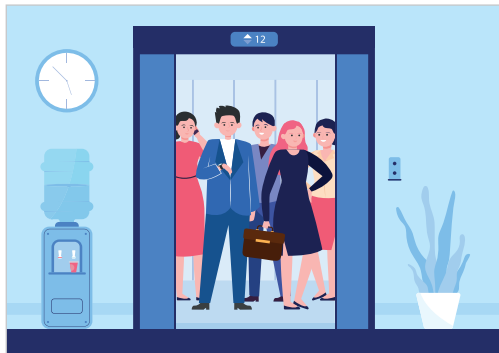
23. ()** Pe o fișă sunt notate n numere întregi nenule (n – impar). Raluca trebuie să determine semnul majoritar al acestora, folosind un mesaj corespunzător. (Ex.: $n=5$; 71, -43, -23, 25, -11 – se va afișa "Majoritatea negativă")

24. (***) Urcând de n ori cu un lift, la diferite etaje ale unei clădiri, stabiliți câte dintre etajele la care s-a urcat au proprietatea că au exact doi divizori proprii. (Ex.: $n=5$; 5, 6, 12, 8, 20 – se va afișa 2, deoarece 6 și 8 au exact 2 divizori proprii)

25. (***) Se consideră un șir cu n valori, numere naturale. Afișați câte dintre valorile din șir sunt pătrate perfecte. (Ex.: $n=5$; 10, 9, 16, 23, 7 – se va afișa 2, deoarece 9 și 16 sunt pătrate perfecte)

26. (***) Se consideră un șir cu n valori, numere naturale. Afișați câte dintre valorile din șir sunt prime. (Ex.: $n=5$; 110, 11, 18, 230, 2 – se va afișa 2, deoarece 11 și 2 sunt numere prime)

27. (***) Se citesc numere până se introduce același număr de 2 ori, consecutiv. Numerele sunt cuprinse în intervalul 1 – 100. Afișați câte numere din șir apar o singură dată. (Ex.: 3, 5, 7, 3, 6, 7, 1, 1 – se va afișa 2 deoarece 5 și 6 apar o singură dată în șir)



Șiruri de frecvență (facultativ)



28. (***) Navele spațiale *BlueInternship* și *Discovery* au inscripționate pe ele niște coduri formate din cifre de la 0 la 9. Determinați și afișați, în ordine descrescătoare, care cifre nu au fost utilizate la inscripționarea codurilor. (Ex.: 456437, 2234561 – se va afișa 9 8 0)

29. (***) Se introduc numere naturale, cuprinse între 1 și 1000, până la întâlnirea valorii 0. Afișați numerele de forma $\overline{xx}$ care nu apar în șirul dat, respectiv

un mesaj potrivit, în cazul în care există toate. (Ex.: 456, 55, 11, 89, 44, 234, 0 – se va afișa 22 33 66 77 88 99)

30. (***) Se dă un șir de n valori numere naturale. Determinați și afișați cifra care apare de cele mai multe ori în numerele din șir. În cazul în care există mai multe cifre cu același număr de apariții să se afișeze cifra mai mică. (Ex.: $n=4$; 22845, 455, 88, 109, 67 – se va afișa 5, numărul de apariții maxim fiind 3 pentru cifrele 5 și 8)

Verificarea unei proprietăți îndeplinită de elementele unui șir

31. (**) Având un șir cu n valori, numere naturale, determinați și afișați, separate printr-un spațiu, elementele din șir care sunt divizori pentru primul element al șirului, exceptând primul. (Ex.: $n=4$; 125, 43, 5, 25 – se va afișa 5 25)

32. (***) Se consideră un șir cu n valori, numere întregi. Afișați, pe linii diferite:

- a. Numerele pare din șir, separate prin spații.
- b. Numerele negative din șir, separate prin spații.
- c. Numerele de 2 cifre din șir (negative sau pozitive).
- d. Numerele multiplu de k din șir, unde k se citește.
- e. Numerele naturale, prime din șir.

