

Concurs Mate-Info – 21 martie 2026
Proba scrisă la Informatică

NOTĂ IMPORTANTĂ:

În lipsa altor precizări:

- Toate operațiile aritmetice se efectuează pe tipuri de date nelimitate (nu există *overflow* / *underflow*).
- Numerotarea indicilor tuturor vectorilor, matricelor și a șirurilor de caractere începe de la 1.
- Toate restricțiile se referă la valorile parametrilor actuali la momentul apelului inițial.
- O subsecvență a unui vector sau a unui șir de caractere este formată din elemente care ocupă poziții consecutive în vector, respectiv în șirul de caractere.
- Dacă pe același rând apar mai multe instrucțiuni de atribuire consecutive, acestea sunt delimitate prin ";".

1. Fie variabilele $a = 5$, $b = 6$, $c = 7$ și $d = 8$. Notăm cu $|$ operația **OR** pe biți și cu $\&$ operația **AND** pe biți.

A. $(a | b) + (c \& d)$

B. $(a | (b + c)) \& d$

C. $a | (b + (c \& d))$

D. $((a | b) + c) \& d$

Care din următoarele expresii au valoarea 8?

2. Se consideră algoritmul `cevaFace(n)`, unde n este număr natural ($10 \leq n \leq 10^6$):

```
Algorithm cevaFace(n):  
    rez ← 1  
    uc ← n MOD 10  
    While n ≥ 10 execute  
        n ← n DIV 10  
    EndWhile  
    For i ← 1, uc execute  
        rez ← rez * n  
    EndFor  
    Return rez  
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului `cevaFace(12345)` algoritmul returnează 5.
- B. Algoritmul `cevaFace(n)` returnează cifra cea mai puțin semnificativă a numărului n la puterea dată de cifra cea mai semnificativă a sa.
- C. În urma apelului `cevaFace(175439)` algoritmul returnează 1.
- D. Algoritmul `cevaFace(n)` returnează cifra cea mai semnificativă a numărului n la puterea dată de cifra cea mai puțin semnificativă a sa.

3. Se consideră numărul natural n ($1 \leq n \leq 10$).

Care dintre algoritmii $F(n, i, P)$ returnează $n!$ în urma apelului $F(n, 1, 1)$?

A.
Algorithm $F(n, i, P)$:
 If $n = i$ then
 Return P
 EndIf
 Return $F(n, i + 1, P * (i + 1))$
EndAlgorithm

B.
Algorithm $F(n, i, P)$:
 If $n = i$ then
 Return P
 EndIf
 Return $F(n, i, P * i)$
EndAlgorithm

C.
Algorithm $F(n, i, P)$:
 If $i = n$ then
 Return P
 EndIf
 $j \leftarrow i + 1$
 $Q \leftarrow P * i$
 Return $F(n, j, Q)$
EndAlgorithm

D.
Algorithm $F(n, i, P)$:
 If $i = n$ then
 Return P
 EndIf
 $j \leftarrow i + 1$
 $Q \leftarrow P * j$
 Return $F(n, j, Q)$
EndAlgorithm

4. Se consideră algoritmul $f(A, n)$, unde n este număr natural ($1 \leq n \leq 10^3$), iar A este un vector cu n elemente numere întregi ($0 \leq A[i] \leq 10^5, 1 \leq i \leq n$).

```
Algorithm f(A, n):  
  If n = 0 then  
    Return 0  
  EndIf  
  Return f(A, n - 1) + (A[n] MOD 2) * (A[n] DIV 10 MOD 2)  
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $f([23, 213, 123, 53, 97, 546, 768], 7)$, algoritmul returnează 3.
- B. În urma apelului $f([77, 3, 657, 345, 123, 789, 45], 7)$, algoritmul returnează 3.
- C. Algoritmul returnează numărul elementelor din vector care au cel puțin 2 cifre.
- D. În urma apelului $f([11, 31, 5, 37, 9], 5)$, algoritmul returnează 3.

