FIȘĂ DE LUCRU · PYTHON · CLASA A IX-A · FIȘA 1 · LECȚIA PY9-01

Gândire computațională și etapele unui program

Durată: 100 min Nivel: mate-info · Python

CE VEI ȘTI DUPĂ LECȚIE

- Recunoști etapa (analiză, proiectare, implementare, testare, depanare) în care se află o activitate și o greșeală.
- Treci același algoritm dintr-o reprezentare în alta: schemă logică, pseudocod, Python.
- Scrii și rulezi programe scurte, cu un tabel de teste completat înainte de a spune „gata".

Ce trebuie să știi (o pagină)

Noțiune	Pe scurt
gândire computațională	descompunere, recunoașterea tiparelor, abstractizare, algoritm
analiză	date de intrare, date de ieșire, condiții, cazuri speciale
proiectare	pașii algoritmului (schemă logică sau pseudocod)
implementare	scrierea programului în limbaj (Python)
testare	rulezi pe date alese: caz obișnuit, cazuri limită, date greșite
depanare	găsești cauza unei erori, o repari, apoi **testezi din nou**
schemă logică	se vede drumul execuției; greu de întreținut la algoritmi mari
pseudocod	aproape de limba română; formatul de la Bac; nu se rulează
limbaj de programare	se rulează și se testează; reguli stricte de scriere
compilator (C++)	traduce tot programul înainte de rulare
interpretor (Python)	execută instrucțiune cu instrucțiune; mod interactiv `>>>`
comunicare	consolă (`input` / `print`), interfață grafică (Tkinter), fișiere
operatori folosiți azi	`+ - * /` , `//` cât întreg, `%` rest, `**` putere

SFAT

Pentru fiecare program de la nivelul 3 completezi întâi analiza (intrare → ieșire) și tabelul de teste, **abia apoi** scrii codul.

Nivelul 1 · Prezice

★ citește și prezice

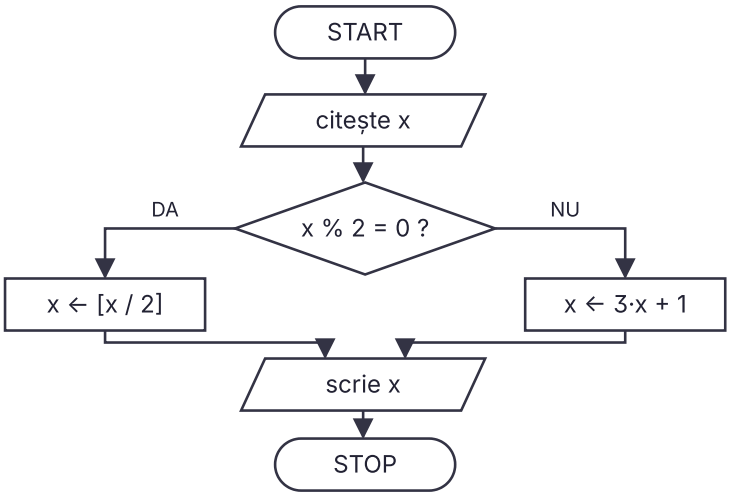
P1. Ce afișează fiecare rând în modul interactiv?

```
>>> 3 + 4 * 2
>>> (3 + 4) * 2
>>> 17 // 5
>>> 17 % 5
>>> 10 / 4
>>> 2 ** 10
```

P2. Ce afișează programul? Atenție la ce este între ghilimele.

```
print("Ana", "are", 3, "mere")
print(5 + 3, "5 + 3")
```

P3. Urmărește schema logică. Ce se afișează pentru `x = 6`, `x = 7` și `x = 1`?



P4. Ce afișează algoritmul pentru `n = 23`? Dar pentru `n = 4` și `n = 5`? Ce calculează, în cuvinte?

```
citește n
k ← 0
cât timp n ≥ 5 execută
    n ← n − 5
    k ← k + 1
scrie k, " ", n
```

P5. În ce etapă (analiză, proiectare, implementare, testare, depanare) se află fiecare activitate? a) Stabilești că rezultatul trebuie afișat cu două zecimale. b) Desenezi schema logică. c) Scrii codul în Thonny. d) Rulezi programul pe 5 date alese și compari cu ce aștepți. e) Programul afișează 0 în loc de 1 și cauți rândul vinovat. f) Te întrebi dacă `n` poate fi 0.

