

Concurs de admitere – 17 iulie 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 un același rând apar mai multe instrucțiuni de atribuire consecutive, acestea sunt delimitate prin ";".

1. Se consideră algoritmul $\text{ceface}(n)$, unde n este număr natural ($0 \leq n \leq 10^3$) și operația $\&$ semnifică **AND** pe biți.

```
Algorithm ceface(n):  
  If (n & 1)  $\neq$  0 then  
    Return 0  
  EndIf  
  Return 1  
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. Dacă n este o putere a lui 2, algoritmul returnează 1.
- B. Dacă n este un număr par, algoritmul returnează 1.
- C. Algoritmul returnează 0 dacă și numai dacă n este număr impar.
- D. Dacă $n = 0$, algoritmul returnează 0.

2. Se consideră algoritmul $\text{gramana}(n, x)$, unde n este număr natural ($1 \leq n \leq 10^3$), iar x este un șir de n caractere ($x[1], x[2], \dots, x[n]$, pentru $i = 1, 2, \dots, n$), toate fiind litere mici.

```
Algorithm gramana(n, x):  
  For i  $\leftarrow$  n, 1, -1 execute  
    For j  $\leftarrow$  2, i execute  
      If  $x[j - 1] > x[j]$  then  
        car  $\leftarrow$   $x[j]$   
         $x[j] \leftarrow x[j - 1]$   
         $x[j - 1] \leftarrow$  car  
      EndIf  
    EndFor  
  EndFor  
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. Complexitatea timp a algoritmului este $O(n^2)$.
- B. Algoritmul realizează o permutare circulară a elementelor șirului dat.
- C. Algoritmul realizează sortarea lexicografică a șirului x în ordine crescătoare.
- D. Algoritmul realizează sortarea lexicografică a șirului x în ordine descrescătoare.

3. Fiind dat numărul binar $101100110000_{(2)}$, precizați care dintre următoarele afirmații sunt adevărate.

- A. Numărul este divizibil cu $16_{(10)}$
- B. Numărul este mai mic decât $10000_{(10)}$
- C. Numărul este par.
- D. Numărul este divizibil cu $32_{(10)}$.

4. Se consideră algoritmul $\text{test}(n)$, unde n este număr natural ($1 \leq n \leq 10^3$).

```
Algorithm test(n):  
  c  $\leftarrow$  1  
  While n  $>$  0 execute  
    n  $\leftarrow$  n - 2  
    c  $\leftarrow$  c * 2  
  EndWhile  
  Return c  
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. Valoarea returnată în urma apelului $\text{test}(n)$ este întotdeauna mai mare decât n .
- B. Valoarea returnată în urma apelului $\text{test}(n)$ este întotdeauna mai mică decât valoarea returnată în urma apelului $\text{test}(n + 1)$.
- C. Dacă numerele naturale distincte n și m au aceeași paritate, valorile returnate în urma apelurilor $\text{test}(n)$ și $\text{test}(m)$ sunt diferite.
- D. Algoritmul $\text{test}(n)$ are complexitate timp $O(\log n)$.

5. Se consideră algoritmul $\text{mister}(x, n)$, unde n este un număr natural ($1 \leq n \leq 10^3$), 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 mister(x, n):
  For i ← 1, n - 1 execute
    For j ← i + 1, n execute
      a ← x[i]
      b ← x[j]
      While a > 10 execute
        a ← a DIV 10
      EndWhile
      While b > 10 execute
        b ← b DIV 10
      EndWhile
      If a < b then
        aux ← x[i]
        x[i] ← x[j]
        x[j] ← aux
      EndIf
    EndFor
  EndFor
  Return x
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Algoritmul $\text{mister}(x, n)$ sortează vectorul x descrescător după prima cifră a elementelor sale.
- B. În urma apelului $\text{mister}([345, 21, 98, 412, 67], 5)$ algoritmul returnează vectorul $[98, 67, 412, 345, 21]$.
- C. Algoritmul $\text{mister}(x, n)$ sortează vectorul x crescător după ultima cifră a numerelor.
- D. În urma apelului $\text{mister}([10, 305, 47, 89], 4)$ algoritmul returnează vectorul $[89, 47, 305, 10]$.

6. La un turneu sportiv participă n ($2 \leq n \leq 100$) jucători. Fiecare partidă se joacă între 2 jucători, pierzătorul fiecărei partide fiind eliminat din turneu.

Câte meciuri trebuie jucate pentru a ști cine câștigă turneul?

- A. $n - 1$
- B. n
- C. $n \text{ DIV } 2$
- D. $(n \text{ DIV } 2) + 1$

7. O expresie logică formată din termeni conectați cu operatorul **OR** ($t_1 \text{ OR } t_2 \text{ OR } \dots \text{ OR } t_n$) se evaluează de la stânga la dreapta doar până când se găsește primul termen cu valoarea *True* în caz că există un astfel de termen (în acest caz valoarea expresiei va fi *True*, iar restul termenilor nu se evaluează). Altfel, dacă toți termenii se evaluează la *False*, valoarea expresiei este *False*.

Se consideră algoritmul $\text{test}(n)$, unde n este un număr natural ($1 \leq n \leq 10^5$).

