

LIBRIS

We know  
books

Admiterea la Facultatea de Matematică și Informatică  
Universitatea Babeș-Bolyai Cluj-Napoca

PROBLEME ȘI TESTE  
PREGĂTITOARE

INFORMATICĂ



UNIVERSITATEA BABEȘ-BOLYAI  
BABEȘ-BOLYAI TUDOMÁNYEGYETEM  
BABEȘ-BOLYAI UNIVERSITÁT  
BABEȘ-BOLYAI UNIVERSITY

TRADITIO ET EXCELLENTIA

# Cuprins

|            |   |            |
|------------|---|------------|
| <b>I</b>   | <b>Teorie și probleme</b>                         | <b>1</b>   |
| 1          | Algoritmi   | 2          |
| 2          | Tipuri de date. Operatori.                        | 6          |
| 3          | Tipuri structurate de date                        | 15         |
| 4          | Algoritmi elementari                              | 31         |
| 5          | Complexitatea algoritmilor                        | 81         |
| 6          | Subprograme                                       | 86         |
| 7          | Recursivitate                                     | 95         |
| 8          | Metodele Backtracking, Divide et Impera și Greedy | 114        |
| 9          | Combinatorică                                     | 135        |
| 10         | Grafuri   | 154        |
| <b>II</b>  | <b>Teste</b>                                      | <b>197</b> |
| 11         | Admitere 2021 - 2024                              | 198        |
| 12         | Concurs 2021 - 2024                               | 302        |
| 13         | Antrenament                                       | 351        |
| <b>III</b> | <b>Răspunsuri și indicații</b>                    | <b>448</b> |
| 14         | Răspunsuri  | 449        |
| 15         | Rezolvări   | 456        |
| 16         | Propunători                                       | 608        |

## Acest capitol acoperă

- Noțiunea de algoritm
- Date, variabile, expresii, operații
- Structuri de bază (liniară, alternativă și repetitivă)

## 1.1 Teorie

### 1.1.1 Introducere

Un algoritm reprezintă un set finit de pași bine definiți, care descriu o secvență logică de acțiuni necesare pentru a rezolva o problemă sau pentru a obține un rezultat specific. Acesta poate fi implementat ca:

- **Algoritm principal:** conține logica principală a programului și poate include operații de citire/scriere
- **Procedură (subalgoritm):** o secvență de instrucțiuni care efectuează o sarcină specifică, comunică doar prin parametri și nu returnează valori
- **Funcție (subalgoritm):** similar cu procedura, dar returnează întotdeauna o valoare și nu ar trebui să aibă efecte laterale (precum citire/scriere)

**Notă:** Subalgoritmii (proceduri și funcții) nu ar trebui să efectueze operații de citire sau scriere direct. Acestea ar trebui realizate în algoritmul principal sau în subalgoritmii special dedicați pentru operații I/O. Comunicarea între subalgoritmi se face exclusiv prin parametri (pentru proceduri) și prin valoarea returnată (pentru funcții).

Datele sunt informațiile procesate de către un algoritm și pot fi de diferite tipuri, precum numerice, logice, textuale, etc. O variabilă este un spațiu de memorie care poate stoca o valoare ce poate fi modificată pe parcursul execuției unui algoritm. Fiecare variabilă este identificată printr-un nume și are un tip de date asociat. O operație este compusă din operanzi și un operator. Operanzii reprezintă datele asupra cărora se aplică operația, iar operatorul este simbolul care indică tipul de operație efectuată asupra acestora. În funcție de scopul acestora, operațiile pot fi clasificate în aritmetice, logice sau relaționale.

Operanzii pot fi constante, variabile, literali, rezultatele unor funcții, rezultatele altor operații. O expresie este o operație care are ca operanzi alte operații.

Interacțiunea cu utilizatorul sau cu fișierele externe se face prin instrucțiuni de intrare (citire) și ieșire (scriere). Aceste operații asigură comunicarea dintre program și mediul înconjurător. În cele ce urmează, vom utiliza instrucțiunile `Write` și `Return` pentru a evidenția valorile afișate sau returnate de algoritmul respectiv.

## 1.1.2 Structuri de bază

- **Structura liniară:** expresiile care sunt evaluate secvențial pe o singură linie de cod, precum declarații, atribuirii și operații similare.