(Ex.: $n=5$; 12 -3 7 8 9 $k=3$ – se va afișa a. 12 8 b. -3 c. 12 d. 12 -3 9 e. 7)

33. (***) Se dă un șir cu n valori, numere naturale, de 5 cifre. Determinați și afișați numerele care au suma dintre prima și ultima cifră un număr par și produsul dintre a treia și a patra cifră un multiplu de 3. (Ex.: $n=5$; 12345, 23453, 23451, 87543, 45678 – se va afișa 12345 45678)

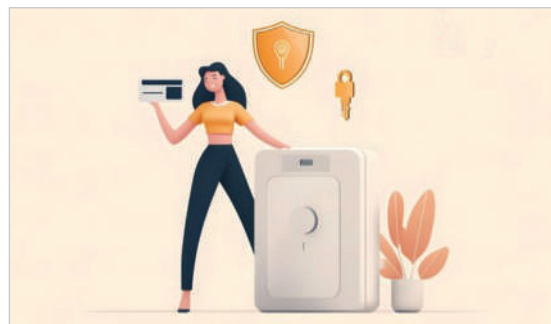
Verificarea existenței unei valori într-un șir

34. (****) Se dă un șir cu p valori, numere întregi. Verificați dacă șirul conține numere pseudo-perfecte și afișați un mesaj adecvat (Ex.: da, există; nu, nu există). Un număr este **pseudoperfect** dacă suma divizorilor săi reprezintă un multiplu al numărului. (Ex.: $n=4$; 4, 120, 17, 25 – se va afișa DA, deoarece 120 este pseudoperfect – având suma divizorilor 360, care este un multiplu al lui 120)



35. (****) La un concurs național de astrofizică participă elevi din diferite orașe. Lucrările elevilor sunt codificate sub forma $\overline{xyzw}$, unde x reprezintă o cifră asociată orașului din care face parte elevul, y reprezintă o cifră asociată școlii de proveniență a elevului, z punctajul lucrării (0...9) și w o cifră ce indică locul pe care se află, iar în cazul în care nu are premiu se va trece 0. Având codurile a n elevi, verificați și specificați printr-un mesaj dacă există copii cu premiul 1 din orașul o , fiind un număr cunoscut. (Ex.: $n=4$ $o=9$; 2432, 9562, 9793, 5450 – se afișează "NU", deoarece nu există elevi din orașul 9 cu premiul 1 doar cu 2 și 3)

36. (****) Pentru a deschide un seif, Maria are nevoie de un număr **palindrom**. Un număr este palindrom dacă acesta este egal cu oglinditul său (Ex.: 1223221). Maria are la dispoziție un șir cu n numere. Ajutați-o pe Maria să găsească un astfel de număr și stabiliți dacă va reuși sau nu să deschidă seiful. (Ex.: $n=4$; 1345, 221122, 2553, 6554 – se va afișa "DA", deoarece 221122 este un număr palindrom)



Verificarea unei proprietăți comune pentru toate elementele unui șir

37. (**) Se consideră un șir cu n valori, numere naturale de 2 cifre. Stabiliți dacă toate numerele din șir au cifre diferite. (Ex.: $n=3$; 12, 34, 88 – se va afișa NU, deoarece 88 nu are cifre diferite)

38. (***) Un voucher de la cinematograful se consideră a fi **voucher cadou** dacă toate cifrele scrise pe el sunt egale. Cunoșcând numărul unui voucher cu c cifre, verificați dacă acesta este voucher cadou sau nu. (Ex.: $c=5$; 7, 7, 7, 7, 7 – se va afișa mesajul "Voucher cadou")

39. (***) Se dă un șir de n valori numere întregi nenule. Verificați dacă toate numerele din șir sunt numere negative și afișați un mesaj corespunzător. (Ex.: $n=5$; -5, -7, 5, -4, -3 – se va afișa NU)

40. (***) Se consideră un șir cu n valori, numere întregi. Stabiliți dacă toți termenii șirului, exceptând primul și ultimul, se pot obține ca sumă a valorilor vecine. (Ex.1: $n=5$; 2, 5, 3, -2, -5 – se va afișa Da; $n=5$; 1, 3, 2, -1, 1 – se va afișa Nu)