```

Algorithm test(n):
  Write '*'
  If n < 10 then
    Return False
  EndIf
  If n MOD 2 = 0 then
    Return True
  EndIf
  If n MOD 5 = 0 then
    Return True
  EndIf
  Return test(n DIV 2) OR test(n DIV 5) OR test(n DIV 10) //#
EndAlgorithm

```

Presupunând că la execuția rândului marcat cu #, unde se găsește o expresie logică formată din trei termeni conectați cu operatorul **OR** se procedează în modul descris mai sus, câte caractere de '*' se vor afișa în urma apelului $\text{test}(159)$?

- A. 10
- B. 15
- C. 3
- D. 8

8. Se consideră algoritmul $f(n)$, unde n este un număr natural ($1 \leq n \leq 10^3$).

```

Algorithm f(n):
  If n = 1 then
    Return 0
  EndIf
  Return f(n DIV 2) + 1
EndAlgorithm

```

Ce se returnează în urma apelului $f(n)$?

- A. Partea întreagă a valorii $\log_2 n$.
- B. Numărul minim de biți necesari pentru a reprezenta n în baza 2.
- C. Numărul de biți de 1 din reprezentarea lui n în baza 2.
- D. O valoare diferită de n .

9. Se consideră algoritmul $ff(n, v)$, unde n este număr natural ($1 \leq n \leq 10^4$), iar v este un vector cu n elemente numere întregi ($v[1], v[2], \dots, v[n]$, $-100 \leq v[i] \leq 100$, pentru $i = 1, 2, \dots, n$).

```

Algorithm ff(n, v):
    c ← 0
    For j ← 2, n execute
        a ← v[j]
        i ← j - 1
        While i > 0 AND v[i] ≥ a execute
            v[i + 1] ← v[i]
            i ← i - 1
            c ← c + 1
        EndWhile
        v[i + 1] ← a
    EndFor
    Return c
EndAlgorithm

```

Ce valoare se returnează în urma apelului $ff(5, [-4, 1, 7, -8, 1])$?

- A. 4
- B. 2
- C. 5
- D. 8

10. Se consideră algoritmul $f(a, b)$, unde a și b sunt numere naturale ($1 \leq a, b \leq 10^5$).

```

Algorithm f(a, b):
    If b = 0 then
        Return 0
    EndIf
    If b MOD 2 = 0 then
        Return f(a + a, b DIV 2)
    Else
        Return f(a + a, b DIV 2) + a
    EndIf
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Algoritmul returnează produsul $a * b$.
- B. Complexitatea timp a algoritmului este $O(\log b)$.
- C. În urma apelului $f(2, 3)$, algoritmul returnează 8.
- D. Algoritmul returnează valoarea a^b .

