

Concurs de admitere – 15 septembrie 2022  
Proba scrisă la Informatică

NOTĂ IMPORTANTĂ:

În lipsa altor precizări, presupuneți că toate operațiile aritmetice se efectuează pe tipuri de date nelimitate (nu există *overflow* / *underflow*).

De asemenea, numerotarea indicilor tuturor șirurilor/vectorilor începe de la 1.

1. Se consideră algoritmul `decide(n, x)`, unde  $n$  este număr natural ( $1 \leq n \leq 10000$ ), 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 decide(n, x):
  b ← True
  i ← 1
  While b = True AND i < n execute
    If x[i] < x[i + 1] then
      b ← True
    else
      b ← False
    EndIf
    i ← i + 1
  EndWhile
  return b
EndAlgorithm
```

Pentru care din următoarele situații algoritmul returnează *True*?

- A. Dacă vectorul  $x$  este format din valorile 1, 2, 3, ..., 10.
- B. Dacă vectorul  $x$  este strict crescător.
- C. Dacă vectorul  $x$  nu are elemente negative.
- D. Dacă vectorul  $x$  are elemente pozitive situate înaintea celor negative.

2. Se consideră un număr natural fără cifre egale cu zero, dat prin șirul  $a$  ( $a[1], a[2], \dots, a[n]$ ) în care se află cele  $n$  cifre ale sale ( $1 \leq n \leq 10$  la momentul apelului inițial). Precizați care dintre următorii algoritmi returnează *True* dacă un număr dat sub această formă este palindrom și *False* în caz contrar. Un număr este palindrom dacă citit de la stânga la dreapta are aceeași valoare ca atunci când se citește de la dreapta la stânga.

A.

```
Algorithm palindrom_1(a, n):
  i ← 1
  j ← n
  k ← True
  While (i ≤ j) AND (k = True) execute
    If a[i] = a[j] then
      i ← i + 1
      j ← j - 1
    else
      k ← False
    EndIf
  EndWhile
  return k
EndAlgorithm
```

B.

```
Algorithm translatare(a, n):
  For i = 1, n - 1 execute
    a[i] ← a[i + 1]
  EndFor
EndAlgorithm

Algorithm palindrom_2(a, n):
  j ← n
  If (j = 0) OR (j = 1) then
    return True
  EndIf
  If a[1] = a[j] then
    translatare(a, n)
    return palindrom_2(a, n - 2)
  EndIf
  return False
EndAlgorithm
```

C.

```
Algorithm palindrom_3(a, n):
  i ← n
  j ← 1
  k ← True
  sum1 ← 0
  sum2 ← 0
  While (i > n DIV 2) AND (j ≤ n DIV 2)
    execute
      sum1 ← sum1 + a[i]
      sum2 ← sum2 + a[j]
      i ← i - 1
      j ← j + 1
  EndWhile
  If sum1 = sum2 then
    k ← True
  else
    k ← False
  EndIf
  return k
EndAlgorithm
```

D.

```
Algorithm palindrom_4(a, n):
  i ← 1
  j ← n
  k ← True
  While (i ≤ j) AND (k = True) execute
    If (a[i] = a[j]) AND (i MOD 2 = 0)
      AND (j MOD 2 = 0) then
      i ← i + 1
      j ← j - 1
    else
      k ← False
    EndIf
  EndWhile
  return k
EndAlgorithm
```

3. Se consideră algoritmul  $F(n)$ , unde  $n$  este număr natural ( $1 \leq n \leq 10^9$ ).

```
Algorithm F(n):
  If n < 10 then
    return n
  EndIf
  u ← n MOD 10
  p ← F(n DIV 10)
  If u MOD 5 ≤ p MOD 5 then
    return u
  EndIf
  return p
EndAlgorithm
```

Precizați care dintre următoarele afirmații sunt corecte:

- A. Dacă  $n = 812376$ , valoarea returnată de algoritm este 6.
- B. Dacă  $n = 8237631$ , valoarea returnată de algoritm este 1.
- C. Dacă  $n = 4868$ , valoarea returnată de algoritm este 8.
- D. Dacă  $n = 51$ , valoarea returnată de algoritm este 0.

4. Se consideră algoritmul  $f(n)$ , unde parametrul  $n$  este număr natural ( $1 \leq n \leq 10^9$ ).