5. O carte de informatică are paginile numerotate de la 1 la n ($100 < n < 500$). Pe paginile care sunt multipli de 3, dar nu și de 4, se găsesc 2 probleme rezolvate. Pe paginile care sunt multipli de 4, dar nu și de 3, sunt 3 probleme rezolvate. Toate celelalte pagini conțin doar teorie.

Care dintre perechile de formule calculează numărul paginilor care conțin doar teorie, notat cu T , și numărul total de probleme rezolvate din carte, notat cu P ?

- A. $T \leftarrow n - (n \text{ DIV } 3 + n \text{ DIV } 4 - n \text{ DIV } 12)$
 $P \leftarrow 2 * (n \text{ DIV } 3 - n \text{ DIV } 12) + 3 * (n \text{ DIV } 4 - n \text{ DIV } 12)$
- B. $T \leftarrow n - ((n \text{ DIV } 3 - n \text{ DIV } 12) + (n \text{ DIV } 4 - n \text{ DIV } 12))$
 $P \leftarrow 2 * (n \text{ DIV } 3 - n \text{ DIV } 12) + 3 * (n \text{ DIV } 4 - n \text{ DIV } 12)$
- C. $T \leftarrow n - (n \text{ DIV } 3 + n \text{ DIV } 4)$
 $P \leftarrow 2 * (n \text{ DIV } 3) + 3 * (n \text{ DIV } 4)$
- D. $T \leftarrow n - (n \text{ DIV } 3 + n \text{ DIV } 4 - n \text{ DIV } 6)$
 $P \leftarrow 2 * (n \text{ DIV } 3 - n \text{ DIV } 6) + 3 * (n \text{ DIV } 4 - n \text{ DIV } 6)$

6. Se consideră algoritmul $f(n)$, unde n este un număr natural ($1 \leq n < 20$).

```
Algorithm f(n):  
  x ← 0  
  For i ← 1, n execute  
    temp ← i  
    p ← 1  
    While temp > 0 execute  
      p ← p * 10  
      temp ← temp DIV 10  
    EndWhile  
    x ← x * p + i  
  EndFor  
  Return x  
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. Pentru valorile lui n divizibile cu 3, algoritmul $f(n)$ returnează o valoare divizibilă cu 3.
- B. Algoritmul returnează întotdeauna un număr format din n cifre.
- C. Nu există n astfel încât valoarea returnată de algoritmul $f(n)$ să fie divizibilă atât cu 3, cât și cu 5.
- D. Valoarea returnată de algoritmul $f(n)$ are aceeași paritate ca și n .

7. Se consideră algoritmul $h(n, k)$, unde n și k sunt numere naturale ($1 \leq n, k \leq 10^5$).

```
Algorithm h(n, k):  
  If n < k then  
    Return n  
  EndIf  
  If n MOD k = 0 then  
    Return k  
  EndIf  
  Return h(n - k, k)  
EndAlgorithm
```

Care este valoarea returnată în urma apelului $h(2000, 21)$?

- A. 2000
- B. 21
- C. 5
- D. În urma apelului $h(2000, 21)$, algoritmul nu își termină execuția.

8. Se consideră algoritmul $f(k)$, unde k este număr natural ($1 \leq k \leq 10^4$).

Algorithm $f(k)$:

```
x ← 0; i ← 1
While i ≤ 2 * k + 1 execute
    x ← x + i
    i ← i + 2
EndWhile
Return x
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $f(10)$, se returnează valoarea 123.
- B. Pentru orice valoare k , se returnează un număr par.
- C. În urma apelului $f(3)$, se returnează valoarea 15.
- D. În urma apelului $f(21)$, se returnează valoarea 484.