11. Se consideră algoritmul $ceFace(a, n, s)$, unde a este un vector de n ($1 \leq n \leq 20$) numere naturale ($a[1], a[2], \dots, a[n]$, $0 \leq a[i] \leq 10^3$ pentru $1 \leq i \leq n$) ordonat crescător, iar s este un număr natural ($0 \leq s \leq 10^3$).

```

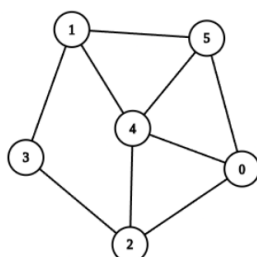
Algorithm ceFace(a, n, s):
    c ← 0
    For i ← 1, n - 2 execute
        j ← i + 1
        k ← n
        While j < k execute
            sum ← a[i] + a[j] + a[k]
            If sum = s then
                c ← c + 1
                j ← j + 1
                k ← k - 1
            Else
                If sum < s then
                    j ← j + 1
                Else
                    k ← k - 1
                EndIf
            EndIf
        EndWhile
    EndFor
    Return c
EndAlgorithm

```

În urma căror apeluri se returnează valoarea 3?

- A. $ceFace([1, 2, 3, 4, 5, 6], 6, 10)$
- B. $ceFace([1, 2, 3, 4, 4, 5, 6], 7, 10)$
- C. $ceFace([1, 2, 3, 4, 4, 5, 6], 7, 8)$
- D. $ceFace([1, 1, 1, 2, 2, 2], 6, 4)$

12. Fie următorul graf:



Care dintre următoarele secvențe de noduri este ciclu Hamiltonian?

- A. 0 1 2 3 4 5 0
- B. 1 3 2 0 4 5 1
- C. 1 3 2 4 2 0 5
- D. 4 2 3 1 5 0 4

13. Prâslea cel Voinic are în grădina sa un pom fermecat cu 27 de mere de aur și 24 de mere de argint. În fiecare zi zmeul fură două mere, însă, în același timp, în pom crește un alt măr: dacă cele două mere luate de zmeu sunt de același fel (ambele de aur sau ambele de argint), în locul lor crește un măr de argint, iar dacă zmeul ia un măr de aur și unul de argint, atunci în locul lor crește un măr de aur.

Care dintre următoarele afirmații sunt adevărate?

- A. În a 50-a zi zmeul a luat un măr de aur și unul de argint.
- B. Nu se poate stabili ce fel de mere a luat zmeul în a 50-a zi.
- C. Există o succesiune de furturi astfel încât, la un moment dat, în pom să existe numai mere de argint.
- D. Pe parcursul procesului descris, numărul total de mere din pom scade cu exact 1 în fiecare zi.

14. Se consideră algoritmul $f(X, n, k)$, unde n și k sunt numere naturale ($1 \leq k \leq n \leq 10$), iar X este un vector de n numere naturale ($X[1], X[2], \dots, X[n]$).

```

Algorithm f(X, n, k):
  If k = n + 1 then
    aux ← 0; ok ← True
    For j ← 1, n execute
      If X[j] = 1 then
        aux ← aux + 1
      EndIf
      If j > 1 AND X[j] = 1 AND X[j - 1] = 1 then
        ok ← False
      EndIf
    EndFor
    If ok AND aux = n DIV 2 then
      For j ← 1, n execute
        Write X[j]
      EndFor
      Write New Line
    EndIf
  Else
    For i ← 0, 1 execute
      X[k] ← i
      f(X, n, k + 1)
    EndFor
  EndIf
EndAlgorithm

```

În urma apelului $f([0, 0, 0, 0], 4, 1)$ prima linie afișată este 0101. Care vor fi următoarele două linii afișate?

