

## Subiecte la testul grilă de Informatică

1. Care dintre operatorii (`==`) și (`=`) pot fi aplicați unei variabile de tip structură ?

- (a) comparare egalitate (`==`)    (b) asignare (`=`)    (c) ambii operatori    (d) nici unul

2. Care dintre relațiile de mai jos indică apartenența variabilei  $x$  întregi la intervalul  $(-10, -2] \cap [-7, 11]$ .

- (a) `!((x <= -10)&&(x > 11))||((x < -7)&&(x > -2))`  
(b) `!((x <= -10)||((x > 11))&&((x < -7)||((x > -2))))`  
(c) `!((x <= -10)&&(x > 11))&&((x < -7)&&(x > -2))`  
(d) `!((x <= -10)||((x > 11))||((x < -7)||((x > -2))))`

3. Ce va afișa programul de mai jos:

```
int x = 5;
int main()
{
    int x = x;
    cout << x;
    return 0;
}
```

- (a) Eroare de compilare    (b) Valoare nedefinită    (c) 5    (d) 0

4. Ce se va afișa la rularea secvenței următoare?

```
int i, j, s = 0;

for (i = 0; i < 5; i = i + 2)
{
    for (j = 5; j > 3; j = i - 2)
    {
        s = s + 1;
    }
}

cout << "s = " << s;
```

- (a)  $s = 1$   
(b)  $s = 2$   
(c)  $s = 3$   
(d)  $s = 4$

5. Ce se afișează la rularea următoarei secvențe de cod (indexul primului element din vector este 0, astfel `x[0]` are valoarea 'a'):

```

{
    int x[] = {'a', 'b', 'c', 'j'}, i;

    for(i=0; i<4; i++)
    {
        if (x[i] < 'c')
        {
            if (x[i] == 3)
            {
                cout << "A";
            }
            else
            {
                cout << "B";
            }
        }
        else
        {
            if (x[i] < 'i')
            {
                if (x[i] > 'g' )
                {
                    cout << "C";
                }
            }
            else
            {
                cout << "D";
            }
        }
    }
    cout << "?";
}

```

- (a) AAD?
- (b) AAC?
- (c) ABCD?
- (d) BCD?

6. Într-un magazin de produse alimentare, prețul produselor scade cu cât acestea se apropie de data de expirare, după următorul algoritim: începând cu 5 zile înaintea datei expirării, prețul scade cu câte 10 % pe zi față de prețul inițial, ajungând în ziua expirării să fie redus cu 50%. Algoritmul de stabilire a prețurilor se rulează în fiecare dimineață înainte de deschiderea magazinului.

De exemplu, dacă un produs costă 100 de lei și expiră pe 20 decembrie 2019, prețul se va reduce pe 16 decembrie la 90 de lei și va ajunge pe 20 decembrie la 50 de lei.

Care este varianta corectă a subrutinei calculPret pentru realizarea acestui algoritim? Se consideră deja scrisă și funcțională o subrutină care calculează câte zile mai sunt până la expirarea produsului.

```

(a)
void calculPret ()
{
    if (zilePanaLaExpirare < 5)
        pretDeAfisat = pretInitial - pretInitial * ((5 - zilePanaLaExpirare) / 10.0);
    else

```

```

        pretDeAfisat = pretInitial;
    }

```

(b)

```

void calculPret ()
{
    if (zilePanaLaExpirare < 5)
        pretDeAfisat = pretInitial * ( (5 - zilePanaLaExpirare) / 10.0);
    else
        pretDeAfisat = pretInitial;
}

```

(c)

```

void calculPret ()
{
    if (zilePanaLaExpirare < 5)
        pretDeAfisat = pretInitial - (5 - zilePanaLaExpirare) / 10.0;
    else
        pretDeAfisat = pretInitial;
}

```

(d)

```

void calculPret ()
{
    if (zilePanaLaExpirare < 5)
        pretDeAfisat = pretInitial;
    else
        pretDeAfisat = pretInitial - (5 - zilePanaLaExpirare) / 10.0;
}

```

7. Care din următoarele declarații ale unui tablou bidimensional sunt corecte:

```

char m1[2][3] = {'a', 'b', 'c', 'd', 'e', 'f'};
char m2[ ][ ] = {'a', 'b', 'c', 'd', 'e', 'f'};
char m3[ ][2] = {'a', 'b', 'c', 'd', 'e', 'f'};
char m4[2][ ] = {'a', 'b', 'c', 'd', 'e', 'f'};

```

- (a) Doar m1
- (b) m3 și m1
- (c) Toate
- (d) m1, m2 și m4

8. Fie matricea  $m$ , care are elementul de pe prima linie și prima coloană cu indicele 0, 0:

```

1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5