```
Algorithm f(n):
  v ← 0; z ← 0;
  For c ← 0, 9 execute
    x ← n
    k ← 0
    While x > 0 execute
      If x MOD 10 = c then
        k ← k + 1
      EndIf
      x ← x DIV 10
    EndWhile
    If k > v then
      v ← k
      z ← c
    EndIf
  EndFor
  return z
EndAlgorithm
```

Care dintre afirmațiile următoare sunt adevărate?

- A. Algoritmul returnează numărul cifrelor numărului  $n$ .
- B. Algoritmul returnează numărul de apariții ale cifrei cu valoarea cea mai mare în numărul  $n$ .
- C. Algoritmul returnează una dintre cifrele cu cel mai mare număr de apariții în numărul  $n$ .
- D. Algoritmul returnează numărul cifrelor având cel mai mare număr de apariții în numărul  $n$ .

5. Care dintre următorii algoritmi afișează reprezentarea binară a numărului natural  $x$  dat ca parametru ( $0 < x \leq 10^9$  la momentul apelului inițial)?

A.

```
Algorithm imp(x):
  If x = 0 then
    r ← x MOD 2
    imp(x DIV 2)
    write r
  EndIf
EndAlgorithm
```

B.

```
Algorithm imp(x):
  If x ≠ 0 then
    r ← x MOD 2
    imp(x DIV 2)
    write r
  EndIf
EndAlgorithm
```

C.

```
Algorithm imp(x):
  If x = 0 then
    r ← x DIV 2
    imp(x DIV 2)
    write r
  EndIf
EndAlgorithm
```

D.

```
Algorithm imp(x):
  If x ≠ 0 then
    r ← x MOD 2
    imp(x)
    write r
  EndIf
EndAlgorithm
```

6. Care dintre următoarele afirmații referitoare la variantele subiectului 5 sunt adevărate?

- A. În timpul execuției algoritmului de la varianta A nu se afișează nimic.
- B. Algoritmul de la varianta B nu se va apela recursiv pentru nicio valoare validă a parametrului  $x$ .
- C. Algoritmul de la varianta C ar fi corect, dacă am schimba "=" cu "≠".
- D. Algoritmul de la varianta D ar fi corect, dacă am schimba "imp(x)" cu "imp(x DIV 2)".

7. Se consideră numerele întregi  $a$  și  $b$  ( $-1000 \leq a, b \leq 1000$ ) și expresia: **NOT** (( $a > 0$ ) **AND** ( $b > 0$ )).

Care dintre următoarele expresii sunt echivalente cu expresia dată mai sus:

- A. (**NOT** ( $a < 0$ )) **AND** (**NOT** ( $b < 0$ ))
- B. ( $a \leq 0$ ) **AND** ( $b \leq 0$ )
- C. (**NOT** ( $a > 0$ )) **OR** (**NOT** ( $b > 0$ ))
- D. **NOT** (( $a > 0$ ) **OR** ( $b < 0$ ))

8. Se consideră algoritmul  $s(n)$ , unde  $n$  este număr natural ( $2 \leq n \leq 10$ ). Operatorul / reprezintă împărțirea reală (ex.  $3 / 2 = 1,5$ ).

```
Algorithm s(n):
  p ← 1
  x ← 0
  For k = 0, n - 1 execute
    p ← p * (k + 1)
    x ← x + 1 / p
  EndFor
  return x
EndAlgorithm
```

Precizați care dintre următoarele sume este returnată de algoritmul dat.

- A.  $\sum_{k=0}^n \frac{1}{k!}$
- B.  $\sum_{k=0}^n \frac{1}{k}$
- C.  $\sum_{k=0}^{n-1} \frac{1}{k!}$
- D.  $\sum_{k=1}^n \frac{1}{k!}$

9. Se consideră algoritmul  $ceFace(n)$ , unde  $n$  este număr natural pozitiv ( $1 \leq n \leq 10000$ ).

**Algorithm**  $ceFace(n)$ :