9. Care dintre următoarele numere date în baza 10 sunt palindrom dacă le reprezentăm în baza 2?

A. $515_{(10)}$

B. $1967_{(10)}$

C. $1057_{(10)}$

D. $2973_{(10)}$

10. Se consideră algoritmul $s(arr, n, x)$, unde n este număr natural ($1 \leq n \leq 10^3$), arr este un vector cu n elemente numere naturale ($arr[1], arr[2], \dots, arr[n]$, $0 \leq arr[i] \leq 10^3$, pentru $i = 1, 2, \dots, n$), iar x este număr natural ($0 \leq x \leq 10^3$):

Algorithm $s(arr, n, x)$:

```
st ← 1
dr ← n
p ← -1
While st ≤ dr execute
    m ← (st + dr) DIV 2
    If arr[m] = x then
        p ← m
        st ← m + 1
    Else
        If arr[m] < x then
            st ← m + 1
        Else
            dr ← m - 1
        EndIf
    EndIf
EndWhile
Return p
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. Algoritmul $s(arr, n, x)$ are complexitatea timp $O(\log \log n)$.
- B. În urma apelului $s([10, 20, 30, 40, 50, 60, 70], 7, 20)$ algoritmul returnează 2.
- C. În urma apelului $s([10, 20, 30, 40, 50, 60, 70], 7, 35)$ algoritmul returnează -1.
- D. În urma apelului $s([1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 4], 11, 4)$ algoritmul returnează 11.

11. Se consideră algoritmul $eCrescatoare(a, n, p, q)$, unde a este un vector de n numere naturale ($2 \leq n \leq 10^9$, $0 \leq a[i] \leq 10^9$ pentru $1 \leq i \leq n$), iar p și q sunt numere naturale ($1 \leq p \leq q \leq n$).

Algorithm $eCrescatoare(a, n, p, q)$:

```
If p = q then
    Return True
EndIf
If p + 1 = q then
    Return a[p] < a[q]
EndIf
...
```

EndAlgorithm

Care dintre următoarele secvențe de instrucțiuni pot fi inserate în locul punctelor de suspensie astfel încât algoritmul să returneze *True*, dacă subsecvența $a[p], \dots, a[q]$ este strict crescătoare?

- A. Return $eCrescatoare(a, n, p, (p + q) \text{ DIV } 2)$ AND $eCrescatoare(a, n, (p + q) \text{ DIV } 2, q)$
- B. Return $eCrescatoare(a, n, p, (p + q + 1) \text{ DIV } 2)$ AND $eCrescatoare(a, n, (p + q + 1) \text{ DIV } 2, q)$
- C. If $p + 2 = q$ then
Return $a[p] < a[p + 1]$ AND $a[p + 1] < a[p + 2]$
EndIf
Return $eCrescatoare(a, n, p, ((p + q) \text{ DIV } 2) + 1)$ AND $eCrescatoare(a, n, ((p + q) \text{ DIV } 2) + 1, q)$
- D. If $p + 2 = q$ then
Return $a[p] < a[p + 1]$ AND $a[p + 1] < a[p + 2]$
EndIf
Return $eCrescatoare(a, n, p, (p + q) \text{ DIV } 2)$ AND $eCrescatoare(a, n, ((p + q) \text{ DIV } 2) + 1, q)$

12. Se consideră algoritmul $\text{altceva}(n, k)$, unde n și k sunt numere naturale ($1 \leq n \leq 10^4$, $0 \leq k \leq 10^4$):

Algorithm $\text{altceva}(n, k)$:

```

If  $n < k$  then
    Return False
EndIf
If  $n = k$  then
    Return True
EndIf
Return  $\text{altceva}(n - k, k + 2)$ 
EndAlgorithm