```

Cum va arăta matricea  $m$  după rularea codului următor:

```

for (i = 0; i < 5; i++)
    for (j = i + 1; j < 5; j++)
    {

```

```

    aux = mat[i][j];
    mat[i][j] = mat[j][i];
    mat[j][i] = aux;
}

```

	1 2 3 4 5	1 1 1 1 1	5 5 5 5 5	5 5 5 5 5
	1 2 3 4 5	2 2 2 2 2	4 4 4 4 4	4 4 4 4 4
(a)	1 2 3 4 5	(b) 3 3 3 3 3	(c) 3 3 3 3 3	(d) 3 3 3 3 3
	1 2 3 4 5	4 4 4 4 4	2 2 2 2 2	2 2 2 2 2
	1 2 3 4 5	5 5 5 5 5	1 1 1 1 1	1 1 1 1 1

9. Fie un vector cu 100 de elemente de tipul *char*. Fiecare element reprezintă o literă din alfabetul latin (de la A la Z). Pentru fiecare element de pe poziția *i* din vector se calculează o valoare  $S_i$  care reprezintă câte dintre literele de pe pozițiile de la 0 la  $i - 1$  sunt înainte în alfabet.

De exemplu, pentru vectorul *S, C, A, U, N*, valoarea  $S_4$  este 2 deoarece doar literele *C* și *A* sunt înaintea literei *N* în alfabet.

Care dintre funcțiile de mai jos returnează corect suma tuturor valorilor  $S_i$ , cu *i* de la 0 la 99?

(a)	(b)
<pre> <b>int</b> f(<b>int</b> v[100]) {     <b>int</b> sum = 0;     <b>int</b> i, j;     <b>for</b> (i = 1; i &lt; 100; i = i + 1)         <b>for</b> (j = 0; j &lt; i; j = j + 1)             <b>if</b> (v[j] - 'A' &gt; v[i] - 'A')                 sum = sum + 1;     <b>return</b> sum; } </pre>	<pre> <b>int</b> f(<b>int</b> v[100]) {     <b>int</b> sum = 0;     <b>int</b> i, j;     <b>for</b> (i = 0; i &lt; 100; i = i + 1)         <b>for</b> (j = 0; j &lt; i; j = j + 1)             <b>if</b> (v[j] &lt; v[i])                 sum = sum + 1;     <b>return</b> sum; } </pre>
(c)	(d)
<pre> <b>int</b> f(<b>int</b> v[100]) {     <b>int</b> sum = 0;     <b>int</b> i, j;     <b>for</b> (i = 1; i &lt; 100; i = i + 1)         <b>for</b> (j = i+1; j &lt; 100; j = j + 1)             <b>if</b> (v[j] - 'A' &lt; v[i] - 'A')                 sum = sum + 1;     <b>return</b> sum; } </pre>	<pre> <b>int</b> f(<b>int</b> v[100]) {     <b>int</b> sum = 0;     <b>int</b> i, j;     <b>for</b> (i = 1; i &lt; 100; i = i + 1)         <b>for</b> (j = 0; j &lt; i; j = j + 1)             <b>if</b> (v[j] &lt; v[i])                 sum = sum + v[i] - v[j];     <b>return</b> sum; } </pre>

10. Se consideră o matrice *M* cu *n* linii și *m* coloane care se completează pe linii cu termenii șirului lui Fibonacci. Pentru o astfel de matrice cu  $m=n=3$ , care va fi valoarea variabilei suma din secvența de cod de mai jos?

```

int suma=0;
for (i = 0; i<m; i++)
{
    suma += M[0][i];
    suma += M[n - 1][i];
}
for (i = 1; i<n - 1; i++)
{
    suma += M[i][0];
    suma += M[i][m - 1];
}

```

- (a) 88
- (b) 83
- (c) 34
- (d) 39

11. Fie următoarea funcție ce are ca parametri de intrare un vector de numere întregi și un număr natural nenul:

```

funcția  $f(v, n)$ 
┌
│  $h \leftarrow n$ 
│  $sum \leftarrow 0$ 
│ cât timp  $h > 0$  execută
│ ┌
│ │ pentru  $i \leftarrow 0$  la  $n - 1$  execută
│ │ ┌
│ │ │  $sum \leftarrow sum + v[i]$ 
│ │ │  $h \leftarrow h/2$ 
│ │ └
│ └
└ returnează  $sum$ 

```

Dacă se apelează funcția cu un vector  $v$  de 1024 de elemente, toate având valoarea 1, care va fi valoarea returnată de aceasta?