- A. 1001
1100
- B. 1010
0110
- C. 0110
1010
- D. 1001
1010

15. Se consideră algoritmul $LU(A, n)$, unde A este o matrice pătratică de dimensiune n ($A[1][1], A[1][2], \dots, A[1][n], \dots, A[n][n], 0 < A[i][j] < 10^3$, pentru $i, j = 1, \dots, n$) care conține numere naturale, iar n este un număr natural ($2 \leq n \leq 10$). Algoritmul $zerosMat(n)$ returnează o matrice pătratică de dimensiune n cu toate elementele egale cu 0.

```

Algorithm LU(A, n):
  L ← zerosMat(n); U ← zerosMat(n)
  For i ← 1, n execute
    L[i][i] ← 1
    For j ← i, n execute
      sum ← 0
      For k ← 1, i - 1 execute
        sum ← sum + L[i][k] * U[k][j]
      EndFor
      U[i][j] ← A[i][j] - sum
    EndFor
    For j ← i + 1, n execute
      sum ← 0
      For k ← 1, i - 1 execute
        sum ← sum + L[j][k] * U[k][i]
      EndFor
      L[j][i] ← (A[j][i] - sum) DIV U[i][i]
    EndFor
  EndFor
  Return L, U
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $LU(A, n)$, matricea L returnată va avea toate elementele deasupra diagonalei principale egale cu 0.
- B. Complexitatea timp a algoritmului este $O(n^2)$
- C. Dacă $A = \begin{pmatrix} 2 & 1 & 2 \\ 4 & 6 & 3 \\ 4 & 2 & 8 \end{pmatrix}$ și $n = 3$, în urma apelului $LU(A, n)$ valoarea matricelor returnate L și U va fi $L = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix}$ și $U = \begin{pmatrix} 2 & 1 & 2 \\ 0 & 4 & -1 \\ 0 & 0 & 4 \end{pmatrix}$.
- D. În urma apelului $LU(\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, 2)$, produsul elementelor de pe diagonala principală a matricei U returnate este egal cu determinantul matricei A .

16. Se consideră algoritmul $f(x, y, v)$, unde x, y și v sunt numere naturale ($1 \leq x, y, v \leq 100, x \leq v$).

```

Algorithm f(x, y, v):
  If x ≥ v then
    Write '1'
  Else
    If x MOD y ≠ 0 then
      f(x + 1, y, v - 1)
    Else
      If y MOD v ≠ 0 then
        f(x, y + 1, v - 1)
        Write '*'
      Else
        f(x + 1, y, v - 2)
        Write '2'
      EndIf
    EndIf
  EndIf
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelurilor $f(3, 5, 10)$, $f(5, 6, 10)$ și $f(2, 5, 9)$ algoritmul afișează același rezultat.
- B. În urma apelului $f(2, 2, 15)$ algoritmul afișează "1*****"
- C. În urma apelului $f(2, 3, 14)$ algoritmul afișează un rezultat care conține cifra 2.
- D. În urma apelurilor $f(2, 2, 11)$, $f(1, 2, 9)$ și $f(3, 7, 9)$ algoritmul afișează același rezultat.

17. Se consideră algoritmul $f(n, x)$ unde n este număr natural ($3 \leq n \leq 10^4$), iar x este un vector de n numere naturale ($x[1], x[2], \dots, x[n], 1 \leq x[i] \leq 10^4$, pentru $i = 1, 2, \dots, n$).

```

Algorithm f(n, x):
  If x[n] - x[n - 1] ≠ x[n - 2] then
    Return 0
  EndIf
  k ← 3
  For i ← n - 1, 3, -1 execute
    If x[i] - x[i - 1] ≠ x[i - 2] then
      Return k
    EndIf
    k ← k + 1
  EndFor
  Return k
EndAlgorithm

```

Care din următoarele afirmații sunt adevărate?

- A. În urma apelului $f(4, [1, 1, 2, 3])$ algoritmul returnează 4.
- B. Algoritmul returnează valoarea n , dacă și numai dacă valorile din vectorul x fac parte dintr-o secvență de numere Fibonacci.
- C. Nu există vector cu 3 elemente pentru care algoritmul să returneze valoarea 1.
- D. Dacă algoritmul returnează valoarea n , atunci valorile din vectorul x sunt ordonate crescător.

18. Se consideră algoritmul $verifica(s1, n1, s2, n2)$, unde $s1$ și $s2$ sunt șiruri de caractere formate doar din litere mici ale alfabetului englez, având lungimile $n1$, respectiv $n2$ ($1 \leq n1, n2 \leq 100$). Algoritmul $zerosVec(n)$ returnează un vector de numere naturale cu n elemente, toate inițializate cu 0.

```

Algorithm verifica(s1, n1, s2, n2):
  If n1 ≠ n2 then
    Return False
  EndIf
  f ← zerosVec(26)
  For i ← 1, n1 execute
    p ← s1[i] - 'a' + 1
    f[p] ← f[p] + 1
  EndFor
  For i ← 1, n2 execute
    p ← s2[i] - 'a' + 1
    ...
    If f[p] < 0 then
      Return False
    EndIf
  EndFor
  Return True
EndAlgorithm

```

Cu care dintre următoarele instrucțiuni pot fi înlocuite punctele de suspensie astfel încât algoritmul $verifica(s1, n1, s2, n2)$ să returneze *True* dacă și numai dacă șirul $s2$ se poate obține prin rearanjarea caracterelor din șirul $s1$?

