

1. Precizați care dintre următoarele expresii are valoarea **ADEVĂRAT** dacă și numai dacă numărul natural n este divizibil cu 6 și are ultima cifră 2 sau 8.

(sursa: concurs/admitere FMI, UBB Cluj-Napoca)

- A. $(n \bmod 6 = 0) \text{ AND } ((n \bmod 10 = 2) \text{ OR } (n \bmod 10 = 8))$
- B. $(n \bmod 9 = 0) \text{ AND } ((n \bmod 10 = 4) \text{ OR } (n \bmod 10 = 8))$
- C. $(n \bmod 3 = 0) \text{ AND } (n \bmod 10 = 2) \text{ AND } (n \bmod 10 = 8)$
- D. $((n \bmod 3 = 0) \text{ AND } (n \bmod 10 = 2)) \text{ OR } ((n \bmod 3 = 0) \text{ AND } (n \bmod 10 = 8))$

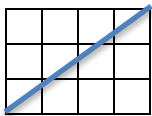
2. Se construiește un dreptunghi de lungime a cm și lățime b cm. Se evidențiază prin paralele cele $a \cdot b$ pătrate de latură 1 cm, ale suprafeței dreptunghiului. Se trasează o diagonală a dreptunghiului care traversează mai multe pătrate ale dreptunghiului.

(sursă: concurs/admitere FMI, UBB Cluj-Napoca).

Exemple: $a=b=2$, diagonală evidențiată traversează 2 pătrate



$a=4$, $b=3$, diagonală evidențiată traversează 6 pătrate



Care afirmații sunt **ADEVĂRATE**?

- A. Pentru $a=3$ și $b=2$ diagonală traversează 3 pătrate.
- B. Pentru $a=6$ și $b=3$ diagonală traversează 6 pătrate.
- C. Pentru $a=60$ și $b=36$ diagonală traversează 86 pătrate.
- D. Pentru $a=60$ și $b=36$ diagonală traversează 84 pătrate.

3. Ce returnează algoritmul **func(m,f,n)**? Vectorul **f** are valorile unei funcții definite pe **{1...m}** cu valori în **{1...n}**.
(pseudocod, sursa: **personală**)

```
Algorithm func (m, f, n):
  If m ≠ n
    Then return false
  EndIf
  If esteFunctie(m,f,n) = false
    then
      return false
  EndIf
  For i← 1, m-1 execute
    For j← i+1, m execute
      If f[i] = f[j]
        then
          return false
        EndIf
      EndFor
    EndFor
  For i← 1,n execute
    If exista (m,f,i) = false
      then
        return false
      EndIf
    EndFor
  return true
EndAlgorithm

Algorithm esteFunctie(m, f, n):
  For i← 1,m execute
    If (f[i] < 1) OR (f[i] > n)
      then
        return false
      EndIf
    EndFor
  return true
EndAlgorithm

Algorithm exista(m, f, val):
  For i← 1,m execute
    If f[i] = val
      then
        return true
      EndIf
    EndFor
  return false
EndAlgorithm
```

- A. true dacă vectorul **f** reprezintă o funcție bijectivă;
- B. false dacă vectorul **f** nu reprezintă o funcție surjectivă;
- C. true dacă vectorul **f** reprezintă o funcție;
- D. false dacă vectorul **f** nu reprezintă o funcție injectivă.

4. Care este rezultatul înmulțirii $121_{(3)} * 121_{(3)}$?

(sursa: personală)

- A. $10111_{(3)}$
- B. $100111_{(3)}$
- C. $100211_{(3)}$
- D. $200111_{(3)}$

5. Ce se afișează la execuția codului următor? (n, m cifre nenule)

(pseudocod, sursa: **personală**)

```
For i ← 1, m execute
    For j ← n, 0, -1 execute
        If i+j > m+n
            Then
                Write i, j newline
            EndIf
        EndFor
    EndFor
```

- A. Se afișează numerele cu suma cifrelor mai mare ca $m+n$;
- B. Se afișează numerele de 2 cifre cu suma cifrelor strict mai mare ca $m+n$;
- C. Nu se afișează niciun număr;
- D. Afirmările de mai sus (**A.**, **B.**, **C.**) sunt false.