41. (****) În urma numerotării caselor pe o stradă, se verifică dacă numerotarea a fost realizată corect. Astfel că, în șirul de valori dat fiecare număr trebuie să aibă aceeași paritate cu vecinii acestuia. Fiind dat un șir cu n numere (ale caselor), verificați și afișați un mesaj corespunzător dacă numerotarea caselor a fost făcută corect sau nu. (Ex.: $n=7$; 21, 23, 25, 27, 29, 31, 33 – se va afișa "Numerotare corectă")

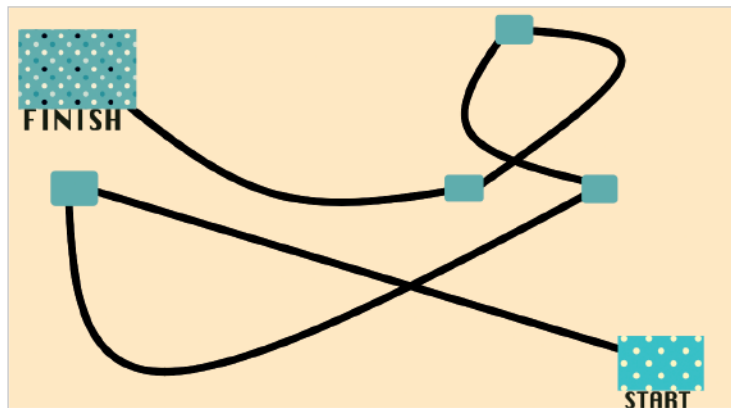


Utilizați limbajul NEPO pentru următoarele aplicații:

Robot didactic virtual

42. (**) Realizați un program care deplasează un robot (la alegere) până la o linie albastră, utilizând senzorul de culoare, iar în cazul întâlnirii unui obstacol, robotul să emită un sunet ales de voi.

43. (***) Realizați un program care deplasează un robot (la alegere) pe un traseu dat de o linie neagră utilizând senzorul de culoare. În cazul în care senzorul cu ultrasunete este la o distanță mai mică de 50 cm de un obstacol robotul trebuie să se oprească.



44. (****) Realizați un concurs de robotică, pe echipe, la nivel de clasă. Fiecare echipă programează un robot la alegere, care se va deplasa pe o traiectorie dată (vezi imaginea) și va depăși obstacolele întâlnite pe traseu, cu ajutorul senzorilor tactili și cel cu ultrasunete. Câștigă echipa care reușește să parcurgă traseul într-un timp cât mai scurt.

Recapitulare finală

Exercițiul 1. Compania **EcoGrow** prezintă câteva date informative despre vânzările de răsaduri în patru regiuni din țară, specificând numărul de răsaduri vândute din fiecare tip: tomate, ardei, castraveți. Creați un registru de calcul numit **EcoGrow.xlsx** și realizați următoarele:

a. un tabel precum cel din imaginea de mai jos. Notați în coloanele **B**, **C** și **D** numărul de răsaduri vândute din fiecare tip. Formatați textul astfel: font **Arial**, dimensiunea **14 pt** pentru primul rând și **11 pt** în rest, linii de bordură punctate între celule și duble pe exterior, având 2 culori la alegere;

b. în coloanele **E** și **F**, determinați numărul total de răsaduri vândute și media vânzărilor pentru fiecare regiune;

	A	B	C	D	E	F	G
1	Denumiri Regiuni	Nr. răsaduri tomate	Nr. răsaduri ardei	Nr. răsaduri castraveți	Nr. total răsaduri vândute	Media vânzări pentru fiecare regiune	Vânzarea depășește 500 de bucăți
2	Regiunea 1 - Muntenia	115	20	300	435	145	NU
3	Regiunea 2 - Dobrogea	205	35	100	340	113,3333333	NU
4	Regiunea 3 - Banat	100	45	95	240	80	NU
5	Regiunea 4 - Bucovina	55	150	320	525	175	DA

c. în coloana **G** verificați dacă numărul total de răsaduri vândute în fiecare regiune depășește 500 de bucăți, afișând un mesaj corespunzător (DA/NU). (30p)

Exercițiul 2. Realizați o pagină web cu numele **EcoGrow.html**, pentru prezentarea companiei **EcoGrow**, care va conține următoarele elemente:

- un titlu – denumirea companiei, centrat, formatat la alegere.
- un tabel cu valorile determinate anterior (numărul total de răsaduri și media vânzărilor).
- o listă ordonată cu regiunile care au numărul total de răsaduri mai mare de 500 (considerăm că există cel puțin o astfel de regiune).
- o imagine reprezentativă, aliniată la stânga.
- fundal al paginii particularizat.