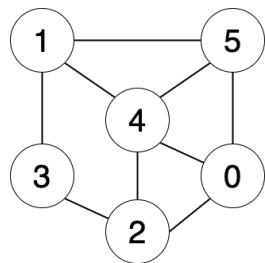
```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $\text{altceva}(256, 0)$ algoritmul va returna *True*.
- B. Dacă numărul n se poate scrie ca produs de două numere consecutive și $k = 0$, în urma apelului $\text{altceva}(n, k)$ se va returna *True*.
- C. Dacă n este număr impar, în urma apelului $\text{altceva}(n, 0)$ se va returna *False*.
- D. În urma apelului $\text{altceva}(272, 0)$ și a apelului $\text{altceva}(380, 0)$ se va returna *True*.

13. Fie următorul graf:

Care dintre următoarele parcurgeri în lățime și în adâncime sunt corecte, dacă se pornește din nodul 3?



- | | | | |
|----|-------------------------------------|----|-------------------------------------|
| A. | Parcursere în lățime: 3 2 4 1 0 5 | B. | Parcursere în lățime: 3 1 4 2 5 0 |
| | Parcursere în adâncime: 3 2 4 1 0 5 | | Parcursere în adâncime: 3 2 4 1 5 0 |
| C. | Parcursere în lățime: 3 2 1 4 0 5 | D. | Parcursere în lățime: 3 1 2 4 5 0 |
| | Parcursere în adâncime: 3 1 5 2 4 0 | | Parcursere în adâncime: 3 1 5 0 4 2 |

14. Se consideră algoritmul $\text{fill}(\text{mat}, n, \text{first}, d)$, unde n este număr natural ($1 \leq n \leq 200$), mat este o matrice cu n linii și n coloane ($\text{mat}[1][1], \text{mat}[1][2], \dots, \text{mat}[1][n], \dots, \text{mat}[n][1], \dots, \text{mat}[n][n]$), iar first și d sunt două numere naturale ($0 \leq \text{first}, d \leq 200$).

Algorithm $\text{fill}(\text{mat}, n, \text{first}, d)$:

```

current  $\leftarrow$  first
For  $i \leftarrow 1, n$  execute
    If  $i \bmod 2 = 0$  then
        For  $j \leftarrow 1, n$  execute
             $\text{mat}[i][j] \leftarrow$  current
            current  $\leftarrow$  current +  $d$ 
        EndFor
    Else
        For  $j \leftarrow n, 1, -1$  execute
             $\text{mat}[i][j] \leftarrow$  current
            current  $\leftarrow$  current +  $d$ 
        EndFor
    EndIf
EndFor
Return current
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $\text{fill}(\text{mat}, 10, 1, 1)$, suma elementelor de pe ultima coloană a matricei este 505.
- B. În urma apelului $\text{fill}(\text{mat}, 10, 1, 1)$, suma elementelor de pe ultima linie a matricei este 955.
- C. În urma apelului $\text{fill}(\text{mat}, n, \text{first}, d)$, valoarea returnată este multiplu al numărului d .
- D. În urma apelului $\text{fill}(\text{mat}, 101, 0, 5)$, elementul din colțul din dreapta jos a matricei este 51000.

15. Se consideră algoritmii $\text{Alg1}(n, x)$ și $\text{Alg2}(n, x)$, unde n este număr natural ($3 \leq n \leq 10^4$), iar x este un vector cu n elemente numere întregi ($x[1], x[2], \dots, x[n]$, $-100 \leq x[i] \leq 100$, pentru $i = 1, 2, \dots, n$).

Algorithm $\text{Alg1}(n, x)$:

```

For  $i \leftarrow 1, n - 1$  execute // (+)
    For  $j \leftarrow 1, n - i$  execute
        If  $x[j] > x[j + 1]$  then // (*)
            aux  $\leftarrow$   $x[j]$ 
             $x[j] \leftarrow x[j + 1]$ 
             $x[j + 1] \leftarrow$  aux
        EndIf
    EndFor
EndFor
EndAlgorithm