P6. Ce afișează fiecare program înainte să apară eroarea? Explică diferența folosind ce știi despre interpretor.

```
print("unu")            # programul A
print("doi")
prnt("trei")
```

```
print("unu")            # programul B
print("doi")
print("trei"
```

P7. Ce mod de comunicare (consolă, interfață grafică, fișiere) alegi și de ce? a) un convertor lei–euro pentru bunica; b) o problemă pbinfo; c) temperaturile măsurate din oră în oră, timp de un an, de o stație meteo; d) un program scurt pe care îl folosești doar tu, ca să-ți verifici media.

P8. Pentru o aplicație de **orar al clasei**, spune ce componentă a gândirii computaționale apare în fiecare propoziție: a) „Contează ziua, ora, materia și sala, nu și culoarea tablei." b) „Împărțim lucrul în: introducerea materiilor, așezarea pe zile, verificarea suprapunerilor, afișarea." c) „Verificarea suprapunerilor seamănă cu verificarea programărilor la medic." d) „Pentru fiecare oră din fiecare zi, compar sala cu sălile deja ocupate la acea oră."

Nivelul 2 · Găsește și repară

★★ găsește și repară

P9. *Greșeală de analiză.* Pe o latură de gard lungă de `L` metri (`L` par) se pune câte un stâlp la fiecare 2 m, **inclusiv la ambele capete**. Programul `print(L // 2)` dă un rezultat greșit pentru `L = 10`. Câți stâlpi sunt de fapt? Repară programul și spune în ce etapă s-a greșit.

P10. Programul ar trebui să afișeze câte cutii de câte 12 ouă se umplu complet cu `n` ouă. Găsește cele **două** erori de scriere. Apoi adaugă un rând care afișează câte ouă rămân. Ce afișează pentru `30`?

```
n = int(input()
a = n // 12
print("Cutii:" a)
```

P11. *Traducere greșită.* Pseudocodul afișează diferența dintre numărul mai mare și cel mai mic. Ce afișează traducerea în Python pentru `7` și `3`? Repară-o.

```
citește a, b
dacă a > b atunci scrie a - b
altfel scrie b - a
```

```
a = int(input())
b = int(input())
if a > b:
    print(a - b)
print(b - a)
```

P12. *Rânduri amestecate.* Ordonează rândurile astfel încât programul să transforme un număr de secunde în ore, minute și secunde. Un rând este în plus. Ce afișează pentru `3725`?

```
A)  m = r // 60
B)  s = int(input())
C)  h = s // 3600
D)  r = s % 3600
E)  print(h, m, r % 60)
F)  r = s // 3600
```

P13. Un elev a testat programul de mai jos (maximul a două numere) pe `(9, 4)`, `(7, 2)` și `(5, 5)`: toate au trecut. Găsește un test pe care pică, apoi repară programul. Ce au greșit testele alese?

```
a = int(input())
b = int(input())
m = a
if b > a:
    m = a
print(m)
```

P14. *Citește mesajul de eroare.* Pentru fiecare mesaj, spune ce a greșit programatorul sau utilizatorul.

```
a)  IndentationError: unexpected indent
b)  ValueError: invalid literal for int() with base 10: '3.5'
c)  ZeroDivisionError: integer division or modulo by zero
```

Nivelul 3 · Scrie cod

★★ scrie cod

Pentru fiecare program: analiza (intrare → ieșire), apoi codul, apoi **toate** testele.

