

INFORMATICĂ – VARIANTA 1

1. În următoarea secvență de cod variabilele p, m și s sunt de tip întreg.

```
p:=10; m:=12345; s:=0;
while(m>0){
    p:=p*10; s:=s+m%p; m=m/p;
}
```

```
p:=10; m:=12345; s:=0;
while m>0 do begin
    p:=p*10; s:=s+m mod p; m:=m div p;
end;
```

Care este ultima cifră (a unităților) a valorii memorate în s la sfârșitul execuției acestei secvențe de cod?

- a) 7 b) 5 c) 8 d) 9

2. În următoarea secvență de cod variabilele x și k sunt de tip întreg. Înainte de executarea acestei secvențe de cod, k este strict mai mare decât x. Stabiliți care este valoarea expresiei $\text{abs}(k - x)$ la sfârșitul executării secvenței, unde abs este o funcție care returnează modulul unui număr întreg primit ca parametru.

```
while (k > x - 3)
    k--;
x++; k--;
```

```
while k > x - 3 do
    k := k - 1;
inc(x); dec(k);
```

- a) 5 b) 4 c) 2 d) 1

3. În următorul algoritm descris în pseudocod, v este un vector de n elemente întregi, primul element fiind pe poziția 1. Se notează prin \leftrightarrow operația de interschimbare.

```
pentru j ← 1, 2 execută
    pentru i ← 1, n-1 execută
        dacă v[i] > v[i+1] atunci
            v[i] ↔ v[i+1]
```

Care este numărul maxim de interschimbări ce se pot realiza prin executarea algoritmului pentru n=5?

- a) 10 b) 8 c) 7 d) 9

4. În următorul algoritm a este o matrice cu n linii și n coloane având elemente întregi; liniile și coloanele matricei a sunt numerotate de la 1 la n. Variabilele i, j, s sunt de tip întreg.

```
s:=0; i:=1;
while(i<=n){
    j:=n;
    while(j>=1){
        if(i==j)
            s = s + a[i][j];
        j--;
    }
    i++;
}
```

```
s:=0; i:=1;
while i<=n do begin
    j:=n;
    while j>=1 do begin
        if i=j then
            s := s + a[i,j];
        dec(j);
    end;
    inc(i);
end;
```

Stabiliți ce reprezintă valoarea memorată în variabila s la finalul execuției algoritmului și care este complexitatea algoritmului.

- a) suma elementelor de pe diagonala principală / $O(n)$ b) suma elementelor de pe diagonala secundară / $O(n)$
c) suma elementelor de pe diagonala principală / $O(2n)$ d) suma elementelor de pe diagonala principală / $O(n^2)$

5. Se consideră următorul subprogram:

```
int doi(int n){
    int p:=1;
    while(n>1){
        n=n/2; p++;
    }
    return p;
}
```

```
function doi(n:integer):integer;
var p:integer;
begin
    p:=1;
    while n>1 do
        begin
            n:=n div 2; p:=p+1;
        end;
    doi := p;
end;
```

Pentru un număr real x notăm cu $[x]$ partea sa întreagă. Care afirmație este valabilă pentru valoarea returnată de apelul $\text{doi}(n)$, unde n este un număr natural strict pozitiv ?

- a) este egală cu $\lfloor \log_2(n) \rfloor$ b) este egală cu puterea la care apare 2 în descompunerea în factori primi a lui n
c) este egală cu $\lfloor \log_2(n) \rfloor + 1$ d) este un număr nenul dacă și numai dacă n este putere a lui 2