```

Algorithm $\text{Alg2}(n, x)$:

```

For  $i \leftarrow 2, n$  execute // (+)
    key  $\leftarrow$   $x[i]$ 
     $j \leftarrow i - 1$ 
    While  $j \geq 1$  AND  $x[j] > \text{key}$  execute // (*)
         $x[j + 1] \leftarrow x[j]$ 
         $j \leftarrow j - 1$ 
    EndWhile
     $x[j + 1] \leftarrow \text{key}$ 
EndFor
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Pentru orice vector x sortat crescător, instrucțiunea marcată cu (*) din algoritmul Alg1(n, x) se execută de mai multe ori decât instrucțiunea marcată cu (*) din algoritmul Alg2(n, x).
- B. Există un vector de intrare x de dimensiune 5 astfel încât după 3 iterații ale ciclului marcat cu (+) din Alg1(n, x) vectorul x va fi [2, 1, 3, 7, 8], iar după 3 iterații ale ciclului marcat cu (+) din Alg2(n, x), vectorul x va fi [2, 3, 7, 8, 1].
- C. În algoritmul Alg1(n, x), dacă la o iterație a buclei marcate cu (+) nu are loc nicio interschimbare, atunci vectorul este deja sortat crescător și iterațiile următoare nu vor mai modifica vectorul.
- D. Algoritmul Alg1(n, x) are complexitatea timp $O(n^2)$, iar algoritmul Alg2(n, x) are complexitatea timp $O(n \log n)$.

16. Se consideră algoritmul $f(n, x)$, unde n este număr natural ($1 \leq n \leq 10^3$), iar x este un vector cu n elemente numere întregi ($x[1], x[2], \dots, x[n], -10^3 \leq x[i] \leq 10^3$, pentru $i = 1, 2, \dots, n$):

Algorithm $f(n, x)$:

```

i ← 2
While i < n execute
    If x[i] = x[i - 1] + x[i + 1] then
        For j ← i, n - 1 execute
            x[j] ← x[j + 1]
        EndFor
        n ← n - 1
    Else
        i ← i + 1
    EndIf
EndWhile
Return n
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelurilor $f(5, [2, 5, 3, 6, 1])$ și $f(5, [2, 5, 3, 1, -1])$ algoritmul returnează aceeași valoare.
- B. Dacă vectorul x este sortat crescător, iar k reprezintă indicele valorii maxime din acest vector, valoarea returnată în urma apelului $f(n, x)$ este k .
- C. Pentru $n = 4$, există un vector x care conține valori pare distincte pentru care valoarea returnată în urma apelului $f(n, x)$ este 2.
- D. Dacă x este un vector care conține doar numere impare, iar m este rezultatul obținut în urma apelului $f(n, x)$, atunci n este egal cu m .

17. Se consideră algoritmul $\text{verif}(x, n, y, m)$, unde n, m sunt numere naturale ($1 \leq n, m \leq 200$), x este un vector cu n elemente numere întregi distincte ($x[1], x[2], \dots, x[n], -10^3 \leq x[i] \leq 10^3$, pentru $i = 1, 2, \dots, n$), iar y este un vector cu m elemente numere întregi distincte ($y[1], y[2], \dots, y[m], -10^3 \leq y[i] \leq 10^3$, pentru $i = 1, 2, \dots, m$).

Algorithm $\text{verif}(x, n, y, m)$:

```

ok ← True
i ← 1
While ok AND i ≤ n execute
    j ← 1
    While j ≤ m AND y[j] ≠ x[i] execute
        j ← j + 1
    EndWhile
    If j = m + 1 then
        ok ← False
    EndIf
    i ← i + 1
EndWhile
Return ok
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Dacă lungimea vectorului x este mai mare ca lungimea vectorului y , atunci algoritmul returnează *False*.
- B. Dacă în urma apelului $\text{verif}(x, n, y, m)$ algoritmul returnează *True*, produsul tuturor elementelor din vectorul x are aceeași ultimă cifră ca produsul tuturor elementelor din vectorul y .
- C. În urma apelurilor $\text{verif}([1, 2, 3, 4], 4, [1, 7, 2, 5, 3, 4, 10], 7)$ și $\text{verif}([4, 2, 1, 3], 4, [1, 10, 2, 5, 3, 4, 7], 7)$ algoritmul returnează același rezultat.
- D. Dacă vectorul x este format din toate numerele pare din intervalul $[-20, 100]$, ordonate crescător, iar vectorul y este format din toate numerele divizibile cu 4 din intervalul $[-20, 100]$, ordonate descrescător, în urma apelului $\text{verif}(x, n, y, m)$ algoritmul returnează *True*.

18. Se consideră algoritmul $\text{enigma}(x, n)$, unde n este un număr natural ($1 \leq n \leq 100$), iar x este un vector cu n elemente numere naturale ($x[1], x[2], \dots, x[n]$, $10 \leq x[i] \leq 10^5$, pentru $i = 1, 2, \dots, n$).