- (a) 1024
- (b) 1048576
- (c) 10240
- (d) 11264

12. Care este complexitatea din punctul de vedere al duratei de execuție (complexitatea timp) a funcției de mai jos?

```

int  $f(\mathbf{int} \ n)$  {
  int  $i, j, x = 0;$ 
  for ( $i = 1; i \leq n; i = i + 1$ )
  {
    for ( $j = 1; j < n; j = j + i$ )
    {
       $x = x + 1;$ 
    }
  }
  return  $n;$ 
}

```

- (a)  $\mathcal{O}(n^2)$
- (b)  $\mathcal{O}(n \cdot \log^2 n)$
- (c)  $\mathcal{O}(n \cdot \log n)$
- (d)  $\mathcal{O}(2 \cdot n)$

13. Considerând că toate declarațiile și inițializările sunt corecte, care implementare a sortării crescătoare a unui vector  $a$  de dimensiune  $n$  nu este corectă?

```

V1
do {
    k = 0;
    for (i = 0; i < n - 1; i++)
    {
        if (a[i] > a[i + 1])
        {
            k++; t = a[i];
            a[i] = a[i + 1];
            a[i + 1] = t;
        }
    }
} while(k);

```

```

V2
m = n;
while (m > 0)
{
    for (i = 0; i < m - 1; i++)
    {
        if (a[i] > a[i + 1])
        {
            t = a[i];
            a[i] = a[i + 1];
            a[i + 1] = t;
        }
    }
    m--;
}

```

```

V3
for (j = 0; j < n - 1; j++)
    for (i = 0; i < n; i++)
        if (a[i] > a[i + 1])
        {
            t = a[i];
            a[i] = a[i + 1]; a[i + 1] = t;
        }

```

- (a) Toate variantele sunt corecte (b) V1 (c) V3 (d) V2

14. Secvența de pseudocod de mai jos ar trebui să implementeze algoritmul lui Euclid. A și B sunt numere naturale.

```

funcția Euclid(A, B)
    cât timp V1 execută
    |   C ← A%B
    |   A ← B
    |   B ← C
    returnează V2;

```

Expresiile corecte pentru secvențele V1 și V2 sunt:

- (a)  $V1 : B \neq 0; V2 : A$  (b)  $V1 : B \neq 0; V2 : C$  (c)  $V1 : A \neq B; V2 : A$  (d)  $V1 : A \neq 0; V2 : B$

15. Un instructor de schi are 5 elevi cu înălțimile de 1.80 m, 1.70 m, 1.64 m, 1.85 m și 1.89 m. Schiurile de care dispune au lungimile următoare: 1.70 m, 1.75 m, 1.80 m, 1.80 m și 1.90 m. Instructorul vrea să distribuie schiurile astfel încât suma diferențelor absolute (în modul) dintre înălțimea fiecărui elev și lungimea schiurilor atribuite să fie minimă. După ce a analizat cu atenție problema, acesta a început distribuția schiurilor. Elevul cu înălțimea de 1.80 m a primit schiurile de 1.80 m. Ce schiuri va primi elevul cu înălțimea de 1.70 m?

- (a) 1.70 m (b) 1.75 m (c) 1.80 m (d) 1.90 m

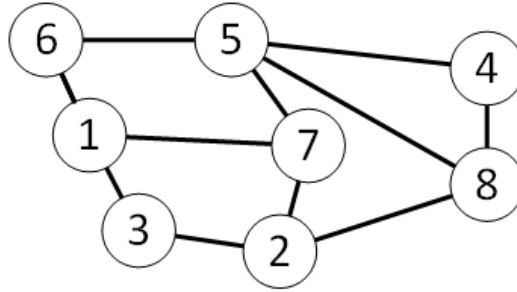
16. Câte posibilități există pentru ca, prin permutarea literelor *PROGRAM*, litera *O* să fie pe penultima poziție?

- (a) 240 (b) 120 (c) 720 (d) 360

17. Fie  $G$  un graf neorientat cu  $n$  noduri. Dacă  $G$  este complet, câte lanțuri hamiltoniene conține?

- (a)  $n^2$  (b)  $n!$  (c)  $n * (n - 1) / 2$  (d)  $n$

18. Fie următorul graf neorientat.



Pentru câte dintre perechile de noduri  $(i, j)$ , cu  $i < j$ , lanțul minim de la nodul  $i$  la nodul  $j$  are lungimea 3?

- (a) 5 (b) 4 (c) 0 (d) 6

19. Fie  $G$  un graf neorientat cu 10 noduri. Dacă  $G$  este format din 3 componente conexe, care este numărul maxim de muchii pe care îl poate conține  $G$ ?

- (a) 28 (b) 12 (c) 32 (d) 40

20. Pe o placă de dezvoltare hardware trebuie să se facă  $n$  găuri. Se consideră graful  $G = (V, E)$ , unde  $V$  este mulțimea celor  $n$  găuri și  $E$  este mulțimea tuturor perechilor de găuri. Se cunosc distanțele dintre găuri. Care este traseul cel mai potrivit pentru a minimiza suma tuturor deplasărilor burghiului de găurit?