## Structura liniară

---

 $a \leftarrow 3$ 
 $b \leftarrow 5$ **Write**  $a + b$ 

▷ se afișează 8

- 
- **Structura alternativă (condițională):** permite luarea unei decizii prin executarea unei ramuri de cod, în funcție de îndeplinirea unei condiții.

## Structura condițională If

1: **If**  $n \text{ MOD } 2 = 0$  **then**2:     **Write** "par"3: **Else**4:     **Write** "impar"5: **EndIf**▷ se afișează "par" dacă  $n$  este par sau "impar" altfel

- 
- **Structura repetitivă:** execută un set de instrucțiuni de mai multe ori, de obicei în funcție de o condiție specifică.

## Exemplu: Structura repetitivă cu test inițial For

1: **For**  $i \leftarrow 0, 9$  **execute**2:     **Write**  $i, ' '$ 3: **EndFor**

▷ se afișează toate cifrele separate de un spațiu

## Exemplu: Structura repetitivă cu test inițial While

1:  $i \leftarrow 0$ 2: **While**  $i \leq 9$  **execute**3:     **Write**  $i, ' '$ 4:      $i \leftarrow i + 1$ 5: **EndWhile**

▷ se afișează toate cifrele separate de un spațiu

## Exemplu: Structura repetitivă cu test final Do...While

1:  $i \leftarrow 0$ 2: **Do**3:     **Write**  $i, ' '$ 4:      $i \leftarrow i + 1$ 5: **While**  $i \leq 9$ 

▷ se afișează toate cifrele separate de un spațiu

## 1.2 Probleme

1. Fie algoritmul `ceFace(a, b)`, unde  $a$  și  $b$  sunt numere întregi:

```
Algorithm ceFace(a, b)
  a ← a DIV b
  b ← a * b
  a ← b DIV a
  Write a, ' ', b
EndAlgorithm
```

Precizați valorile inițiale ale lui  $a$  și  $b$  astfel încât algoritmul să afișeze 10 20.

- A.  $a = 18, b = 20$   
 B.  $a = 28, b = 10$   
 C.  $a = 20, b = 10$   
 D.  $a = 10, b = 20$

2. Fie  $a$  și  $b$  două numere întregi ( $-10^4 \leq a, b \leq 10^4$ ). Precizați care dintre următorii algoritmi realizează corect interschimbarea valorilor  $a$  și  $b$  și afișează valorile acestora interschimbate. De exemplu: Pentru  $a = 12$  și  $b = 6$ , algoritmul ar trebui să afișeze 6 12.

A.

```
Algorithm SWAP1(a, b)
  a ← b + a
  b ← a - b
  a ← b - a
  Write a, ' ', b
EndAlgorithm
```

B.

```
Algorithm SWAP2(a, b)
  a ← a * b
  b ← a DIV b
  a ← a DIV b
  Write a, ' ', b
EndAlgorithm
```

C.

```
Algorithm SWAP3(a, b)
  a ← a DIV b
  b ← a * b
  a ← b DIV a
  Write a, ' ', b
EndAlgorithm
```

D.

```
Algorithm SWAP4(a, b)
  a ← a - b
  b ← a - b
  a ← b + a
  Write a, ' ', b
EndAlgorithm
```

3. Fie algoritmul `pink(a, b, k)`, unde  $a, b, k$  sunt numere naturale ( $1 \leq a, b, k \leq 10^4$ ).

```
Algorithm PINK(a, b, k)
  If a > b then
    a ← a ^ b
    b ← a ^ b
    a ← a ^ b
  EndIf
  While a ≤ b execute
    a ← a + k
    b ← b - k
    Write a, ' '
  EndWhile
EndAlgorithm
```

Operatorul  $\wedge$  este operatorul XOR pe biți; tabelul de adevăr este următorul:

|          |   |   |
|----------|---|---|
| $\wedge$ | 0 | 1 |
| 0        | 0 | 1 |
| 1        | 1 | 0 |

Exemplu:  $3 \wedge 5$  convertit în binar este  $011 \wedge 101 = 110 = 6$ , iar  $1 \wedge 4$  convertit în binar este  $001 \wedge 100 = 101 = 5$ .