```

Algorithm enigma(x, n):
  If n = 0 then
    Return 0
  EndIf
  a ← x[n]
  s ← 0
  While a > 0 execute
    s ← s + (a MOD 10)
    a ← a DIV 10
  EndWhile
  p ← enigma(x, n - 1)
  If s > p then
    Write '*'
  Return s
EndIf
Return p
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Dacă în urma apelului $\text{enigma}(x, n)$ se afișează n caractere '*', atunci vectorul x este sortat crescător.
- B. În urma apelului $\text{enigma}([19, 28, 37, 46], 4)$, algoritmul returnează 10.
- C. Dacă vectorul x este sortat crescător, atunci în urma apelului $\text{enigma}(x, n)$ se vor afișa n caractere '*'.
- D. În urma apelului $\text{enigma}([99, 95, 91, 11], 4)$, algoritmul nu va afișa niciun caracter '*'.

19. Se consideră algoritmul $\text{modifica}(X, n)$, unde n este număr natural ($1 \leq n \leq 100$), iar X este un vector cu n elemente numere naturale ($X[1], X[2], \dots, X[n]$, $10 \leq X[i] \leq 10^5$, pentru $i = 1, 2, \dots, n$):

```

Algorithm modifica(X, n):
  For i ← 1, n execute
    For j ← 1, n - i execute
      a ← X[j]
      While a ≥ 100 execute
        a ← a DIV 10
      EndWhile
      a ← a MOD 10
      b ← X[j + 1]
      While b ≥ 100 execute
        b ← b DIV 10
      EndWhile
      b ← b MOD 10
      If a > b then
        X[j] ← X[j] + X[j + 1]
        X[j + 1] ← X[j] - X[j + 1]
        X[j] ← X[j] - X[j + 1]
      EndIf
    EndFor
  EndFor
  Return X
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $\text{modifica}([51, 24, 65, 756, 38, 82, 13, 22], 8)$ se returnează vectorul $[51, 82, 22, 13, 24, 65, 756, 38]$.
- B. În urma apelului $\text{modifica}([1845, 9076, 3545, 242, 13, 512, 61, 86], 8)$ se returnează vectorul $[9076, 512, 61, 13, 242, 3545, 86, 1845]$.
- C. În urma apelului $\text{modifica}([372, 4932, 42, 5201, 8193, 4672, 195, 88, 57], 9)$ se returnează vectorul $[42, 57, 88, 195, 372, 4932, 4672, 5201, 8193]$.
- D. În urma apelului $\text{modifica}([87, 19, 28, 656, 943, 2061, 1121], 7)$ se returnează vectorul $[2061, 1121, 943, 656, 87, 28, 19]$.

20. Se consideră algoritmul $\text{ceface}(k, n, x)$, unde n și k sunt numere naturale ($1 \leq k \leq n \leq 10^3$), iar x este un vector de n numere naturale ($0 \leq x[1], x[2], \dots, x[n] \leq 10^3$).