```
m ← 0
p ← 10
While p < n execute
    r ← n MOD p
    m ← m + r
    p ← p * 10
EndWhile
return m
EndAlgorithm
```

Precizați care dintre următoarele afirmații sunt adevărate:

- A. Pentru  $n = 125$  algoritmul returnează valoarea 521.
- B. Algoritmul  $ceFace(n)$  returnează oglinditul numărului  $n$ .
- C. Pentru  $n = 125$  algoritmul returnează valoarea 155.
- D. Pentru  $n = 340$  algoritmul returnează valoarea 40.

10. Se consideră algoritmul  $f(v, n)$ , unde  $n$  este număr natural nenul ( $1 \leq n \leq 10000$ ) și  $v$  este un vector cu  $n$  numere naturale pozitive ( $v[1], v[2], \dots, v[n]$ ). Presupunem că algoritmul  $prim(d)$  returnează *True* dacă  $d$  (număr natural) este prim și *False* în caz contrar.

**Algorithm**  $f(v, n)$ :

```
x ← 1
a ← 0
For i ← 1, n execute
    For d ← 2, (v[i] DIV 2) execute
        If (prim(d) = True) AND (v[i] MOD d = 0) then
            x ← x * d
        EndIf
    EndFor
EndFor
For d ← 2, (x DIV 2) execute
    If (x MOD d = 0) AND (prim(d) = True) then
        a ← a + 1
    EndIf
EndFor
return a
EndAlgorithm
```

Precizați care dintre următoarele afirmații sunt adevărate:

- A. Algoritmul returnează numărul divizorilor proprii primi distincți ai tuturor numerelor din vectorul  $v$ .
- B. Algoritmul returnează produsul divizorilor primi ai numerelor din vectorul  $v$ .
- C. Algoritmul returnează numărul numerelor prime din vectorul  $v$ .
- D. Algoritmul returnează numărul total al tuturor divizorilor numerelor din vectorul  $v$ .

11. Se consideră algoritmul  $f(n)$ , unde  $n$  este număr natural ( $0 < n \leq 10^9$  la momentul apelului). Variabila locală  $v$  este un vector.

```

Algorithm f(n):
  m ← 0
  While n > 0 execute
    m ← m + 1
    v[m] ← n MOD 10
    n ← n DIV 10
  EndWhile
  x ← 0
  mx ← 0
  While mx > -1 execute
    x ← x * 10 + mx
    mx ← -1
    j ← 1
    For i = 1, m execute
      If v[i] > mx then
        j ← i
        mx ← v[i]
      EndIf
    EndFor
    v[j] ← -1
  EndWhile
  return x
EndAlgorithm

```

Precizați care dintre următoarele afirmații sunt adevărate:

- A. Algoritmul returnează cel mai mare număr care se poate obține folosind cifrele lui  $n$ .
- B. Algoritmul returnează cea mai mare putere a lui 10 care divide numărul  $n$ .
- C. Algoritmul returnează prima cifră din stânga a numărului  $n$ .
- D. Algoritmul returnează suma cifrelor numărului  $n$ .

12. Se consideră algoritmul  $f(n)$ , unde parametrul  $n$  este număr natural ( $1 \leq n \leq 1000^2$  la momentul apelului).

```

Algorithm f(n):
  z ← 0; p ← 1;
  While n ≠ 0 execute
    c ← n MOD 10
    n ← n DIV 10
    If c MOD 3 = 0 then
      z ← z + p * (9 - c)
      p ← p * 10
    EndIf
  EndWhile
  return z
EndAlgorithm

```

Care este valoarea returnată dacă algoritmul se apelează pentru  $n = 103456$ ?

- A. 639
- B. 963
- C. 693
- D. 369

13. Se consideră algoritmul  $f(n)$  dat în enunțul subiectului 12, dar acum parametrul  $n$  este număr natural cu două cifre ( $10 \leq n \leq 99$  la momentul apelului).

Care dintre următoarele variante conțin doar numere pentru care algoritmul returnează valoarea 3?

- A. 61, 65, 67
- B. 62, 66, 68
- C. 16, 56, 76
- D. 26, 66, 86

14. Se dă algoritmul  $\text{ceFace}(a, b)$ , unde  $a$  și  $b$  sunt numere naturale pozitive ( $1 \leq a, b \leq 10000$ ).