Care dintre următoarele afirmații de mai jos nu sunt adevărate?

- A. Numărul de valori afișate de algoritm este egal cu  $(b - a) \text{ DIV } (k \cdot 2) + 1$ .
- B. Pentru  $a = 12$ ,  $b = 100$  și  $k = 4$ , algoritmul afișează 11 valori.
- C. Pentru  $a = 70$ ,  $b = 20$  și  $k = 6$ , algoritmul afișează 26 32 38 44 50.
- D. Algoritmul afișează întotdeauna valori din intervalul  $[a, (\frac{a+b}{2})]$ , multiple de  $k$ .
4. Fie algoritmi  $\text{good}(a, b)$  și  $\text{bad}(a, b)$ , unde  $a$  și  $b$  sunt numere întregi ( $-10^4 \leq a, b \leq 10^4$ ).

```

Algorithm GOOD(a, b)
  a ← a * b
  b ← a DIV b
  a ← a DIV b
  Write a, ' ', b
EndAlgorithm

```

```

Algorithm BAD(a, b)
  a ← a - b
  _____
  a ← b - a
  Write a, ' ', b
EndAlgorithm

```

Alegeți variantele care completează corect spațiul subliniat de mai sus astfel încât cei doi algoritmi să afișeze mereu aceleași valori pentru  $a$  și  $b$ , indiferent de valorile inițiale ale acestora.

- A.  $b \leftarrow a + b$
- B.  $b \leftarrow a - b$
- C.  $b \leftarrow b - a$
- D.  $b \leftarrow a * b \text{ DIV } a + a$
5. Fie algoritmul  $\text{container}(n)$ , unde  $n$  este un număr natural ( $1 \leq n \leq 10^4$ ).

```

Algorithm CONTAINER(n)
  a ← 0
  b ← 1
  While n > 1 execute
    a ← a + 1
    b ← b * 2
    n ← n - b
  EndWhile
  Return a
EndAlgorithm

```

Care dintre următoarele afirmații de mai jos sunt adevărate?

- A. Pentru  $n = 1024$ , algoritmul returnează 9.
- B. Pentru  $n = 336$ , algoritmul returnează 8.
- C. Algoritmul returnează întotdeauna o putere a lui 2.
- D. Algoritmul  $\text{container}(n)$  returnează cel mai mare număr  $k$ , cu proprietatea că numărul  $2^k$  este mai mic sau egal decât  $n$ .
6. Fie expresia  $E = n \text{ DIV } 100 + n \text{ MOD } 100 \text{ DIV } 10 + n \text{ MOD } 10$ , unde  $n$  este un număr natural nenul ( $1 \leq n < 10^3$ ). Precizați pentru câte valori ale lui  $n$ , care respectă specificațiile din enunț, expresia  $E$  are valoarea 9.

- A. 9
- B. 10
- C. 45
- D. 55

## Acest capitol acoperă

- Care sunt principalele tipuri de date și cum se reprezintă?
- Care sunt tipurile de operatori și cum sunt aceștia utilizați?

## 2.1 Teorie

### 2.1.1 Noțiunea de tip de date

Tipul de dată definește domeniul de valori pe care le poate lua o variabilă și operațiile permise asupra acesteia. În limbajele de programare, tipurile de date pot fi: numerice întregi, numerice cu virgulă mobilă (detaliat mai jos), caracter, logice sau structurate (tablou, structură).

### 2.1.2 Definirea tipurilor de date

Tipurile de date fundamentale pot fi descrise, din punct de vedere numeric, prin intervale ce derivă din numărul de biți alocați fiecărui tip. De obicei, un tip de date reprezentat pe  $n$  biți (cu semn, folosind complementul lui 2 - se va detalia în cadrul materiei Arhitectura Sistemelor de Calcul) poate reprezenta valori în intervalul  $[-2^{n-1}, 2^{n-1} - 1]$ , iar același tip, fără semn, poate reprezenta valori în  $[0, 2^n - 1]$ . Totuși, pentru tipurile uzuale de date avem:

- **char** (8 biți):
  - **signed char**:  $[-2^7, 2^7 - 1]$ ;
  - **unsigned char**:  $[0, 2^8 - 1]$ .
- **short** (16 biți):
  - **short**:  $[-2^{15}, 2^{15} - 1]$ ;
  - **unsigned short**:  $[0, 2^{16} - 1]$ .
- **int** (Minim 16 biți, dar poate fi și mai mult, în funcție de limbaj):
  - **int**: **Minim**  $[-2^{15}, 2^{15} - 1]$ ;
  - **unsigned int**: **Minim**  $[0, 2^{16} - 1]$ .
- **long** (Minim 32 biți, dar poate fi și mai mult, în funcție de limbaj):
  - **long**: **Minim**  $[-2^{31}, 2^{31} - 1]$ ;
  - **unsigned long**: **Minim**  $[0, 2^{32} - 1]$ .
- **long long** (Minim 64 biți, dar poate fi și mai mult, în funcție de limbaj):
  - **long**: **Minim**  $[-2^{63}, 2^{63} - 1]$ ;

**EBRIS** | We know books.

– unsigned long: Minim  $[0, 2^{64} - 1]$ .

- **float și double:** Acestea reprezintă numere reale în format cu virgulă mobilă, codificate conform standardului IEEE 754. Intervalele nu se exprimă simplu sub formă de puteri ale lui 2 pentru întregul domeniu numeric, deoarece reprezintă numere în formă normalizată (se va detalia în cadrul materiei Logică computațională). Totuși, precizia lor în biți este:

- float: 32 biți (aprox. 24 biți pentru mantisă);
- double: 64 biți (aprox. 53 biți pentru mantisă).

### 2.1.3 Operatori

Operatorii sunt de 3 tipuri: aritmetici, logici și relaționali. Cei pe care noi îi vom folosi în cele ce urmează sunt:

- **Operatori aritmetici:** +, -, \*, /, %;
- **Operatori logici:** AND, OR, NOT;
- **Operatori relaționali:** <, >, ≤, ≥, =, ≠

Operatorii aritmetici sunt evaluați conform regulilor standard de prioritate:

- Operatorii \*, / și % au prioritate mai mare decât + și -;
- Expresiile grupate prin paranteze au prioritate în timpul evaluării unei expresii;
- Pentru operatorii de aceeași prioritate se aplică evaluarea de la stânga la dreapta.

Operatorii logici operează pe valori de adevăr (True sau False) și se evaluează după următoarele principii:

- **AND:** Returnează True doar dacă ambii operanzi sunt True;
- **OR:** Returnează True dacă cel puțin unul dintre operanzi este True;
- **NOT:** Inversează valoarea operandului (adică, NOT False = True și NOT True = False).

| A     | B     | A AND B | A OR B |
|-------|-------|---------|--------|
| False | False | False   | False  |
| False | True  | False   | True   |
| True  | False | False   | True   |
| True  | True  | True    | True   |

Tabela 2.1 Tabela de adevăr pentru operatorii logici AND și OR.

Operatorii relaționali compară două valori și returnează un rezultat boolean, după cum urmează:

- **<:** Verifică dacă valoarea din stânga este mai mică decât cea din dreapta.
- **>:** Verifică dacă valoarea din stânga este mai mare decât cea din dreapta.

- $\leq$ : Verifică dacă valoarea din stânga este mai mică sau egală cu cea din dreapta.
- $\geq$ : Verifică dacă valoarea din stânga este mai mare sau egală cu cea din dreapta.
- $=$ : Verifică egalitatea dintre cele două valori.
- $\neq$ : Verifică inegalitatea dintre cele două valori.

Evaluarea expresiilor relaționale se efectuează după evaluarea expresiilor aritmetice, respectând regulile standard de comparare.