(30p)

Exercițiul 3. Scrieți în limbajul C++ un algoritm de rezolvare a problemei: Se introduc într-un șir de valori **numărul total de răsaduri** vândute pentru fiecare regiune, determinate la primul exercițiu. Verificați dacă există un număr prim de răsaduri în acest șir și afișați un mesaj corespunzător (DA/NU). Salvați fișierul cu numele **prim.cpp**. (Ex. $n=4$; 435, 340, 240, 525 – se va afișa mesajul NU) (30p)

Oficiu: 10 puncte

Total: 100 puncte

Timp de lucru: 50 de minute.

Mult succes!

Barem de corectare:

- 30p – 5p – nume fișier, 10p – creare tabel conform cerințelor, 5p – determinare total, 5p – calcul medie, 5p – verificarea condiției cerute;
- 30p – 5p – nume fișier, 5p – titlu pagină, 5p – tabel, 5p – listă, 5p – imagine, 5p – fundal pagină;
- 30p – 5p – nume fișier, 10p – creare șir de valori, 10p – verificare număr prim, 5p – verificare existență și afișare mesaj corespunzător.

Evaluare finală

Exercițiul 1. Firma **I&T Ville** prezintă terenuri din 3 orașe ale județului, din zonele **Premium**, **Riviera** și **Residence**, specificând aria acestora. Creați un registru de calcul numit **ITVille.xlsx** și realizați următoarele:

- un tabel precum cel din imagine. Notați în coloanele **B**, **C** și **D** ariile terenurilor. Formatați textul astfel: font **Calibri**, dimensiunea **16 pt** pentru primul rând și **12 pt** în rest, linii de bordură întrerupte între celule și îngroșate pe exterior, având 2 culori la alegere;
- în coloanele **E** și **F** determinați aria maximă și media ariilor terenurilor din fiecare oraș;
- în coloana **G** verificați dacă suma ariilor din fiecare oraș depășește 300 și precizați un mesaj corespunzător (DA/NU).

	A	B	C	D	E	F	G	H
1	Orașe	I&T Ville Premium	I&T Ville Riviera	I&T Ville Residence	Aria Maximă	Media ariilor din fiecare oraș	Suma ariilor din fiecare oraș > 300 (DA/NU)	
2	Oraș 1							
3	Oraș 2							
4	Oraș 3							
5								

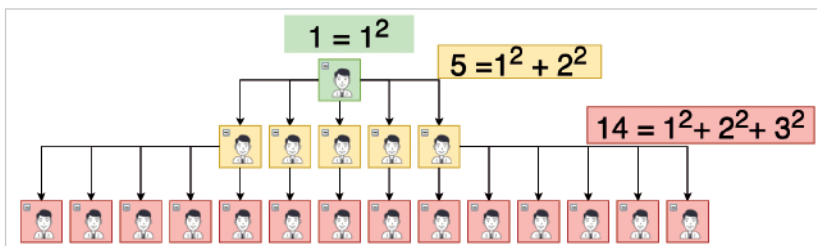
Exercițiul 2. Realizați o pagină web cu numele **ITVille.html**, pentru prezentarea firmei **I&T Ville**, care va conține următoarele elemente:

- un titlu – denumirea firmei, centrat, formatat la alegere;
- un tabel cu valorile determinate anterior (Aria maximă și media);
- o listă neordonată cu orașele care au suma ariilor mai mare de 300 (considerăm că există cel puțin un astfel de oraș);
- o imagine reprezentativă, aliniată la dreapta. Particularizați fundalul paginii.

Exercițiul 3. Scrieți în limbajul

C++ un algoritm de rezolvare a problemei:

Se introduc într-un șir de valori ariile maxime determinate la primul exercițiu, verificați dacă există un număr piramidal în acest șir și afișați un mesaj corespunzător (DA/NU). Un număr este **piramidal** dacă egal cu suma de pătrate perfecte consecutive, pornind de la 1. Salvați fișierul cu numele **piramidal.cpp**.