6. Ce conține vectorul **v**, după execuția algoritmului **detVect(n)**? (**n** întreg). Algoritmul **detVect(n)** apelează algoritmul **caut(c,v,nr)** cu **v** vector și **nr** lungimea lui **v**.
(pseudocod; sursă: **personală**)

```

Algorithm detVect(n):
  For i ← 1,10 execute
    v[i] ← 0
  EndFor
  nr ← 0
  If n < 0
    then
      n ← -n
  EndIf
  If n = 0
    then
      nr ← nr+1
      v[nr] ← 0
    else
      While n > 0 execute
        p ← 1
        m ← n
        While m > 0 execute
          m ← m DIV 10
          p ← p*10
        EndWhile
        p ← p DIV 10
        a ← n DIV p
        If caut(a,v,nr) = 0
          then
            nr ← nr+1
            v[nr] ← a
          EndIf
        n ← n MOD p
      EndWhile
    EndIf
  EndAlgorithm

```

```

Algorithm caut(c,v,nr):
  For i ← 1,nr execute
    If v[i] = c
      then
        return 1
    EndIf
  EndFor
  return 0
EndAlgorithm

```

- A. Dacă $n = 3025 \Rightarrow nr=4$, iar vectorul $v=[3,0,2,5,0,0,0,0,0,0]$;
 B. Dacă $n = 4561 \Rightarrow nr=4$, iar vectorul $v=[4,5,6,1,0,0,0,0,0,0]$;
 C. Dacă $n = -45616 \Rightarrow nr=4$, iar vectorul $v=[4,5,6,1,0,0,0,0,0,0]$;
 D. Dacă $n = 6060 \Rightarrow nr=1$, iar vectorul $v=[6,0,0,0,0,0,0,0,0,0]$.

7. Ce se returnează după execuția algoritmului **ceFace(n)**, **n** natural nenul?
(pseudocod, sursă: **personală**)

```

Algorithm ceFace(n):
    p ← 1
    m ← 0
    While n > 0 execute
        u ← n MOD 10
        If u MOD 2 = 0
            then
                m ← m + u*p
                p ← p*10
        EndIf
        n ← n DIV 10
    EndWhile
    return m
EndAlgorithm

```

- A. 6** pentru apelul **ceFace(12345)**;
B. 642 pentru apelul **ceFace(123456)**;
C. 1357 pentru apelul **ceFace(1234567)**;
D. 2468 pentru apelul **ceFace(123456789)**.

8. Care afirmații sunt **ADEVĂRATE** după execuția algoritmului **s(n,d)**?, **n** natural, nenul.
(pseudocod; sursă: **personală**).

```

Algorithm s(n,d):
    If n = 1
        then
            return 1
    EndIf
    If d*d = n
        then
            return d
    EndIf
    If d*d > n
        then
            return 0
    EndIf
    If (n MOD d) = 0
        then
            return d + (n DIV d) + s(n,d+1)
        else
            return s(n,d+1)
    EndIf
EndAlgorithm

```

- A. pentru $n \in \{16, 25\}$ și $d=1$** se returnează **31**;
B. pentru $n \in \{14, 15, 23\}$ și $d=1$ se returnează **24**;
C. pentru $n \in \{20, 26, 41\}$ și $d=1$ se returnează **42**;
D. pentru $n \in \{24, 36, 59\}$ și $d=1$ se returneaza **60**.

9. Fie secvența: (pseudocod, sursă: **personală**)

```
Read m,n,x {numere natural nenule, m<n}
p←0
While (m < n) AND (p = 0) execute
  If (m MOD x = 0) AND (n MOD x = 0)
    then
      p← x
    else
      If m MOD x = 0
        then
          n← n-1
        else
          m← m+1
      EndIf
    EndIf
  EndIf
EndWhile
Write m,' ',n
```

Care afirmații sunt **ADEVĂRATE**?

- A. m=30, n=40, x=17, se afișează 18 19
- B. m=10, n=40, x=17, se afișează 18 20
- C. m=11, n=30, x=7, se afișează 14 28
- D. m=14, n=15, x=5, se afișează 15 15

10. Ce returnează algoritmul $s(n,v)$, $n > 1$, natural? $v[1], \dots, v[n]$ un vector cu numere întregi.
(pseudocod, sursa: **personală**)

```
Algorithm prim(n)
...
    {returnează 1, dacă n e prim}
    {returnează 0, dacă n nu e prim}
EndAlgorithm

Algorithm s(n,v):
    s1 ← 0
    s2 ← 0
    For i ← 1, n execute
        If (prim(i)=0) AND (prim(v[i])=1)
            then
                s1 ← s1+v[i]
            else
                s2 ← s2+v[i]
        EndIf
    EndFor
    return s1-s2
EndAlgorithm
```

- A. Diferența dintre suma valorilor prime din vector și suma valorilor neprime;
- B. Diferența dintre suma numerelor prime de pe pozițiile neprime din vector și suma celorlate valori din vector;
- C. Diferența dintre suma numerelor prime de pe pozițiile neprime din vector și suma valorilor neprime de pe pozițiile prime din vector;
- D. Pentru $n=6$, $v = [1, 2, 14, 13, 15, 7]$ se returnează **-12**.