## 2.2 Probleme

7. Fie variabila  $n$  care memorează un număr natural. Care dintre expresiile de mai jos are valoarea `True` dacă și numai dacă  $n$  este divizibil cu 5 și cu 9?
- A.  $(n \text{ MOD } 5 \neq 1) \text{ AND } (n \text{ MOD } 9 = 0)$   
 B.  $((n \text{ DIV } 10) \text{ MOD } 5 \neq 1) \text{ AND } (n \text{ DIV } 9 = 0)$   
 C.  $(n \text{ MOD } 5 \neq 1) \text{ OR } (n \text{ MOD } 9 = 0)$   
 D.  $(n \text{ MOD } 5 = 0) \text{ AND } (n \text{ MOD } 9 = 0)$
8. Fie variabila  $y$  care memorează un număr natural. Care dintre expresiile de mai jos are valoarea `True` dacă și numai dacă  $y$  este mai mare decât 10 și nu este divizibil cu 4?
- A.  $(y > 10) \text{ AND } (y \text{ MOD } 4 \neq 0)$   
 B.  $(y > 10) \text{ OR } (y \text{ MOD } 4 = 0)$   
 C.  $\text{NOT}(y \leq 10) \text{ AND } (y \text{ MOD } 4 = 0)$   
 D.  $(y > 10) \text{ AND } (y \text{ MOD } 2 \neq 0)$
9. Fie variabila  $k$  care memorează un număr natural. Care dintre expresiile de mai jos are valoarea `True` dacă și numai dacă  $k$  este un număr par mai mic decât 20?
- A.  $(k \text{ MOD } 2 = 0) \text{ AND } (k < 20)$   
 B.  $(k \text{ MOD } 2 \neq 0) \text{ AND } (k \geq 20)$   
 C.  $(k \text{ MOD } 2 = 0) \text{ OR } (k \geq 20)$   
 D.  $(k \text{ MOD } 2 = 0) \text{ AND } (k \leq 20)$
10. Fie variabila  $m$  care memorează un număr întreg. Care dintre expresiile de mai jos este `True` dacă și numai dacă  $m$  este un număr negativ divizibil cu 8?
- A.  $(m > 0) \text{ AND } (m \text{ MOD } 8 \neq 0)$   
 B.  $(m < 0) \text{ OR } (m \text{ MOD } 8 \neq 0)$   
 C.  $(m < 0) \text{ AND } (m \text{ MOD } 8 = 0)$   
 D.  $(m \text{ MOD } 8 = 0) \text{ AND } (m > 0)$
11. Fie variabilele  $a = 16$ ,  $b = 4$  și  $c = 2$ . Ce valoare va avea expresia următoare?

$$(a \text{ MOD } b + b * c) \text{ DIV } (a \text{ DIV } b - c) + ((a + 1) \text{ MOD } (b - 1))$$



A. 4

B. 5

C. 6

D. 7

12. Fie variabilele  $x = 20$ ,  $y = 7$  și  $z = 3$ . Ce valoare va avea expresia următoare?

$$\left( (x + 1) \text{ MOD } y + z \right) * \left( (x - 1) \text{ DIV } z \right) - \left( y \text{ MOD } (z - 1) \right)$$

A. 18

B. 17

C. 21

D. 23

13. Fie variabilele  $a = 10$ ,  $b = 5$  și  $c = 8$ . Ce valoare va avea  $a$  după executarea următoarei instrucțiuni?

$$a = a + (b * c) \text{ DIV } (c \text{ MOD } (b + 2)) - ((c - 1) * (b + 2))$$

A. 1

B. 8

C. 9

D. 10

14. Fie variabilele  $x = 8$ ,  $y = 15$  și  $z = 5$ . Ce valoare va avea următoarea expresie?

$$\left( (x \text{ MOD } (y - 1)) + (z * (x + 1)) \right) \text{ DIV } ((y - 1) \text{ MOD } (z - 1))$$

A. 25

B. 26

C. 27

D. 28

15. Se consideră următoarea expresie logică:

$$(A \text{ OR } B) \text{ AND } (\text{NOT } A \text{ OR } C) \text{ OR } (B \text{ AND } (\text{NOT } C \text{ OR } A))$$

Precizați pentru ce valori ale lui  $A, B, C$ , expresia are valoarea True.

A.  $A = \text{True}, B = \text{False}, C = \text{True}$ C.  $A = \text{True}, B = \text{True}, C = \text{False}$ B.  $A = \text{False}, B = \text{True}, C = \text{False}$ D.  $A = \text{False}, B = \text{False}, C = \text{True}$ 

16. Se consideră următoarea expresie logică:

$$(\text{NOT } C \text{ AND } B) \text{ OR } (A \text{ AND NOT } B) \text{ AND } (\text{NOT } A \text{ OR } C)$$

Precizați pentru ce valori ale lui  $A, B, C$ , expresia are valoarea True.