Un număr este **piramidal** dacă egal cu suma de pătrate perfecte consecutive, pornind de la 1. Salvați fișierul cu numele **piramidal.cpp**.

Oficiu: 10 puncte

Total: 100 puncte

Timp de lucru: 50 de minute.

Mult succes!

Barem de corectare:

- 30p** – 5p – nume fișier, 10p – creare tabel conform cerințelor, 5p – determinare maxim, 5p – calcul medie, 5p – verificarea condiției cerute;
- 30p** – 5p – nume fișier, 5p – titlu pagină, 10p – tabel, 5p – listă, 5p – imagine și fundal;
- 30p** – 5p – nume fișier, 10p – creare șir de valori, 10p – verificare număr piramidal, 5p – verificare existență și afișare mesaj corespunzător.

FIȘĂ DE OBSERVARE A COMPORTAMENTULUI ȘI ACTIVITĂȚII ELEVULUI

Nume și prenume elev: _____

Data: _____

Lecția: _____

Contextul în care se realizează observația: ora de *Informatică și TIC*

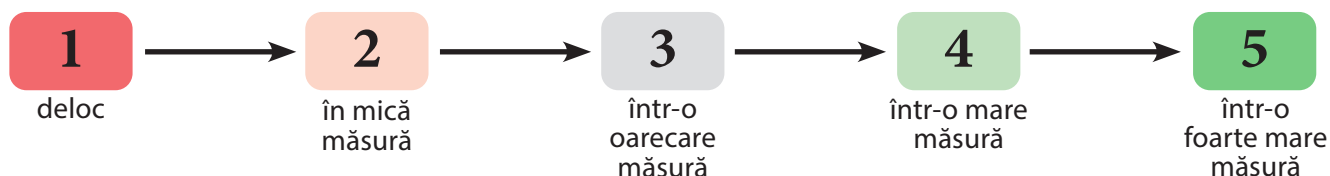
Scop: înregistrarea unor indici comportamentali și de activitate în vederea asigurării progresului școlar

Comportament/activitate urmărit(ă)	Autoevaluare elev 	Evaluare cadru didactic 
Am fost atent(ă) la lecție și la explicațiile profesorului.		
Am înțeles toate noțiunile prezentate.		
Am știut să răspund la toate întrebările adresate de profesor.		
M-am implicat activ în lecție/discuții, pe tot parcursul orei.		
Am solicitat ajutor și lămuriri suplimentare, când a fost nevoie.		
Am înțeles toate sarcinile de lucru.		
Am rezolvat toate sarcinile de lucru.		
Mi-a plăcut să lucrez în echipă, când a fost posibil.		
Am oferit ajutor colegilor mei, când mi s-a solicitat.		
Am colaborat cu colegii pentru a obține cele mai bune rezultate.		

Dragi elevi,
evaluați-vă la finalul fiecărei ore comportamentul/activitatea, completând în această fișă coloana *Autoevaluare elev*. În acest sens, acordați-vă un punctaj de la 1 la 5 (unde 1 înseamnă *deloc*, iar 5 într-o *foarte mare măsură*).

Coloana alăturată poate fi completată de către profesorul vostru.

Analizați împreună cu profesorul vostru punctajele acordate!



INDICAȚII ȘI RĂSPUNSURI

Capitolul 1 – Calcul tabelar:

Lecția 1 – Aplicația de calcul tabelar Microsoft Excel. Operații cu registrul de calcul:

Fișa de lucru 1 : 1. 4-Office, 2-calcul, 7-registru, 5-xlsx, 8-litere, 3-numere, 6-celulă, 1-adresă

Capitolul 2 – Pagini web:

Lecția 8 – Limbajul HTML. Editoare de pagini web

Fișa de lucru 21 : 1. 1-b, 2-c, 3-a. 2. https (hypertext transfer protocol secure), ssl (secure sockets layer = protocol de securitate). 3. Utilizarea unei parole puternice, actualizări periodice ale platformelor cu care sunt create site-urile, folosirea de certificate SSL, realizarea de back-up-uri etc.

Verifică-ți cunoștințele: 1. a) Etichete/tag-uri/marcaje b) html c) Google Sites/Notepad++ d) hyperlink.