- A. $f[p] \leftarrow f[p] - 1$
- B. $f[p] \leftarrow f[p] + 1$
- C. $f[p] \leftarrow f[p] - i$
- D. $f[p] \leftarrow (f[p] + 1) \text{ MOD } 2$

19. Se consideră algoritmul `produs_str(n1, n2)`, unde ***n1*** și ***n2*** sunt șiruri de caractere care reprezintă numere naturale nenule (ex. ***n1*** = "123" reprezintă numărul 123). Algoritmul returnează produsul celor două numere sub formă de șir de caractere. De exemplu, `produs_str("2", "3")` returnează "6", iar `produs_str("123", "456")` returnează "56088". Algoritmul `len(sir)` returnează lungimea șirului de caractere ***sir***. Algoritmul `int(car)` returnează cifra corespunzătoare caracterului ***car***. Algoritmul `str(nr)` returnează șirul de caractere corespunzător numărului ***nr*** (ex. `str(123)` returnează "123"). Algoritmul `add(n, m)` returnează suma numerelor ***n*** și ***m***, reprezentate prin șiruri de caractere (ex. `add("123", "457")` returnează numărul 580). Algoritmul `zero(k)` returnează un vector de ***k*** numere, toate egale cu zero. Pentru șiruri de caractere, operatorul + reprezintă operația de concatenare.

Care din următoarele variante reprezintă implementări corecte ale algoritmului `produs_str(n1, n2)`?

A.

```
Algorithm produs_str(n1, n2):
    rez ← ""
    For i ← 1, len(n1) execute
        For j ← 1, len(n2) execute
            produs ← int(n1[i]) * int(n2[j])
            rez ← rez + str(produs)
        EndFor
    EndFor
    Return rez
EndAlgorithm
```

B.

```
Algorithm produs_str(n1, n2):
    rez ← "0"
    For i ← len(n1), 1, -1 execute
        cif ← int(n1[i])
        For j ← 1, cif execute
            rez ← str(add(rez, n2))
        EndFor
    EndFor
    Return rez
EndAlgorithm
```

C.

```
Algorithm produs_str(n1, n2):
    rez ← zero(len(n1) + len(n2))
    For i ← len(n1), 1, -1 execute
        For j ← len(n2), 1, -1 execute
            produs ← int(n1[i]) * int(n2[j])
            poz1 ← i + j - 1
            poz2 ← i + j
            sum ← produs + rez[poz2]
            rez[poz2] ← sum MOD 10
            rez[poz1] ← rez[poz1] + (sum DIV 10)
        EndFor
    EndFor
    rez_str ← ""
    For i ← 1, len(n1) + len(n2) execute
        digit ← rez[i]
        If NOT ((len(rez_str) = 0) AND (digit = 0))
            then
                rez_str ← rez_str + str(digit)
        EndIf
    EndFor
    Return rez_str
EndAlgorithm
```

D.

```
Algorithm produs_str(n1, n2):
    rez ← zero(len(n1) + len(n2))
    For i ← len(n1), 1, -1 execute
        cifTransp ← 0
        For j ← len(n2), 1, -1 execute
            produs ← int(n1[i]) * int(n2[j]) +
                cifTransp + rez[i + j]
            rez[i + j] ← produs MOD 10
            cifTransp ← produs DIV 10
        EndFor
        rez[i] ← rez[i] + cifTransp
    EndFor
    rez_str ← ""
    For i ← 1, len(n1) + len(n2) execute
        digit ← rez[i]
        If NOT ((len(rez_str) = 0) AND (digit = 0))
            then
                rez_str ← rez_str + str(digit)
        EndIf
    EndFor
    Return rez_str
EndAlgorithm
```