11. Se dă algoritmul **a(m,x,n,y)**. Parametrii **x** și **y** sunt vector, **m** este lungimea lui **x** iar **n** este lungimea lui **y**. Care afirmații sunt **ADEVĂRATE**?
(pseudocod, sursa: **personală**)

```

Algorithm a(m,x,n,y):
  b ← c ← 0
  d ← -1
  For i ← 1, m execute
    For j ← 1, n execute
      k ← 0
      While (i+k ≤ m) AND (j+k ≤ n) AND (x[i+k] = y[j+k]) execute
        k ← k+1      (*)
      EndWhile
      If k > d
        then
          b ← i      (**)
          d ← k
          c ← j
        EndIf
      EndFor
    EndFor
  EndFor

```

- A. Grupul de 3 instrucțiuni care începe la **(**)** se execută de 4 ori pentru
m=8, x= {1, 4, -1, 5, 10, 2, 30, 17}; n=12, y= {-2, -7, 1, 5, -1, 7, 10, 2, 30, 17, 5, 10};
- B. Grupul de 3 instrucțiuni care începe la **(**)** se execută de 3 ori pentru
m=8, x= {1, 4, -1, 5, 10, 2, 30, 17}; n=12, y= {-2, -7, 1, 5, -1, 7, 10, 2, 30, 17, 5, 10};
- C. Instrucțiunea **(*)**, adică **k ← k+1** se execută de 8 ori pentru
m=7, x= {1, 4, -1, 5, 10, 2, 30}; n=10, y= {-2, -7, 1, 5, -1, 7, 10, 2, 30, 17};
- D. Instrucțiunea **(*)**, adică **k ← k+1** se execută de 9 ori pentru
m=7, x= {1, 4, -1, 5, 10, 2, 30}; n=10, y= {-2, -7, 1, 5, -1, 7, 10, 2, 30, 17};

12. Se dă algoritmul **pU(...)** de mai jos. Algoritmul are parametri de intrare un vector **x** și doi indici **st**, **dr**. Apelul lui **pU(...)** (din exterior) se face cu **st=1** și **dr=lungimea lui v**. (pseudocod, sursa: **personală**)

```
Algorithm pU(x, st, dr):  
    i ← st  
    For j ← st, dr execute  
        If x[j] < x[dr]  
            then  
                x[i] ↔ x[j]  
                i ← i+1  
        EndIf  
    EndFor  
    x[i] ↔ x[dr]  
    return i  
EndAlgorithm
```

Care afirmații sunt **ADEVĂRATE**?

- A. Pentru **x = {-10, 10, -10, 10, -10, 10}**, **n=6**, algoritmul **pU** returnează **4**;
- B. Pentru **x = {-14, 13, -12, 11, -10, 9}**, **n=6**, algoritmul **pU** returnează **1**;
- C. Pentru **x = {-9, 10, -11, 12, -13, 14}**, **n=6**, algoritmul **pU** returnează **3**;
- D. Pentru **x = {9, 3, 15, 12, -100, 2}**, **n=6**, algoritmul **pU** returnează **2**.

13. Algoritmul **creV(n)**, crează vectorul **v** cu numere naturale, **n** este dată de intrare.
(pseudocod, sursa: **personală**)

```

Algorithm creV(n):
    v[1] ← 1
    i ← k ← 2                                     ///i indice pentru v
    While i ≤ n execute
        j ← 2
        v[i] ← k
        i ← i+1
        u ← 0
        While (i ≤ n) AND (j ≤ (k DIV 2)) execute
            If (k MOD j) = 0
                then
                    v[i] ← j
                    i ← i+1
                    If u = 0
                        then
                            u ← k DIV j
                        EndIf
                    EndIf
                j ← j+1
            EndWhile
        j ← 2
        While (i ≤ n) AND (j ≤ u) execute
            v[i] ← u
            i ← i+1
            j ← j+1
        EndWhile
        k ← k+1
    EndWhile
EndAlgorithm

```

Care din afirmații sunt **CORECTE**?