2. a) F b) F c) A d) A e) F f) F.

Capitolul 3 – Algoritmi:

Lecția 16 – Limbajul de programare – noțiuni recapitulative

Fișa de lucru 22 : 1. b 2. a 3. c. 4. b 5. c 6. d 7. a

Aplicații suplimentare!

1.

```
#include<iostream>
using namespace std;
int main()
{
    unsigned int a, p, inva, invb;
    cout<<"Introdu numărul de stele de pe o parte a zmeului ";
    cin>>a;
    cout<<"Introdu numărul de stele de pe cealaltă parte a
zmeului ";
    cin>>b;
    if (a>9 && a<100)
        inva = (a%10)*10 + a/10;
    else
        if (a>1 && a<10)
            inva = a;
    if (b>9 && b<100)
        invb = (b%10)*10 + b/10;
    else
        if (b>1 && b<10)
            invb = b;
    if (a==inva && b==invb)
        cout<<"Robert iese la joacă cu zmeul";
    else
        cout<<"Robert dă zmeul surioarei mai mici Ana";
    return 0;
}
```

2.

```
#include<iostream>
using namespace std;
int main()
{
    unsigned int cod, min, max, nr_c, a, securizat;
    bool exista=0;
    do
    {
        cout<<"Introdu un cod ";
        cin>>cod;
        min=9;
        max=0;
        nr_c=0;
        a=cod;
        while(a!=0)
        {
            if(min>a%10)
                min=a%10;
            if(max<a%10)
                max=a%10;
            a=a/10;
            nr_c++;
        }
        if(min%2==0 && max%3==0 && nr_c%2==1)
        {
            exista=1;
            securizat=cod;
        }
    }while(cod!=0);
    if (exista)
        cout<<securizat;
    else
        cout<<"Nu exista";
    return 0;
}
```

Lecția 17 – Șir de valori – noțiuni introductive. Operații cu șiruri de valori

Fișa de lucru 23 : 1. d 2. c 3. a. 4. c

Fișa de lucru 24 :

1.

```
#include<iostream>
using namespace std;
int main()
{
    int cadre[25], i, n;
    cout<<"Specificati numarul de filmari ";
    cin>>n;
    for (i = 0; i < n; i++)
    {
        cout<<"cadre["<< i<<"]="";
        cin>> cadre[i];
    }
    for (i = n - 1; i >= 0; i--)
        cout<< cadre[i] <<" ";
    return 0;
}
```

2.

```
#include<iostream>
using namespace std;
int main()
{
    unsigned int flori[25], i, n;
    cout<<"Specificati numarul de flori ";
    cin>>n;
    for (i = 0; i < n; i++)
    {
        cout<<"flori["<< i<<"]="";
        cin>> flori[i];
    }
    for (i = 0; i < n; i+= 5)
        cout<< flori[i] <<" ";
    return 0;
}
```

3.

```
#include<iostream>
using namespace std;
int main()
{
    int f[25], i, n;
    cout<<"Specificati numarul de elemente ";
    cin>>n;
    f[0]=1;
    f[1]=5;
    for (i = 2; i < n; i++)
        f[i]=2*f[i-1] - f[i-2]+ 5;
    for (i = 0; i < n; i++)
        cout<< f[i] <<" ";
    return 0;
}
```

4.

```
#include<iostream>
using namespace std;
int main()
{
    int s[25], i, n, x;
    cout<<"Specificati numarul de elemente ";
    cin>>n;
    cout<<"Dati primul element ";
    cin>>x;
    s[0]=x;
    for (i = 1; i < n; i++)
        s[i]=4+s[i-1];
    for (i = 0; i < n; i++)
        cout<< s[i] <<" ";
    return 0;
}
```

Lecția 18 – Algoritmi de bază pentru șiruri de valori – numărare, sumă/produs și minim/maxim

Fișa de lucru 25 :

1.

```
#include<iostream>
using namespace std;
int main()
{
    int temp[25], i, t, s=0, nr=0;
    i = 0;
    do
    {
        cout<<"Introdu o temperatura ";
        cin>>t;
        temp[i]= t;
        i++;
    }while(t >= 0);
    n=i-1;
    for (i = 0; i < n; i++)
    {
        s=s + temp[i];
        nr++;
    }
    cout<<"Temperatura medie este "<<(float) s/nr;
    return 0;
}
```