```

Algorithm ceface(k, n, x):
  For a ← 1, n execute
    x[k] ← a
    If n - x[k] > 0 then
      ceface(k + 1, n - x[k], x)
    Else
      print(k, x)
    EndIf
  EndFor
EndAlgorithm

```

```

Algorithm print(k, x):
  For i ← 1, k execute
    Write x[i], " "
  EndFor
  Write new line
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului ceface(1, 5, x), pe prima linie se vor afișa doar valori egale.
- B. În urma apelului ceface(1, 5, x), pe a cincea linie se vor afișa valorile 2 și 3.
- C. Algoritmul afișează toate submulțimile cu k elemente a mulțimii $\{1, 2, \dots, n\}$.
- D. În urma apelului ceface(1, n, x) se vor afișa 2^{n-1} linii.

21. Se consideră algoritmul calcul(n), unde n este număr natural ($2 < n < 10$). Algoritmul zerosMatrice(n) returnează o matrice pătratică cu n linii și n coloane, având toate elementele egale cu 0. Algoritmul zerosVector(n) returnează un vector cu n elemente, toate egale cu 0.

Algorithm calcul(n):

```

a ← zerosMatrice(n)
v ← zerosVector(n - 1)
a[1][2] ← 0
a[2][2] ← 1
v[1] ← 1
For i ← 2, n - 1 execute
    v[i] ← 0
    For j ← 1, i - 1 execute
        a[i + 1][j + 1] ← a[i][j] + a[i][j + 1]
        v[i] ← v[i] * 10 + a[i + 1][j + 1]
    EndFor
    a[i + 1][i + 1] ← 1
    v[i] ← v[i] * 10 + a[i + 1][i + 1]
EndFor
Return v
EndAlgorithm

```

Ce valori va conține vectorul returnat în urma apelului calcul(n)?

- A. Puterile de la 0 la $n - 2$ ale numărului 11
- B. Primii n multipli ai numărului 11
- C. Elemente din șirul $2 * x[i] - 1$ ($i > 0$), unde șirul x este definit astfel: $x[1] = 1, x[2] = 6, x[k] = x[k - 1] * 10 + 1$ (pentru $k > 2$)
- D. Elemente din șirul x , unde șirul x este definit astfel: $x[1] = 1, x[k] = x[k - 1] * 11$ (pentru $k > 1$)

22. Fie algoritmul genereaza(n, a, k, p), unde n, k și p sunt numere naturale ($1 \leq n \leq 12, 0 \leq p \leq n, 1 \leq k \leq 10^6$), iar a este un vector format din n numere naturale ($a[1], a[2], \dots, a[n], 0 \leq a[i] \leq 10^4$, pentru $i = 1, 2, \dots, n$).

Algorithm genereaza(n, a, k, p):

```

If p = n then
    Return k - 1
EndIf
r ← k
For i ← 1, n execute
    If r > 0 AND NOT contine(a, p, i) then
        a[p + 1] ← i
        r ← genereaza(n, a, r, p + 1)
    EndIf
EndFor
Return r
EndAlgorithm

```

Algorithm contine(a, n, v):

```

For i ← 1, n execute
    If a[i] = v then
        Return True
    EndIf
EndFor
Return False
EndAlgorithm

```

Care va fi valoarea vectorului a în urma apelului genereaza(6, [0, 0, 0, 0, 0, 0], 482, 0)?

- A. [4, 1, 2, 3, 6, 5]
- B. [1, 1, 3, 2, 3, 2]
- C. [5, 1, 2, 3, 6, 4]
- D. [5, 1, 2, 4, 6, 3]

23. Se consideră algoritmul $f(n, a)$, unde n este număr natural ($2 \leq n \leq 10^6$), iar a este un vector ($a[1], a[2], \dots, a[n]$, $1 \leq a[i] \leq 10^6$ pentru $i = 1, 2, \dots, n$) de n elemente numere naturale. Algoritmul $\text{zerosMat}(n)$ returnează o matrice cu n linii și n coloane, având toate elementele egale cu 0.