- A. $n=12 \Rightarrow v = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$
- B. $n=14 \Rightarrow v = \{1, 1, 2, 1, 3, 1, 2, 4, 1, 5, 1, 2, 3, 6\}$
- C. $n=16 \Rightarrow v = \{1, 2, 3, 4, 2, 5, 6, 2, 3, 3, 7, 8, 2, 4, 4, 4\}$
- D. $n=19 \Rightarrow v = \{1, 2, 3, 4, 2, 2, 5, 6, 2, 3, 3, 3, 7, 8, 2, 4, 4, 4, 4\}$

14. Se dă algoritmul $a(n, x, m, y, k, z)$ cu $n > 0, m > 0, k$ valori naturale iar x, y, z sunt șiruri de numere naturale. Algoritmul $a(\dots)$ apelează algoritmul $e(m, y, val)$ cu $n > 0$ și valori naturale, precum și y vector. Care afirmații sunt **ADEVĂRATE**? (Algoritmul $a(\dots)$ este apelat cu $a(n, x, m, y, 0, z)$) (pseudocod, sursa: **personală**)

```

Algorithm a(n, x, m, y, k, z) :
    If  n > 0
        then
            If e(m, y, x[n]) = 1
                then
                    k ← k+1
                    z[k] = x[n]
                EndIf
            a(n-1, x, m, y, k, z)
        EndIf
    EndAlgorithm

```

```

Algorithm e(m, y, val) :
    For i ← m, 1, -1 execute
        If  val = y[i]
            then
                return 1
        EndIf
    EndFor
    return 0
EndAlgorithm

```

- A. Dacă $x = \{1, 3, 5, 7, 9\}$, $n=5$; $y = \{1, 2, 3, 4, 5\}$, $m=5 \Rightarrow Z = \{1, 2, 3, 4, 5, 5, 7, 9\}$;
 B. Dacă $x = \{3, 7, 9, 5, 1\}$, $n=5$; $y = \{1, 2, 3, 4, 5\}$, $m=5 \Rightarrow Z = \{1, 5, 3\}$;
 C. Dacă $x = \{1, 3, 5, 7, 9\}$, $n=5$; $y = \{1, 2, 3, 4, 5\}$, $m=5 \Rightarrow Z = \{3, 5, 1\}$;
 D. Dacă $x = \{1, 3, 5, 7, 9\}$, $n=5$; $y = \{1, 2, 3, 4, 5\}$, $m=5 \Rightarrow Z = \{5, 3, 1\}$.

15. Cu ajutorul unui algoritm se generează permutările cifrelor pare nenule în ordinea de mai jos.
(pseudocod, sursă: **personală**)

```
1.  2  4  6  8
2.  2  4  8  6
3.  2  6  4  8
4.  2  6  8  4
5.  2  8  4  6
6.  2  8  6  4
7.  4  2  6  8
8.  4  2  8  6
...

```

Care afirmații sunt **FALSE**?

- A. permutarea cu numărul de ordine 15 este: 6 4 8 2;
- B. permutarea cu numărul de ordine 15 este: 6 4 2 8;
- C. există în total 2^4 variante;
- D. există în total $2^3 \cdot 3$ variante.

16. Ce face secvența de instrucțiuni pentru șirul: 15, 24, 35, 25, 75, 26, 0.
(pseudocod, sursă: **concurs/admitere la FMI, UBB Cluj-Napoca**)

```
Read x {x natural}
nr←0
s←0
While x ≠ 0 execute
    nr← nr+1
    If ((nr MOD 2) = 0) AND ((x MOD 5) = 0)
        atunci
            s← s + x MOD 10
    EndIf
    Read x
EndWhile
Write s, " ", nr

```

- A. Afișează 15 7;
- B. Afișează 20 5;
- C. Afișează 5 6;
- D. Afișează suma dintre ultimele cifre ale numerelor de indice par din șirul citit, și numărul de elemente citite.

17. Fie s un șir de numere naturale unde elementele s_i sunt de forma

$$s_i = \begin{cases} x, & \text{dacă } i = 1 \\ x + 1, & \text{dacă } i = 2 \\ s_{i-2} @ s_{i-1} & \text{dacă } i > 2 \end{cases} \quad (i = 1, 2, \dots).$$

Operatorul $@$ concatenează cifrele operandului stâng cu cifrele operandului drept, în această ordine (cifre aferente reprezentării în baza 10), iar x este un număr natural ($1 \leq x \leq 99$). De exemplu, dacă $x = 2$, șirul s va conține valorile 2, 3, 23, 323, 23323, ...

