

Alexandra TRANDABUR

ÎNDRUMAR BACALAUREAT LA INFORMATICĂ

Structurare teoretică, exemplificări
și sugestii metodologice



Editura PASTEL, Brașov, 2020

CUPRINS

ARGUMENT	5
CAPITOLUL I	7
SUBIECTUL I AL PROBEI SCRISE LA INFORMATICĂ	7
I.1. Expresii și funcții predefinite	7
I.2. Stiva și coada alocate static	18
I.3. Grafuri neorientate	21
I.4. Arbori	28
I.5. Grafuri orientate	31
I.6. Recursivitate	38
I.7. Backtracking	44
CAPITOLUL II	57
SUBIECTUL II AL PROBEI SCRISE LA INFORMATICĂ	57
II.1. Subpunctul 1a)	57
II.2. Subpunctul 1b)	59
II.3. Subpunctul 1c)	60
II.4. Subpunctul 1d)	60
II.5. Structuri	75
CAPITOLUL III	83
SUBIECTUL III AL PROBEI SCRISE LA INFORMATICĂ	83
III.1. Subprograme	83
III.2. Matrice (tablouri bidimensionale)	95
III.3. Șiruri de caractere	105
III.4. Complexitatea algoritmilor	117
BIBLIOGRAFIE	137

CAPITOLUL I

SUBIECTUL I AL PROBEI SCRISE LA INFORMATICĂ

Subiectul I constă în cinci itemi de tip grilă care vizează câte unul dintre următoarele capitole:

- **expresii și funcții predefinite;**
- **cozi și stive alocate static;**
- **grafuri neorientate;**
- **arbori;**
- **grafuri orientate;**
- **recursivitate;**
- **backtracking.**

Doi itemi pot cuprinde întrebări și din capitolele **Tablouri bidimensionale (matrice)**, respectiv **Șiruri de caractere**, dar acestea vor fi tratate în capitolul III.

I.1. Expresii și funcții predefinite

Obiectele cu ajutorul cărora lucrează algoritmi

Se disting două tipuri de obiecte:

A. date: constante și variabile

B. expresii: aritmetice și logice (boolene)

 **Expresia** reprezintă o combinație validă de operatori și operanzi, care este folosită în scopul efectuării calculelor. **Operanzii** desemnează valorile ce intră în calcul (ei pot fi variabile sau constante), iar **operatorii** sunt operațiile care se execută în cadrul expresiei, ei fiind reprezentați cu ajutorul unor simboluri.

Exemplu:

În expresia $x-7$, operanzii sunt x și 7 , iar operatorul este $-$.

❗ Un operand poate fi la rândul său o expresie, caz în care va fi delimitat de paranteze rotunde.

Exemplu:

În expresia $(x - 7) * (y - 5)$, avem operanzii: $x - 7$ și $y - 5$ și operatorul $*$. Operandul $x - 7$ este la rândul său o expresie alcătuită din operanzii x și 7 și operatorul $-$, la fel cum și operandul $y - 5$ este alcătuit din operanzii y și 5 și operatorul $-$.

📖 **Expresiile aritmetice** sunt expresii care efectuează operații matematice și au ca rezultat un număr.

În cadrul acestora sunt folosiți **operatorii aritmetici**:

+ pentru adunare
- pentru scădere } **operatori aditivi**

* pentru înmulțire
/ pentru împărțire
% pentru restul împărțirii } **operatori multiplicativi**

❗ 1) a/b returnează rezultate diferite în funcție de tipul variabilelor a și b . Astfel, dacă cel puțin una dintre variabilele a sau b este reală, atunci rezultatul va fi real, iar dacă ambele sunt întregi, atunci a/b va returna câtul împărțirii întregi lui a la b .

❗ 2) Operatorul $\%$ se poate aplica doar variabilelor de tip întreg.

Prioritatea operatorilor aritmetici (stabilește ordinea în care se execută operațiile): ca și la matematică, operatorii multiplicativi se execută primii, urmați de cei aditivi.

❗ 1) În cazul în care avem mai mulți operatori de aceeași prioritate, aceștia se vor executa, în ordine, de la stânga la dreapta.

Exemplu:

Fie expresia: $27/10*20/3$. Pentru rezolvarea corectă a acesteia se parcurg următoarele etape: $27/10*20/3 = 2*20/3 = 40/3 = 13$. Dacă am fi tentați, spre exemplu, să calculăm mai întâi împărțirile și apoi să realizăm înmulțirea, rezultatul obținut ar fi **greșit**: $27/10*20/3 = 2*6 = 12$.

! 2) Pentru a schimba prioritatea operatorilor se pot folosi parantezele rotunde.