20. Se consideră algoritmul `P(s, n)`, unde ***s*** este un șir de caractere având lungimea ***n*** ($3 \leq n \leq 255$). Algoritmul `copiază(s, p, q)` returnează subsecvența șirului de caractere ***s*** începând pe poziția ***p*** și terminând pe poziția ***q*** ($1 \leq p \leq q \leq n$). Operatorul + este operatorul de concatenare pentru șirurile de caractere.

```
Algorithm P(s, n):
    i ← n DIV 3
    j ← 2 * (n DIV 3)
    k ← n
    If n MOD 2 = 0 then
        Return copiază(s, i, i + 1) + copiază(s, j - 1, j + 1) + copiază(s, k, k)
    Else
        Return copiază(s, 1, 1) + copiază(s, i, i + 1) + copiază(s, j - 1, j + 1)
    EndIf
EndAlgorithm
```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $P(\text{"calculator"}, 10)$ algoritmul returnează "lcular".
- B. Șirul de caractere returnat de algoritmul $P(s, n)$ are întotdeauna mai puțin de n caractere.
- C. În urma apelului $P(s1, n1)$ se poate returna șirul de caractere "ubbubb", dacă și numai dacă $n1$ este mai mare sau egal cu 8, și în urma apelului $P(s2, n2)$ se poate returna șirul de caractere "ubbfmi", dacă și numai dacă $n2$ este mai mare sau egal cu 9.
- D. În urma apelului $P(s, n)$ se returnează un șir de caractere format din caractere distincte, dacă și numai dacă n este mai mare sau egal cu 9 și șirul s este alcătuit din caractere distincte.

21. Se consideră algoritmul $f(n, x)$, unde n este număr natural ($1 \leq n \leq 10^3$), x este un vector cu n elemente numere întregi ($x[1], x[2], \dots, x[n]$, $-100 \leq x[i] \leq 100$, pentru $i = 1, 2, \dots, n$), iar $\text{abs}(k)$ returnează valoarea absolută a numărului întreg k .

```

1.  Algorithm f(n, x):
2.      m ← n
3.      a ← 1
4.      b ← -1
5.      For i ← 1, n execute
6.          p ← -1
7.          j ← n
8.          While p = -1 AND j > i execute
9.              If abs(x[i] - x[j]) = 1 then
10.                  p ← j
11.              EndIf
12.              j ← j - 1
13.          EndWhile
14.          t ← i - 1 + n - p
15.          If t < m then
16.              m ← t
17.              a ← i
18.              b ← p
19.          EndIf
20.      EndFor
21.      Return m
22. EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. După executarea instrucțiunii de pe linia 20, atunci $m = n$ sau, dacă $b \neq -1$, $\text{abs}(x[a] - x[b]) = 1$.
- B. Dacă vectorul x conține primele n ($n \geq 3$) elemente ale șirului lui Fibonacci, indiferent de aranjarea lor în vector, se va returna o valoare mai mare decât 0.
- C. Pentru un vector x cu n elemente, și y un vector cu aceleași elemente, permutate circular cu k poziții la dreapta ($1 \leq k < n$) pornind de la vectorul inițial x , $f(n, x)$ și $f(n, y)$ returnează aceeași valoare, indiferent de valoarea lui k .
- D. În urma apelului $f(9, [8, 2, 4, 5, 2, 5, 3, 4, 6])$ algoritmul returnează 6.

22. Se consideră algoritmul $f(v, n, i, a, j)$, unde i, j, n și a sunt numere naturale ($0 \leq j, 1 \leq i, n \leq 20, 1 \leq a \leq 10^4$), v este un vector cu n elemente numere naturale nenule ($v[1], v[2], \dots, v[n]$, $1 \leq v[k] \leq 10^4$, pentru $k = 1, 2, \dots, n$).

```

Algorithm f(v, n, i, a, j):
    If a = 0 AND j MOD 2 = 1 then
        Return 1
    EndIf
    If a = 0 then
        Return 0
    EndIf
    If a < 0 OR i = n + 1 then
        Return 0
    EndIf
    If v[i] MOD 2 = 0 then
        Return f(v, n, i + 1, a, j)
    EndIf
    Return f(v, n, i + 1, a - v[i], j + 1) +
           f(v, n, i + 1, a, j)
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelului $f([2, 4, 6], 3, 1, 11, 0)$ algoritmul returnează 4.
- B. În urma apelului $f([1, 7, 11, 2, 3], 5, 2, 11, 0)$ algoritmul returnează 2.
- C. În urma apelului $f(v, n, 1, 1000, 0)$ algoritmul va returna 0 indiferent de valoarea lui n și de valorile din vectorul v .
- D. În urma apelului $f([1, 2, 3, 4, 5, 6, 3, 7], 8, 1, 7, 0)$ algoritmul returnează 2.