- (a) circuit eulerian (b) circuit hamiltonian de lungime minimă (c) circuit elementar (d) problema nu are soluție

21. Se consideră un vector  $V$  ce conține nodurile unui arbore binar cu următoarele proprietăți:  $V[1]$  conține rădăcina,  $V[0]$  este neutilizat, vectorul nu conține poziții libere iar valoarea rădăcinii fiecărui subarbore este mai mare decât valorile tuturor descendenților. Dacă părintele este pe poziția  $i$ , fiu săi pot fi pe pozițiile  $2*i$  și  $2*i+1$ . Într-un astfel de arbore cu  $N$  noduri se inserează valoarea  $a$  cu următorul algoritm descris în pseudocod, păstrând proprietățile enunțate.

```

fiu, parinte si aux sunt indecsi in  $V$   $N \leftarrow N + 1$ ;
 $V[N] \leftarrow a$ ;
 $fiu \leftarrow N$ ;
 $parinte \leftarrow N/2$ ;

```

**cât timp**  $parinte \geq 1$  **execută**

```

    dacă  $V[parinte] < V[fiu]$  atunci
         $aux \leftarrow V[parinte]$ ;
         $V[parinte] \leftarrow V[fiu]$ ;
         $V[fiu] \leftarrow aux$ ;
         $E1$ ;
         $parinte \leftarrow parinte/2$ ;
    altfel
         $E2$ ;

```

Care sunt expresiile corecte pentru  $E1$  și  $E2$ ?

- (a)  $fiu \leftarrow parinte * 2$ ;  $E2 : fiu \leftarrow parinte$   
 (b)  $fiu \leftarrow parinte/2$ ;  $E2 : parinte \leftarrow 0$   
 (c)  $fiu \leftarrow parinte$ ;  $E2 : parinte \leftarrow 0$   
 (d)  $fiu \leftarrow parinte$ ;  $E2 : parinte \leftarrow fiu$

22. Fie următoarele declarații:

```
struct PUNCT {  
    float x, y;  
};
```

```
struct CERC {  
    struct PUNCT centru;  
    float raza;  
};
```

Fie variabila  $C1$  asociată unui cerc din planul  $xOy$ . Cercul este de rază 2 și centrul cercului este în punctul de coordonate  $(-1, -1)$ . Definiția variabilei  $C1$  este:

```
struct CERC C1;
```

Care dintre următoarele secvențe de instrucțiuni trebuie folosită pentru a încărca corect valorile pentru rază și centrul cercului în variabila  $C1$ ?

- (a)  $C1.raza = 2; C1.centru.x = -1; C1.centru.y = -1;$
- (b)  $CERC.raza = 2; CERC.PUNCT.x = -1; CERC.PUNCT.y = -1;$
- (c)  $C1.CERC.raza = 2; C1.PUNCT.x = -1; C1.PUNCT.y = -1;$
- (d)  $C1.raza = 2; C1.x = -1; C1.y = -1;$

23. Se consideră următoarea funcție recursivă, definită incomplet:

```
int F(long x, int d)  
{  
    if (...)  
        return d;  
    else  
        return F(x, d-1);  
}
```

Cu ce expresie trebuie înlocuite punctele de suspensie astfel încât, în urma apelului  $F(x,99)$ , funcția să returneze cel mai mare divizor de cel mult două cifre al numărului natural transmis prin parametrul  $x$ .

- (a)  $x \% d < 100$
- (b)  $x \% d == 1$
- (c)  $x \% d == 0$
- (d)  $x / 10 == 0 \ \&\& \ d < 100$

24. Se consideră următoarele funcții:

```
int f2(int x);  
int f1(int x){  
    if(x<=1) return 1;  
    return x*f2(x-1);  
}  
int f2(int x){  
    if(x<=1) return 1;  
    return f1(x-1);  
}
```

Care dintre următoarele expresii este egală cu  $x!$ , pentru orice  $x$  număr natural pozitiv?

- (a)  $f1(x)$
- (b)  $f1(x) \cdot f2(x)$
- (c)  $f2(x)$
- (d)  $x \cdot f2(x)$

25. Știind că înainte de apel variabila  $a$  are valoarea 1 și variabila  $b$  valoarea 0, ce valoare vor avea variabilele  $a$  și  $b$  după apelul funcției  $f(102304, a, b)$  ?



```

void functie(unsigned long n, unsigned long &a, unsigned long &b)
{
    if (n<10)
        if (n%10==0)
            a=n;
        else
            b=n;
    else {
        functie(n/10,a,b);
        if (n%10==0)
            a=a*10+n%10;
        else
            b=b*10+n%10;
    }
}

```

(a) a = 0, b = 102304;    (b) a = 100, b = 1234;    (c) a = 1, b=1234;    (d) a =1234, b=102304.