```
Algorithm ceFace(a, b):
  For i ← 2, a, 2 execute
    If a MOD i = 0 then
      If b MOD i = 0 then
        Write i
        Write new line
      EndIf
    EndIf
  EndFor
EndAlgorithm
```

Dacă  $a = 600$ , precizați pentru care valori ale lui  $b$  se afișează 4 numere în urma executării algoritmului  $\text{ceFace}(a, b)$ :

- A.  $b = 20$                       B.  $b = 50$                       C.  $b = 12$                       D.  $b = 90$

15. Considerând algoritmul de la subiectul 14, precizați care dintre următoarele afirmații sunt adevărate:

- A. Algoritmul afișează divizorii comuni ai numerelor  $a$  și  $b$ .  
B. Algoritmul afișează divizorii proprii comuni ai numerelor  $a$  și  $b$ .  
C. Algoritmul afișează divizorii impari comuni ai numerelor  $a$  și  $b$ .  
D. Algoritmul afișează divizorii pari comuni ai numerelor  $a$  și  $b$ .

16. Fie un program care generează, în ordine crescătoare, toate numerele naturale de exact 5 cifre distincte care se pot forma cu cifrele 2, 3, 4, 5, 6.

Precizați numărul generat imediat înainte și numărul generat imediat după secvența următoare:  
34256, 34265, 34526, 34562.

- A. 32645 și 34625              B. 32654 și 34655              C. 32654 și 34625              D. 32645 și 34655

17. Fie șirul  $x = (1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 7, \dots)$ , care se continuă conform regulii care se poate observa din elementele enumerate.

Considerând că primul element din șir este pe poziția 1, în care dintre următoarele subsecvențe va apărea doar valoarea 11?

- A.  $x[100], \dots, x[109]$       B.  $x[113], \dots, x[120]$       C.  $x[140], \dots, x[152]$       D.  $x[123], \dots, x[132]$

18. Câte din primele 100 de elemente ale șirului  $x$  descris în subiectul 17 sunt numere prime?

- A. 4                                  B. 34                                  C. 36                                  D. 30

19. Se consideră numerele naturale  $a$  și  $n$  ( $1 \leq a, n \leq 1000$ ), vectorul  $V$  cu  $n$  elemente numere naturale ( $V[1], V[2], \dots, V[n]$ ) și algoritmi  $\text{one}(a, n, V)$  și  $\text{two}(a, n, V)$ :

```
Algorithm one(a, n, V):
  p ← 1; i ← 1;
  While (i ≤ n) AND (a > V[p]) execute
    p ← p + 1
    i ← i + 1
  EndWhile
  return p
EndAlgorithm
```

```
Algorithm two(a, n, V):
  p ← 1; i ← 1;
  While i ≤ n execute
    If a > V[i] then
      p ← p + 1
    EndIf
    i ← i + 1
  EndWhile
  return p
EndAlgorithm
```

Ce proprietate poate avea vectorul  $V$ , astfel încât, pentru orice  $n$  și  $V$  cu proprietatea dată, cei doi algoritmi să returneze valori egale pentru orice valoare a lui  $a$ ?

- A. În vectorul  $V$  toate elementele sunt egale.
- B. În vectorul  $V$  toate elementele sunt distincte și sortate crescător.
- C. În vectorul  $V$  toate elementele sunt distincte și sortate descrescător.
- D. În vectorul  $V$  elementele sunt sortate crescător, dar nu sunt neapărat distincte.

20. Se consideră algoritmul  $\text{suma}(n)$  unde  $n$  este număr natural ( $0 < n \leq 10000$  la momentul apelului inițial).