Exemplu:

Dacă, pentru expresia anterioară, dorim ca mai întâi să se realizeze împărțirile, atunci vom scrie: $(27/10)*(20/3)$, iar rezultatul va fi: $(27/10)*(20/3) = 2*6 = 12$.

! 3) În informatică, spre deosebire de matematică, se folosesc doar parantezele rotunde. Așadar, o expresie poate să conțină mai multe perechi de paranteze rotunde, caz în care expresia se evaluează dinspre interior spre exterior.

Exemplu:

În expresia $((7 - 5)*(10 + 2) - 3)*4$ se rezolvă mai întâi parantezele $(7 - 5)$ și $(10 + 2)$, iar rezultatele lor sunt folosite în continuare în evaluarea expresiei, aceasta devenind: $(2*12 - 3)*4 = 21*4 = 84$.

📖 **Expresiile logice (boolene)** sunt reprezentări ale uneia sau mai multor condiții și au ca rezultat o valoare de adevăr: adevărat - condiție îndeplinită sau fals - condiție neîndeplinită.

! În informatică, pentru „adevărat” vom folosi valoarea 1, iar pentru „fals” valoarea 0.

Astfel, dacă vom fi întrebați: „care dintre următoarele expresii are valoarea 1?”, aceasta va însemna, de fapt, „care dintre următoarele expresii este adevărată?”.

Expresiile logice folosesc două tipuri de operatori: operatori logici și operatori relaționali.

- **Operatorii logici** sunt: și logic (AND), sau logic (OR) și negația logică (NOT).
- **Operatorii relaționali** sunt: $>$, $\geq$, $<$, $\leq$, $\neq$, $=$.

Pentru a putea evalua expresiile în care apar mai multe condiții legate prin operatori logici, este necesar să cunoaștem anumite reguli, cuprinse în următoarele tabele de adevăr:

Tabelul de adevăr pentru **și logic (AND)**

a	b	a și b (a AND b)
adevărat	adevărat	adevărat
adevărat	fals	fals
fals	adevărat	fals
fals	fals	fals

Tabelul de adevăr pentru **sau logic (OR)**

a	b	a sau b (a OR b)
adevărat	adevărat	adevărat
adevărat	fals	adevărat
fals	adevărat	adevărat
fals	fals	fals

! 1) Dacă cel puțin una dintre condițiile a sau b, numite în informatică și propoziții logice, este falsă, atunci și rezultatul expresiei „a și b” este fals.

! 2) Dacă cel puțin una dintre condițiile a sau b este adevărată, atunci și rezultatul expresiei „a sau b” este adevărat.

Tabelul de adevăr pentru **negația logică (NOT)**

a	NOT a
adevărat	fals
fals	adevărat

Negația aplicată operatorilor logici și relaționali:

Operator	NOT operator
$>$	$\leq$
$<$	$\geq$
$\geq$	$<$
$\leq$	$>$
$=$	$\neq$
$\neq$	$=$
AND	OR
OR	AND

Prioritatea operatorilor logici și relaționali:

Prima dată se execută negația logică, urmată de operatorii relaționali și ceilalți doi operatori logici AND și OR, în această ordine.

! Pentru a schimba prioritatea operatorilor se pot folosi parantezele rotunde.

Exemplu:

Fie expresia: $3 < 5$ sau $7 > 9$ și $10 > 20$. În evaluarea corectă a acesteia se parcurg următorii pași: (A) sau $7 > 9$ și $10 > 20 \Rightarrow$ (A) sau (F) și $10 > 20 \Rightarrow$ (A) sau (F) și (F) $\Rightarrow$ (A) sau (F) $\Rightarrow$ (A).

În cazul în care am dori ca mai întâi să se evalueze prima parte a expresiei, atunci aceasta ar trebui scrisă: $(3 < 5 \text{ sau } 7 > 9)$ și $10 > 20$, iar evaluarea sa ar fi: $((A) \text{ sau } (F))$ și (F) $\Rightarrow$ (A) și (F) $\Rightarrow$ (F).

Prioritatea operatorilor aritmetici, logici și relaționali:

Nivel prioritate	Operator
1 (cea mai mare prioritate)	negație logică (NOT)
2	*, /, %
3	+, -
4	>, ≥, <, ≤, ≠, =
5	și logic (AND)
6 (cea mai mică prioritate)	sau logic (OR)

❗ În informatică, unii operatori sunt reprezentați diferit față de notația cunoscută de la matematică. Astfel, corespondența este următoarea:

$\geq \rightarrow >=$ $= \rightarrow ==$ $\leq \rightarrow <=$ $\neq \rightarrow !=$

și logic (AND) $\rightarrow \&\&$ sau logic (OR) $\rightarrow |$

negație logică (NOT) $\rightarrow !$

FUNCȚII PREDEFINITE C/C++

1) **sqrt(x)** – returnează radicalul numărului x

Exemple:

sqrt(25) = 5; sqrt(12) = 3.4641

2) **abs(x)** – returnează modulul numărului x

Exemple:

abs(12) = 12; abs(-12) = 12

3) **floor(x)** – returnează cel mai apropiat întreg $\leq x$

Exemple:

floor(3.15) = 3; floor(3) = 3

4) **ceil(x)** – returnează cel mai apropiat întreg $\geq x$

Exemple:

ceil(3.15) = 4; ceil(3) = 3

❗ Pentru folosirea tuturor funcțiilor de mai sus trebuie inclusă biblioteca „cmath”.