3.

```
#include <iostream>
using namespace std;
int main()
{
    int piesa[35], i, n, nr = 0;
    cout<<"Dați numărul de piese vestimentare ";
    cin>>n;
    for (i = 0; i < n; i++)
    {
        cout<<"Introdu prețul unei piese ";
        cin>> piesa[i];
    }
    for (i = 0; i < n; i++)
        if (piesa[i] <= 15)
            nr ++;
    cout<<"Numărul de piese este "<< nr;
    return 0;
}
```

2.

```
#include<iostream>
using namespace std;
int main()
{
    unsigned int lalea[35], i, n, maxi, mini, pmax, pmin;
    cout<<"Specificati numarul de standuri ";
    cin>>n;
    for (i = 0; i < n; i++)
    {
        cout<<"lalea["<<i<<"]=" ";
        cin>> lalea[i];
    }
    maxi= lalea[0];
    mini= lalea[0];
    pmin= 0;
    pmax= 0;
    for (i = 1; i < n; i++)
    {
        if (lalea[i] > maxi)
        {
            maxi = lalea[i];
            pmax = i+1;
        }
        if (lalea[i] < mini)
        {
            mini = lalea[i];
            pmin = i+1;
        }
    }
    cout<<"Standul "<<pmax<<"-"<<maxi<<" lalele ";
    cout<<"Standul "<<pmin<<"-"<<mini<<" lalele ";
    return 0;
}
```

4.

```
min = x[1];
for (i = 2; i <= n; i++)
    if (x[i] < min)
        min = x[i];
cout<<"Numărul minim este " << min;
cout<<" și se află pe pozițiile ";
for (i = 1; i <= n; i++)
    if (x[i] == min)
        cout<< i<<" ";
```

Lecția 19 – Algoritmi de bază pentru șiruri de valori – verificare a unei proprietăți

Fișa de lucru 26 :

1.

```
#include <iostream>
using namespace std;
int main()
{
    float atom[100], a;
    int i, n;
    bool ok = false;
    cout<<"Introdu numărul de elemente ";
    cin>> n;
    for (i = 0; i < n; i++)
    {
        cout<<"atom["<<i<<"]=" ";
        cin>> atom[i];
    }
    for (i = 0; i < n; i++)
        if ((int) atom[i] == atom[i])
            ok = true;
    if (ok== true)
        cout<<"DA";
    else
        cout<<"NU";
    return 0;
}
```

3.

```
#include <iostream>
using namespace std;
int main()
{
    int p[100], i, n;
    bool ok = true;
    cout<<"Introdu numărul de bilanțuri ";
    cin>> n;
    for (i = 0; i < n; i++)
    {
        cout<<"Introdu bilanțul ";
        cin>> p[i];
    }
    for (i = 0; i < n; i++)
        if (p[i] < 0)
            ok = false;
    if (ok== true)
        cout<<"DA";
    else
        cout<<"NU";
    return 0;
}
```

2.

```
#include <iostream>
using namespace std;
int main()
{
    int a[100], b[100], i, n, d, Sa, Sb;
    bool ok = false;
    cout<<"Introdu numărul de elevi ";
    cin>> n;
    for (i = 0; i < n; i++)
    {
        cout<<"a["<<i<<"]=" ";
        cin>> a[i];
        cout<<"b["<<i<<"]=" ";
        cin>> b[i];
    }
    for (i = 0; i < n; i++)
    {
        Sa = 0;
        Sb = 0;
        for (d = 1; d <= a[i]/ 2; d++)
            if (a[i] % d == 0)
                Sa = Sa + d;
        for (d = 1; d <= b[i]/ 2; d++)
            if (b[i] % d == 0)
                Sb = Sb + d;
        if (Sa == b[i] && Sb == a[i])
            ok = true;
    }
    if (ok== true)
        cout<<"DA";
    else
        cout<<"NU";
    return 0;
}
```


Lecția 20 – Codebot – controlul unui robot didactic virtual

Fișa de lucru 27 :