```

Algorithm suma(n):
  If n = 0 then
    return 0
  else
    return suma(n - 1) + n DIV (n + 1) + (n + 1) DIV n
  EndIf
EndAlgorithm

```

Care dintre afirmațiile de mai jos NU sunt adevărate?

- A. Algoritmul returnează valoarea  $n + 1$
- B. Algoritmul calculează și returnează suma divizorilor proprii ai lui  $n$
- C. Apelul  $\text{suma}(1)$  returnează 2
- D. Algoritmul calculează și returnează dublul părții întregi a mediei aritmetice a primelor  $n$  numere naturale

21. Fie următorul algoritm, având ca parametri de intrare numerele naturale  $a$  și  $b$  ( $0 \leq a, b \leq 10000$  la momentul apelului inițial):

```

Algorithm ceFace(a, b):
  While a * b ≠ 0 execute
    If a > b then
      return ceFace(a MOD b, b)
    else
      return ceFace(a, b MOD a)
    EndIf
  EndWhile
  return a + b
EndAlgorithm

```

Precizați care dintre următoarele afirmații sunt adevărate:

- A. Algoritmul returnează suma numerelor  $a$  și  $b$ .
- B. Algoritmul returnează numărul nenul  $x$  în urma apelului  $\text{ceFace}(x, 0)$  sau  $\text{ceFace}(0, x)$ , respectiv 0 pentru  $\text{ceFace}(0, 0)$ .
- C. Algoritmul returnează cel mai mare divizor comun al numerelor  $a$  și  $b$ .
- D. Algoritmul returnează  $a$  la puterea  $b$ .

22. Se consideră algoritmul  $\text{afișare}(n)$  unde  $n$  este număr natural ( $1 \leq n \leq 10^9$ ):

```

Algorithm afișare(n):
  For i = 1, n - 1 execute
    For j = i + 1, n execute
      If (j - i) < (n DIV 2) then
        Write i, " ", j - i
        Write new line
      else
        If (j - i) ≠ (n DIV 2) then
          Write j - i, " ", i
          Write new line
        EndIf
      EndIf
    EndFor
  EndFor
EndAlgorithm

```

Câte perechi de numere se vor afișa în urma execuției algoritmului pentru  $n = 7$ ?

- A. 21
- B. 15
- C. 11
- D. 17

23. Considerând secvența de cod de mai jos, determinați de câte ori se afișează șirul de caractere UBB, știind că  $n = 3^k$ , unde  $k$  este număr natural ( $1 \leq k \leq 30$ )?

```

j ← n
While j > 1 execute
  i ← 1
  While i ≤ n execute
    i ← 3 * i
    Write 'UBB'
  EndWhile
  j ← j DIV 3
EndWhile

```

- A.  $k^2$
- B.  $k * 3^k$
- C.  $k * (k+1)$
- D.  $3 * k$

24. Se dau următoarele secvențe de cod și numerele naturale  $i, j, a, b$  ( $1 < a, b \leq 10^9$ ).

**Secvența 1 (S1)**

```

i ← 1
While i ≠ b execute
  j ← 1
  While j ≠ a execute
    Write '*'
    j ← j + 1
  EndWhile
  i ← i + 1
EndWhile

```

**Secvența 2 (S2)**

```

i ← 1
While i ≠ a execute
  j ← 1
  While j ≠ b execute
    Write '*'
    j ← j + 1
  EndWhile
  i ← i + 1
EndWhile

```

Care dintre afirmațiile următoare sunt adevărate?

- A. Numărul de caractere afișate de secvența S1 este diferit față de numărul de caractere afișate de secvența S2.
- B. Ambele secvențe au aceeași complexitate timp.
- C. Numărul de caractere afișate de secvența S1 este  $(a - 1) * (b - 1)$ .
- D. Numărul de caractere afișate de secvența S2 este  $a * b$ .

25. Se consideră algoritmul ceFace(nr), unde nr este un număr natural ( $100 \leq nr \leq 2 * 10^9$  la momentul apelului).