Pentru un număr natural k ($1 \leq k \leq 30$) precizați numărul cifrelor aceluși termen din șirul s care este succesorul termenului format din $k1$ cifre, unde $k1$ este cel mai mare număr cu proprietatea că $k \leq k1 \leq 30$ și există un termen format din $k1$ cifre. (sursă: concurs/admitere la FMI, UBB Cluj-Napoca)

- A. dacă $x = 15$ și $k = 8$, numărul cifrelor termenului căutat este 42.
- B. dacă $x = 2$ și $k = 6$, numărul cifrelor termenului căutat este 55.
- C. dacă $x = 14$ și $k = 27$, numărul cifrelor termenului căutat este 26.
- D. dacă $x = 5$ și $k = 12$, numărul cifrelor termenului căutat este 34.

18. Avem matricea:

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Care din metodele următoare afișează elementele matricii în ordinea următoare (paralele cu diagonala principală, inclusiv diagonala principală)?

(pseudocod, sursă: **personală**)

21, 16,22, 11,17,23, 6,12,18,24, 1,7,13,19,25, 2,8,14,20, 3,9,15, 4,10, 5

A.

```

Algorithm a(n):                                     {n=5 la apel}
  For k ← 1, 3*n execute
    If k ≤ n
      then
        For i ← 1, k execute
          Write a[n-k+i][i] + ','
        EndFor
      else
        For i ← k, 2*n-1 execute
          Write a[i-k+1][i-n+1] + ','
        EndFor
      EndIf
    EndIf
  EndAlgorithm

```

B.

```

Algorithm b(n):                                     {n=5 la apel}
  For k ← 1, 2*n+1 execute
    If k ≤ n
      then
        For i ← 1, k execute
          Write a[n-k+i][i] + ','
        EndFor
      else
        For i ← k, 2*n-1 execute
          Write a[i-k+1][i-n+1] + ','
        EndFor
      EndIf
    EndIf
  EndAlgorithm

```

C.

```

Algorithm c(n):                                     {n=5 la apel}
  For k ← 1, 2*n-1 execute
    If k ≤ n
      then
        For i ← 1, k execute
          Write a[n-k+i][i] + ','
        EndFor
      else
        For i ← k, 2*n-1 execute
          Write a[i-k+1][n-i+1] + ','
        EndFor
      EndIf
  EndAlgorithm

```

D.

```

Algorithm d(n):                                     {n=5 la apel}
  For k ← 1, 2*n-1 execute
    If k ≤ n
      then
        For i ← 1, k execute
          Write a[n-k+i][i] + ','
        EndFor
      else
        For i ← k+1, 2*n-1 execute
          Write a[i-k+1][i-n+1] + ','
        EndFor
      EndIf
  EndAlgorithm

```

19. Se consideră algoritmul **verifica(n, p1, p2)**, unde n, p1 și p2 sunt numere naturale ($1 \leq n, p1, p2 \leq 10^6$):
(sursă: concurs/admitere la FMI, UBB Cluj-Napoca)

```

Algorithm verifica(n, p1, p2):
  bt ← (p1 + p2) DIV 2
  If p1 > p2
    then
      Return False
  EndIf
  If bt * bt = n
    then
      Return True
  EndIf
  If bt * bt > n
    then
      Return verifica(n, p1, bt-1)
  EndIf
  Return verifica(n, bt+1, p2)
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Dacă numerele **p1**, **p2** și **n** sunt prime între ele, atunci apelul **verifica(n, p1, p2)** returnează **True**.
- B. Algoritmul folosește metoda căutării binare și dacă numărul **n** este prim, apelul **verifica(n, 1, n)** returnează **True**.
- C. Pentru apelul **verifica(n, 1, n)** algoritmul returnează **True** dacă și numai dacă numărul **n** este pătrat perfect.
- D. Dacă $p1 \leq n \leq p2$ și în intervalele $[p1, n]$ și $[n, p2]$ există cel puțin câte un pătrat perfect, atunci apelul **verifica(n, p1, p2)** returnează **True**.

20. Se consideră algoritmul **ceFace(n)**, unde **n** este un număr natural ($1 \leq n \leq 3000$).
(sursă: concurs/admitere la FMI, UBB Cluj-Napoca)