A.  $A = \text{True}, B = \text{False}, C = \text{False}$ C.  $A = \text{True}, B = \text{True}, C = \text{False}$ B.  $A = \text{False}, B = \text{True}, C = \text{False}$ D.  $A = \text{False}, B = \text{False}, C = \text{True}$ 

17. Se consideră următoarea expresie logică:

$$((A \text{ AND } (\text{NOT } B \text{ OR } C)) \text{ OR } ((\text{NOT } A \text{ AND } B) \text{ AND } (\text{NOT } C \text{ OR } A))) \text{ AND } (B \text{ OR NOT } C)$$

Precizați pentru ce valori ale lui  $A, B, C$ , expresia are valoarea True.

A.  $A = \text{True}, B = \text{False}, C = \text{False}$ C.  $A = \text{True}, B = \text{True}, C = \text{False}$ B.  $A = \text{False}, B = \text{True}, C = \text{False}$ D.  $A = \text{True}, B = \text{False}, C = \text{True}$ 

18. Se consideră următoarea expresie logică:

$$(\text{NOT}((A \text{ OR } B) \text{ AND } C) \text{ OR } ((B \text{ AND } \text{NO } C) \text{ OR } (\text{NOT } A \text{ AND } (C \text{ OR } \text{NOT } B)))) \text{ AND } A$$

Precizați pentru ce valori ale lui  $A, B, C$ , expresia are valoarea True.

A.  $A = \text{True}, B = \text{True}, C = \text{False}$ C.  $A = \text{True}, B = \text{False}, C = \text{False}$ B.  $A = \text{False}, B = \text{False}, C = \text{True}$ D.  $A = \text{True}, B = \text{False}, C = \text{True}$ 

19. Se consideră următoarea expresie logică:

$$(((A \text{ AND } B) \text{ OR } (\text{NOT } C \text{ AND } A)) \text{ AND } (B \text{ OR } \text{NOT } A)) \text{ OR } ((\text{NOT } B \text{ AND } C) \text{ AND } (\text{NOT } A \text{ OR } B))$$

Precizați pentru ce valori ale lui  $A, B, C$ , expresia are valoarea True.

A.  $A = \text{True}, B = \text{False}, C = \text{False}$ C.  $A = \text{True}, B = \text{False}, C = \text{True}$ B.  $A = \text{False}, B = \text{False}, C = \text{True}$ D.  $A = \text{True}, B = \text{True}, C = \text{False}$ 

20. Se consideră următoarea expresie logică:

$$(((A \text{ OR } (\text{NOT } B \text{ AND } C)) \text{ AND } (\text{NOT } A \text{ OR } B)) \text{ OR } ((\text{NOT } C \text{ OR } A) \text{ AND } (\text{NOT } B \text{ OR } \text{NOT } A))) \text{ AND } C$$

Precizați pentru ce valori ale lui  $A, B, C$ , expresia are valoarea False.

A.  $A = \text{True}, B = \text{True}, C = \text{False}$ C.  $A = \text{False}, B = \text{True}, C = \text{False}$ B.  $A = \text{True}, B = \text{False}, C = \text{True}$ D.  $A = \text{False}, B = \text{False}, C = \text{True}$ 

21. Fie variabilele  $p = 12, q = 3$  și  $r = 4$ . Ce valoare va avea expresia următoare?

$$\left( ((p+1) \text{ MOD } q) + (r * (p+1)) \right) - \left( q \text{ MOD } (r-1) \right) + \left( (p+1) \text{ DIV } (q+2) \right)$$

A. 50;

B. 52;

C. 55;

D. 56.

22. Fie variabilele  $a = 9, b = 2$  și  $c = 5$ . Ce valoare va avea expresia următoare?

$$\left( (a-1) \text{ MOD } b \right) + \left( b * (c+1) \right) - \left( (a-1) \text{ DIV } b \right)$$

A. 7;

B. 8;

C. 9;

D. 10.

23. Fie variabilele  $x = 25, y = 4$  și  $z = 2$ . Ce valoare va avea următoarea expresie?

$$(x \text{ MOD } y) + \left( (y+1) * z \right) - \left( x \text{ DIV } (y+1) \right)$$