```

Algorithm testProprietateNr(n):
  If n ≤ 1 then
    return False
  EndIf
  d ← 2
  While d * d ≤ n execute
    If n MOD d = 0 then
      return False
    EndIf
    d ← d + 1
  EndWhile
  return True
EndAlgorithm

```

```

Algorithm ceFace(nr):
  s ← 0
  c1 ← nr MOD 10
  nr ← nr DIV 10
  c2 ← nr MOD 10
  nr ← nr DIV 10
  While nr ≠ 0 execute
    c3 ← nr MOD 10
    t ← c3 * 100 + c2 * 10 + c1
    If testProprietateNr(t) then
      s ← s + c1 + c2 + c3
    EndIf
    c1 ← c2
    c2 ← c3
    nr ← nr DIV 10
  EndWhile
  return s
EndAlgorithm

```



Precizați valoarea pe care o returnează algoritmul `ceFace(nr)` pentru  $nr = 1271211312$ ?

A. 31

B. 32

C. 33

D. 34

26. Care dintre următorii algoritmi determină corect și returnează valoarea rădăcinii pătrate a numărului natural  $n$  ( $0 < n < 10^5$ ), rotunjit în jos la cel mai apropiat întreg. Operatorul `/` reprezintă împărțirea reală (ex.  $3 / 2 = 1,5$ ).

A.

```
Algorithm radical_A(n):
  x ← 0
  z ← 1
  While z ≤ n execute
    x ← x + 1
    z ← z + 2 * x
    z ← z + 1
  EndWhile
  return x
EndAlgorithm
```

C.

```
//Algoritmul se apelează inițial
//în forma radical_C(n, n)
Algorithm radical_C(n, x):
  eps ← 0.001
  y ← 0.5 * (x + n / x)
  If x - y < eps then
    //se returnează partea
    //întreagă a lui x
    return [x]
  EndIf
  return radical_C(n, y)
EndAlgorithm
```

B.

```
Algorithm radical_B(n):
  s ← 1
  d ← n DIV 2
  While s < d execute
    k ← (s + d) DIV 2
    If k * k ≥ n then
      d ← k
    else
      s ← k + 1
    EndIf
  EndWhile
  If s * s ≤ n then
    return s + 1
  else
    return s - 1
  EndIf
EndAlgorithm
```

D.

```
Algorithm radical_D(n):
  s ← 0
  p ← 0
  k ← 0
  While k < n execute
    k ← k + 3 + p
    p ← p + 2
    s ← s + 1
  EndWhile
  return s
EndAlgorithm
```

27. Știind că  $x$  este număr natural, care dintre următoarele expresii au valoarea *True* dacă și numai dacă  $x$  este număr par care **NU** aparține intervalului deschis  $(10, 20)$ ?

- A. `NOT((x > 10) AND (x < 20)) AND (NOT (x MOD 2 = 1))`
- B. `(x MOD 2 = 0) AND ((x < 10) OR (x > 20))`
- C. `NOT(x MOD 2 = 1) AND ((x > 10) AND (x < 20))`
- D. `NOT((x MOD 4 = 1) OR (x MOD 4 = 3) OR ((x > 10) AND (x < 20)))`

28. Se dă un șir  $a$  de  $n$  numere naturale distincte ( $a[1], a[2], \dots, a[n], 2 \leq n \leq 1000$ ) ordonate strict crescător. Într-un șir se numește *vârf local* un număr cu proprietatea că este strict mai mare decât numărul de pe poziția anterioară, dar și decât numărul de pe poziția următoare. Primul și ultimul element din șir nu pot fi vârfuri locale. Se dorește un algoritm `rearanzare(a, n)` care rearanjează valorile din șir astfel încât șirul să aibă un număr maxim de vârfuri locale și returnează noul șir. Variabila locală  $b$  este un șir. Care dintre următorii algoritmi sunt corecți?

A.

```

Algorithm rearanjare(a, n):
  i ← n
  For p ← 2, n, 2 execute
    b[p] ← a[i]
    i ← i - 1
  EndFor
  For p ← 1, n, 2 execute
    b[p] ← a[i]
    i ← i - 1
  EndFor
  return b
EndAlgorithm

```

B.