În continuare sunt prezentate câteva exemple de exerciții cu expresii:

- 1) Variabilele x și y sunt de tip `int`. Care dintre expresiile C/C++ de mai jos are valoarea 1, dacă și numai dacă, valorile întregi nenule, memorate în variabilele x și y , sunt egale?

(Bacalaureat 2009 – Varianta 2).

- a. $(x \% y == 0) \ \&\& \ (y \% x == 0) \ \&\& \ (x * y > 0)$
- b. $(x \leq y) \ \&\& \ (y < x)$
- c. $(x \leq y) \ || \ (y \leq x)$
- d. $x * x == y * y$

Acest exercițiu face parte din categoria „mai multe răspunsuri par a fi corecte”. La prima vedere, doar varianta b) „pică”. Este foarte important să le analizăm pe toate! Celelalte răspunsuri par a fi corecte. Cum decidem totuși care dintre ele este cel cu adevărat corect. În enunț, apar câteva cuvinte care fac diferența, și anume: **„are valoarea 1, dacă și numai dacă”**, adică **„este adevărată, dacă și numai dacă”**.

Așadar, vom elimina acele expresii care sunt adevărate și pentru valori care nu îndeplinesc condiția de a fi egale.

Spre exemplu:

- pentru $x = 3$ și $y = 4$ (două valori care nu sunt egale) expresia de la c) este adevărată $\Rightarrow$ răspunsul c) este incorect.
- pentru $x = 3$ și $y = -3$ (două valori care nu sunt egale) expresia de la d) este adevărată $\Rightarrow$ răspunsul d) este incorect.

Rezultă că răspunsul a) este cel corect.

- 2) Care este valoarea expresiei C/C++: $50 - (100 - 300/2/(2 + 3))$?
(Bacalaureat 2009 – Varianta 40).

- a. -30
- b. 70
- c. -20
- d. 60

La tipul acesta de exerciții, dacă ținem cont de prioritatea operatorilor și de faptul că „/” execută împărțire întreagă, când cei doi termeni ai împărțirii sunt întregi, și reală, când cel puțin unul este real, nu avem ce să greșim.

Așadar: $50 - (100 - 300/2/(2 + 3)) = 50 - (100 - 300/2/5) = 50 - (100 - 150/5) = 50 - (100 - 30) = 50 - 70 = -20$.

- 3) Care este cea mai mică valoare pe care o poate avea expresia $C/C++$: $x/7-x\%7$, dacă variabila x , de tip `int`, memorează un număr natural cu o singură cifră? (Bacalaureat 2009 – Varianta 26).
- a. 0 b. 1.14 c. -6 d. 1

Pentru ca expresia să aibă valoare minimă înseamnă că trebuie să scădem cea mai mare valoare posibilă, adică $x\%7$ să fie maxim. $x\%7$ reprezintă restul împărțirii lui x la 7, deci valoarea maximă este 6. $x\%7 = 6$ pentru $x = 6 \Rightarrow$ răspuns corect c) -6.

- 4) Câte valori distincte, numere naturale, poate primi variabila x pentru ca valoarea expresiei $x/2/2$ scrisă în $C/C++$ să fie egală cu 1? (Bacalaureat 2009 – Varianta 28).
- a. 1 b. 4 c. 2 d. 0

Pentru ca $x/2/2$ să fie 1, $x/2$ poate fi 2 sau 3. Pentru $x/2 = 2$, x poate fi 4 sau 5, iar pentru $x/2 = 3$, x poate fi 6 sau 7. Așadar, 4 valori posibile $\Rightarrow$ răspuns corect b) 4.

- 5) Variabila x este de tip real. Care dintre următoarele expresii $C/C++$ are valoarea 1, dacă și numai dacă, numărul real memorat în variabila x nu aparține intervalului $(2,9]$? (Bacalaureat 2009 – Varianta 5).
- a. $(x > 2) \&\& (x \leq 9)$ b. $(x \leq 2) \&\& (x > 9)$
c. $(x \leq 2) \parallel (x > 9)$ d. $(x < 2) \parallel (x > 9)$