6. Câți dintre următorii vectori nu pot reprezenta vectorul de tați al unui arbore cu rădăcină?
 (3, 4, 0, 3, 4, 1, 2, 1, 2, 1), (0, 6, 1, 2, 8, 4, 1, 1, 1, 1), (0, 3, 4, 1, 2, 1, 2, 1, 2, 3), (0, 1, 2, 3, 4, 5, 6, 7, 6, 5)

a) 0

b) 1

c) 2

d) 3

7. Fie G un graf neorientat cu $n > 2$ noduri și m muchii. Numărul de subgrafuri ale lui G cu cel puțin două noduri este:

a) $2^m - 1$

b) $2^n - n - 1$

c) $2^n - n$

d) $2^m - 2$

8. Care sunt numărul minim și numărul maxim de arce ale unui graf orientat tare conex cu 10 vârfuri?

a) 10 și 45

b) 9 și 45

c) 10 și 90

d) 9 și 90

9. Generarea folosind metoda *backtracking* a tuturor șirurilor de 3 elemente, fiecare element putând fi orice număr din mulțimea $\{1, 2, 3, 4, 5\}$, se realizează cu ajutorul unui algoritmul echivalent cu cel de generare a:

a) permutărilor

b) aranjamentelor

c) combinărilor

d) produsului cartezian

10. Se consideră două variabile globale x și y , ambele inițializate cu valoarea 1 și următorul subprogram:

```
void f(int x){
    x+=3;
    y--x;
}
```

```
procedure f(x:integer);
begin
    inc(x,3); x:=x-1; y:=x;
end;
```

Care sunt valorile variabilelor globale x și y după execuția apelului $f(2)$?

a) 4 și 4

b) 4 și 5

c) 3 și 3

d) 1 și 4

11. Dacă G este un graf neorientat eulerian cu 10 noduri și 16 muchii și lista de adiacență a fiecărui nod din G este formată din cel puțin un element, atunci câte dintre afirmațiile de mai jos sunt adevărate?

- G este conex
- G are cel puțin un nod de grad egal cu 2
- G este hamiltonian
- G nu conține cicluri elementare de lungime 3.

a) 1

b) 2

c) 3

d) 4

12. Se consideră funcția f definită mai jos. Ce valoare va returna $f(1,2)$?

```
int f(int m, int n)
{
    if (m==0) return n+1;
    if (m>0 && n==0) return f(m-1,1);
    if (m>0 && n>0) return f(m-1,f(m,n-1));
}
```

```
function f(m,n:integer): integer;
begin
    if m=0 then f:= n+1;
    if (m>0) AND (n=0) then f:= f(m-1,1);
    if (m>0) AND (n>0) then f:= f(m-1,f(m,n-1));
end;
```

a) 1

b) 3

c) 2

d) 4

13. Fie f și g două subprograme cu definițiile de mai jos. Ce valoare va returna apelul $g(6)$?

```
int f(int x){
    if (x%2==0)
        return f(x/2);
    else return x;
}
int g(int x){
    if (x<1) return 1;
    else return f(x*g(x-1));
}
```

```
function f(x:integer): integer;
begin
    if x mod 2=0 then f:= f(x div 2)
    else f:= x;
end;
function g(x:integer): integer;
begin
    if x < 1 then g:=1
    else g:= f(x*g(x-1));
end;
```

a) 315

b) 3

c) 45

d) 15

14. Fie A , B și C 3 stive inițial vide. Se consideră că, în oricare dintre cele 3 stive, o valoare poate fi adăugată doar dacă este strict mai mică decât valoarea aflată în vârf sau dacă stiva este vidă. Printr-o mutare a unei valori înțelegem scoatere ei dintr-o stivă și adăugarea ei în altă stivă. Dacă în stiva A sunt introduse pe rând numerele 5, 4, 3, 2, 1 în această ordine, care este numărul minim de mutări de valori folosind cele 3 stive în urma cărora stiva B conține toate elementele care inițial erau în stiva A .

a) 5!

b) 2^5

c) $2^5 - 1$

d) 10

15. Se dau mulțimile A și B având același număr n de elemente. Reprezentăm mulțimile prin vectori sortați crescător. Care este complexitatea algoritmului optim de aflare a intersecției celor două mulțimi?

a) $O(n^2)$

b) $O(n \log(n))$

c) $O(n)$

d) $O(\log(n))$

→ interclasare