```

Algorithm rearanjare(a, n):
  i ← n
  For p ← 2, n, 2 execute
    b[p] ← a[i]
    i ← i - 1
  EndFor
  b[p - 1] ← a[i]
  i ← i - 1
  EndFor
  If n MOD 2 = 1 then
    b[n] ← a[i]
  EndIf
  return b
EndAlgorithm

```

C.

```

Algorithm rearanjare(a, n):
  i ← n
  For p ← 2, n, 2 execute
    b[p] ← a[i]
    i ← i - 1
  EndFor
  i ← 1
  For p ← 1, n, 2 execute
    b[p] ← a[i]
    i ← i + 1
  EndFor
  return b
EndAlgorithm

```

D.

```

Algorithm rearanjare(a, n):
  i ← n
  For p ← 2, n, 3 execute
    b[p] ← a[i]
    i ← i - 1
  EndFor
  b[p - 1] ← a[i]
  i ← i - 1
  If p + 1 ≤ n then
    b[p + 1] ← a[i]
    i ← i - 1
  EndIf
  EndFor
  If n MOD 3 = 1 then
    b[n] ← a[i]
  EndIf
  return b
EndAlgorithm

```

29. Se consideră algoritmul  $f(n, p_1, p_2)$ , unde  $n, p_1$  și  $p_2$  sunt numere naturale strict pozitive ( $1 < n, p_1, p_2 \leq 10^4$  la momentul apelului).

```

Algorithm f(n, p1, p2):
  c ← 0
  While p1 ≤ n execute
    c ← c + n DIV p1
    p1 ← p1 * p2
  EndWhile
  return c
EndAlgorithm

```

Precizați care dintre următoarele afirmații sunt adevărate:

- A. Dacă cei trei parametri au valori egale ( $n = p_1 = p_2$ ), atunci algoritmul returnează întotdeauna valoarea 1.
- B. Dacă  $p_1 = 5$  și  $p_2 = 5$ , algoritmul returnează numărul de cifre de 0 pe care le are  $n!$  la sfârșit.
- C. Dacă  $p_1$  și  $p_2$  au valori egale și mai mari decât 2, atunci algoritmul returnează  $\lceil \log_{p_1} n \rceil$ .
- D. Niciuna dintre celelalte trei afirmații nu este adevărată.

30. Care dintre următorii algoritmi returnează numărul de numere *sumative* din intervalul  $[a, b]$  ( $0 < a < b < 10^6$ )? Un număr natural nenul  $n$  este *sumativ* dacă  $n^2$  se poate scrie ca sumă a  $n$  numere naturale nenule consecutive. De exemplu, 1 și 7 sunt *sumative* deoarece  $1 = 1$ , respectiv  $49 = 4 + 5 + 6 + 7 + 8 + 9 + 10$ .

A.

```
Algorithm sumative(a, b):  
  k ← 0  
  For i ← a, b execute  
    If i MOD 2 ≠ 0 then  
      k ← k + 1  
    EndIf  
  EndFor  
  return k  
EndAlgorithm
```

B.

```
Algorithm sumative(a, b):  
  return (b - a) DIV 2 + (b - a) MOD 2  
        + (a MOD 2 + b MOD 2) DIV 2  
EndAlgorithm
```

C.

```
Algorithm sumative(a, b):  
  k ← 0  
  For i ← a, b execute  
    i2 ← i * i  
    For j ← 2, i - 1 execute  
      If i2 = j * i + (i * (i + 1) DIV 2) then  
        k ← k + 1  
      EndIf  
    EndFor  
  EndFor  
  return k  
EndAlgorithm
```

D.

```
Algorithm sumative(a, b):  
  k ← 0  
  For i ← a, b execute  
    i2 ← i * i  
    For j ← 2, i DIV 2 execute  
      If i2 = j * i + (i * (i + 1) DIV 2) then  
        k ← k + 1  
      EndIf  
    EndFor  
  EndFor  
  return k  
EndAlgorithm
```