P15. Pizza. O pizza are 8 felii, iar fiecare dintre cele `n` persoane mănâncă 3 felii. Câte pizza comanzi? Cu ce problemă din lecție seamănă?

intrare	8	3	1	0	16
ieșire	3	2	1	0	6

P16. Termometrul. Citește o temperatură în grade Celsius (număr întreg) și afișează-o în grade Fahrenheit: `F = C · 9 / 5 + 32`.

intrare	0	100	-40	37
ieșire	`32.0`	`212.0`	`-40.0`	`98.6`

P17. Restul. Plătești cu o bancnotă de 100 lei un produs de `p` lei. Afișează restul, câte bancnote de 10 lei și câte monede de 1 leu primești (cât mai multe de 10). Dacă `p > 100`, afișează `Bani insuficienți`.

intrare	ieșire
`37`	`63 6 3`
`100`	`0 0 0`
`91`	`9 0 9`
`120`	`Bani insuficienți`

P18. Ora sosirii. Pleci la ora `h` și minutul `m` (citite pe rânduri separate), iar drumul durează `d` minute. Afișează ora și minutul sosirii (după 23:59 urmează 0:00).

intrare (`h` , `m` , `d`)	ieșire
`14` , `50` , `25`	`15 15`
`23` , `30` , `45`	`0 15`
`0` , `0` , `0`	`0 0`
`22` , `0` , `1500`	`23 0`

P19. Trei reprezentări. Suma `1 + 2 + … + n` se calculează cu formula lui Gauss `n · (n + 1) / 2`. Scrie algoritmul ca schemă logică, în pseudocod și în Python (în Python folosește `//`, ca rezultatul să fie întreg).

intrare	1	10	0	100000
ieșire	1	55	0	5000050000

P20. Parcarea. Prima oră costă 5 lei, iar fiecare oră **începută** după ea costă 3 lei. Citește durata în minute și afișează prețul (0 minute = 0 lei).

intrare	0	30	60	61	150
ieșire	0	5	5	8	11

Nivelul 4 · Pune mintea la contribuție

★★★ pune mintea la contribuție

P21. Problema inversă. Pentru schema logică de la P3, ce valori naturale ale lui `x` duc la afișarea lui `10`? Găsește-le pe **toate** și explică de ce nu există altele.

P22. De ce funcționează? Numărul de autobuze de 50 de locuri pentru `n` elevi se poate calcula și fără `if`, cu `(n + 49) // 50`. Explică de ce formula dă rezultatul corect, scriind `n = 50 · q + r`, cu `0 ≤ r ≤ 49`. Ce formulă similară ai folosi la P15?

P23. Alege reprezentarea. Ai un algoritm cu o buclă în altă buclă. Ce reprezentare alegi ca să-l explici: a) unui elev de clasa a VI-a; b) unui coleg, pe o foaie de examen; c) calculatorului? Argumentează fiecare alegere într-o propoziție.

P24. Mini-proiect: cantina școlii. Elevii aleg din meniu, primesc totalul de plată și un bon. a) Descompune aplicația în 4–5 module. b) Pentru modulul „calculul totalului", fă analiza: date de intrare, date de ieșire, cazuri speciale. c) Ce mod de comunicare ar folosi elevul la casă și ce mod ar folosi cantina pentru vânzările din fiecare zi?

P25. Două tipuri de autobuze. Un autobuz mare are 50 de locuri și costă 1000 lei, unul mic are 20 de locuri și costă 500 lei. Fă analiza și proiectarea (în pseudocod) unui algoritm care găsește costul minim pentru `n` elevi. Calculează de mână costul minim pentru `n = 70`, `110`, `130` și spune ce autobuze iei.

P26. Diagrama etapelor. Alege una dintre problemele P16, P18 sau P20. a) Desenează diagrama etapelor de elaborare (analiză, proiectare, implementare, testare, depanare), cu câte o propoziție pentru ce faci concret în fiecare etapă la problema aleasă. b) Clasifică datele ei în date de intrare, de

ieșire și de manevră. c) Imaginează o greșeală care te întoarce de la testare la proiectare, nu la implementare.

Probleme de lucru în clasă (pbinfo)

Le rezolvăm împreună în ședință, în ordinea de mai jos.

- [ ] #941 **urare** — ușoară
  https://www.pbinfo.ro/probleme/941/urare
- [ ] #103 **curte** — ușoară
  https://www.pbinfo.ro/probleme/103/curte

Temă pentru acasă (pbinfo)

Trimite soluțiile pe pbinfo și notează punctajul obținut. Bifează ce ai terminat.

- [ ] #182 **cub** — ușoară
  https://www.pbinfo.ro/probleme/182/cub
- [ ] #1260 **asii** — ușoară
  https://www.pbinfo.ro/probleme/1260/asii