23. Se consideră algoritmul $f(x)$, unde x este un număr natural ($0 \leq x \leq 10^9$).

```

Algorithm f(x):
  If x = 0 then
    Return 0
  EndIf
  If x MOD 2 = 0 then
    Return f(x DIV 2) + ((x + 1) MOD 2)
  EndIf
  Return f(x DIV 2) - (x MOD 2)
EndAlgorithm

```

```

Algorithm f2(x):
  If x = 0 then
    Return 0
  EndIf
  v ← f2(x DIV 2)
  If x MOD 2 = 0 then
    Return v + ((x + 1) MOD 2)
  EndIf
  Return v - (x MOD 2)
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. În urma apelurilor $f(100)$ și $f(24)$ se returnează aceeași valoare.
- B. În urma apelului $f(1024)$ se returnează valoarea 10.
- C. În intervalul $[128, 255]$ există 36 de valori pentru care algoritmul returnează 0.
- D. Pentru orice număr x , $f2(x) = f(x)$; cei doi algoritmi au aceeași complexitate timp.

24. Se consideră algoritmul $nrd(M, r, c, n)$, unde n ($2 \leq n \leq 10$), r și c ($1 \leq r, c \leq n$) sunt numere naturale, iar M este o matrice pătratică cu $n \times n$ elemente ($M[1][1], \dots, M[1][n], \dots, M[n][1], \dots, M[n][n]$, unde $M[i][j] \in \{0, 1\}$, pentru $i, j = 1, \dots, n$). Un drum în matricea M reprezintă o succesiune de poziții vecine pe linie sau coloană, toate având valoarea 1, în care niciuna din poziții nu se repetă. Două drumuri sunt distincte dacă diferă prin cel puțin o poziție. Valorile 0 reprezintă pereți care nu pot fi traversați.

```

Algorithm nrd(M, r, c, n):
  If r = n AND c = n then
    Return 1
  EndIf
  If r < 1 OR r > n OR c < 1 OR c > n OR M[r][c] = 0 then
    Return 0
  EndIf
  total ← 0
  ...
  Return total
EndAlgorithm

```

Dacă în locul punctelor de suspensie se introduce secvența de instrucțiuni dată în varianta de răspuns corespunzătoare, care dintre următoarele afirmații sunt adevărate?

- A.
 $M[r][c] \leftarrow 0$
 $total \leftarrow nrd(M, r + 1, c, n) + nrd(M, r - 1, c, n) + nrd(M, r, c + 1, n) + nrd(M, r, c - 1, n)$
 $M[r][c] \leftarrow 1$
 Algoritmul returnează numărul drumurilor distincte de la poziția $M[r][c]$ până la poziția $M[n][n]$ (în momentul apelului inițial, $M[r][c] = M[n][n] = 1$).
- B.
 $M[r][c] \leftarrow 0$
 $total \leftarrow nrd(M, r + 1, c, n) + nrd(M, r - 1, c, n) + nrd(M, r, c + 1, n) + nrd(M, r, c - 1, n)$
 $M[r][c] \leftarrow 1$
 Complexitatea timp a algoritmului este $O(n^4)$.
- C.
 $total \leftarrow nrd(M, r + 1, c, n) + nrd(M, r - 1, c, n) + nrd(M, r, c + 1, n) + nrd(M, r, c - 1, n)$
 Algoritmul returnează numărul drumurilor distincte de la poziția $M[r][c]$ până la poziția $M[n][n]$ (în momentul apelului inițial, $M[r][c] = M[n][n] = 1$).
- D.
 $total \leftarrow nrd(M, r + 1, c, n) + nrd(M, r, c + 1, n)$
 În urma apelului $nrd(M, 1, 1, 3)$, unde matricea M este formată doar din valori de 1, algoritmul va returna 6.

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

Concurs de admitere – 17 iulie 2026
Proba scrisă la INFORMATICĂ

BAREM ȘI REZOLVARE

OFICIU: 10 puncte

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