```

Algorithm ceFace(n) :
    s ← 0
    i ← 1
    While s < n execute
        s ← s + i
        If s = n
            Then
                Return True
            Else
                i ← i + 2
        EndIf
    EndWhile
    Return False
EndAlgorithm

```

Care dintre următoarele afirmații sunt adevărate?

- A. Dacă **n = 36**, algoritmul returnează **True**.
- B. Dacă **n** este egal cu o sumă de numere impare consecutive începând de la **1**, algoritmul returnează **True**.
- C. Dacă **n** este pătrat perfect, algoritmul returnează **True**, altfel returnează **False**.
- D. Dacă **n = 64**, algoritmul returnează **False**.

21. Se consideră algoritmul **ceFace(a)**, unde **a** este număr natural ($1 \leq a \leq 10^4$).
(sursă: concurs/admitere la FMI, UBB Cluj-Napoca)

```

Algorithm ceFace(a) :
    ok ← 0
    While ok = 0 execute
        b ← a
        c ← 0
        While b ≠ 0 execute
            c ← c * 10 + b MOD 10
            b ← b DIV 10
        EndWhile
        If c = a
            then
                ok ← 1
            Else
                a ← a + 1
        EndIf
    EndWhile
    Return a
EndAlgorithm

```

Care afirmații sunt adevărate?

- A. Algoritmul returnează cel mai mic palindrom mai mare sau egal cu **a**.
- B. Algoritmul returnează cel mai mare palindrom mai mic sau egal cu **a**.
- C. Secvența de cod a instrucțiunii **While imbricate** are complexitatea **O(b)**.
- D. Secvența de cod a instrucțiunii **While imbricate** are complexitatea **O(lg(b))**.

22. Se consideră vectorul de cifre $a = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]$. Cu scopul de a afișa elementele vectorului a într-o altă ordine, se construiește vectorul b (inițial vid). La fiecare pas, se poate alege una din următoarele două operații:

- **Adaugă** – se adaugă primul element din vectorul a la finalul vectorului b și se elimină din vectorul a .
- **Șterge** – se afișează, apoi se șterge ultimul element din vectorul b .

Observații:

- Elementele vectorului a se prelucerează în ordinea dată.
- Nu se poate folosi operația **Adaugă** dacă vectorul a este vid și nu se poate folosi operația **Șterge**, dacă vectorul b este vid.
- Prelucrarea se termină când vectorii a și b sunt vizi.

(sursă: concurs/admitere la FMI, UBB Cluj-Napoca)

Respectând regulile de mai sus, în ce ordine **NU** pot fi afișate cifrele?

- A. 0 1 2 3 4 5 6 7 8 9
- B. 9 8 7 6 5 4 3 2 1 0
- C. 2 3 1 4 5 0 8 9 7 6
- D. 2 4 6 5 3 7 0 1 9 8

23. Se consideră algoritmul **something(n, x)**, unde n este număr natural ($1 \leq n \leq 10^4$), iar x este un vector de n numere naturale ($x[1], x[2], \dots, x[n], 1 \leq x[i] \leq 10^6$, pentru $i = 1, 2, \dots, n$).

(sursă: concurs/admitere la FMI, UBB Cluj-Napoca)

```

Algorithm something(n, x):
    s ← 0
    For i ← 1, n execute
        nr ← 0
        While x[i] > 9 execute
            nr ← nr + 1
            x[i] ← x[i] DIV 10
        EndWhile
        s ← s + nr
    EndFor
    Return s
EndAlgorithm

```

Care afirmații sunt corecte?

- A. Algoritmul returnează suma numărului de cifre ale elementelor vectorului
- B. Algoritmul returnează suma numărului de cifre ale elementelor vectorului – n .
- C. Algoritmul are complexitatea $O(\lg(n))$
- D. Algoritmul are complexitatea $O(n)$.

24. Se consideră algoritmul **expresie(x, y, z)**, unde **x, y, z** sunt numere naturale ($0 \leq x, y, z \leq 10^4$):
(sursă: concurs/admitere la FMI, UBB Cluj-Napoca)

```
Algorithm expresie(x,y,z):  
    If x = 0  
        Then  
            Return z  
        Else  
            Return expresie(x-1,y,x*x+y*y+z)  
        EndIf  
EndAlgorithm
```

Ce returnează algoritmul?

- A. $\sum_{i=1}^x i^2 + \sum_{j=1}^y x * y + \sum_{k=1}^z 1$
- B. $\sum_{i=1}^x i^2 + \sum_{j=1}^y j^2 + \sum_{k=1}^z 1$
- C. $\sum_{i=1}^x i^2 + \sum_{j=1}^y j^2 + \sum_{k=1}^z k$
- D. $\sum_{i=1}^x i^2 + x*y^2 + z$