Algorithm $f(n, a)$:

```

m ← zerosMat(n)
i ← 1
j ← n
While i ≤ n AND j ≥ 1 execute
    If a[i] < a[j] then
        m[i][j] ← 1
        i ← i + 1
    Else
        m[i][j] ← 2
        j ← j - 1
    EndIf
EndWhile
Return m
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Dacă vectorul a este ordonat crescător, atunci în urma apelului $f(n, a)$ valoarea 2 apare în matrice de n ori.
- B. Dacă vectorul a este ordonat descrescător, atunci în urma apelului $f(n, a)$ valoarea 2 apare în matrice o singură dată.
- C. Dacă vectorul a are toate valorile identice, atunci în urma apelului $f(n, a)$ fiecare rând al matricei va conține cel puțin o valoare nenulă.
- D. În urma apelului $f(5, [1, 1, 4, 6, 9])$, pe ultima linie a matricei se vor afla doar valori egale cu 2.

24. Avem mărgelile de trei culori (pe care le vom nota prin șirurile de caractere "0", "1" și "2") și folosind algoritmul $\text{colier}(n)$, unde n este un număr natural ($4 \leq n \leq 3 \cdot 2^{10}$), dorim să construim un colier de mărgelile de lungime n care să nu conțină două subsecvențe consecutive identice. De exemplu, pentru $n = 4$, nu putem avea "0211" (ar apărea de două ori consecutiv secvența "1"), dar nici "0202" (secvența "02" apare de două ori la rând). O soluție posibilă este "0201". Operatorul + reprezintă concatenarea șirurilor de caractere, iar algoritmul $\text{len}(s)$ returnează lungimea șirului de caractere s .

Algorithm $\text{extinde}(cuv)$:

```

res ← ""
i ← 1
While i ≤ len(cuv) execute
    c ← cuv[i]
    If c = "0" then
        res ← res + "012"
    Else
        If c = "1" then
            res ← res + "02"
        Else
            res ← res + "1"
        EndIf
    EndIf
    i ← i + 1
EndWhile
Return res
EndAlgorithm

```

Algorithm $\text{colier}(n)$:

```

cuv ← "0"
While len(cuv) < n execute
    cuv ← extinde(cuv)
EndWhile
res ← ""
i ← 1
While i ≤ n execute
    res ← res + cuv[i]
    i ← i + 1
EndWhile
Return res
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Algoritmul $\text{colier}(n)$ returnează reprezentarea unui colier conform cerințelor, indiferent de valoarea lui n .
- B. În urma apelului $\text{colier}(8)$, se va returna reprezentarea "01202101".
- C. În timpul apelurilor $\text{colier}(10)$ și $\text{colier}(12)$, algoritmul $\text{extinde}(cuv)$ se va apela de același număr de ori.
- D. Algoritmul $\text{colier}(n)$ returnează reprezentarea unui colier conform cerințelor dacă și numai dacă n este de forma $3 \cdot 2^k$ ($1 \leq k \leq 10$).

UNIVERSITATEA BABEȘ-BOLYAI CLUJ-NAPOCA
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

Concurs Mate-Info – 21 martie 2026
Proba scrisă la INFORMATICĂ

BAREM ȘI REZOLVARE

OFICIU: 10 puncte

1	BD	3.75 puncte
2	CD	3.75 puncte
3	AD	3.75 puncte
4	AD	3.75 puncte
5	B	3.75 puncte
6	AD	3.75 puncte
7	C	3.75 puncte
8	D	3.75 puncte
9	BCD	3.75 puncte
10	BCD	3.75 puncte
11	ABC	3.75 puncte
12	BCD	3.75 puncte
13	D	3.75 puncte
14	AB	3.75 puncte
15	ABC	3.75 puncte
16	CD	3.75 puncte
17	AC	3.75 puncte
18	B	3.75 puncte
19	ABD	3.75 puncte
20	AD	3.75 puncte
21	AD	3.75 puncte
22	C	3.75 puncte
23	AD	3.75 puncte
24	ABC	3.75 puncte