

MATEK-INFO verseny – 2026 március 21
Informatika írásbeli

FONTOS MEGJEGYZÉS:

Más pontosítások hiányában:

- Az aritmetikai műveleteket végtelen adattípusokon végezzük (nincs túlcsordulás és alulcsordulás).
- Minden vektort, mátrixot és karakterláncot 1-től sorszámozunk (indexelünk).
- Az aktuális paraméterek értékeire vonatkozó megszorítások a kezdeti hívás pillanatában érvényesek.
- Egy vektor vagy karakterlánc tömbszakaszát a vektor vagy karakterlánc olyan elemei alkotják, amelyek egymás utáni pozíciókon találhatók.
- Ha ugyanabban a sorban több egymásutáni értékadó utasítás található, ezek ";" -vel vannak elválasztva.

1. Legyenek az $a = 5$, $b = 6$, $c = 7$ és $d = 8$ változók. A $|$ -gyal a bitenkénti **OR** műveletet és az $\&$ -sel a bitenkénti **AND** műveletet jelöljük.

A mellékelt kifejezések közül melyeknek az értéke 8?

- A. $(a | b) + (c \& d)$
- B. $(a | (b + c)) \& d$
- C. $a | (b + (c \& d))$
- D. $((a | b) + c) \& d$

2. Legyen a `cevaFace(n)` algoritmus, ahol n egy természetes szám ($10 \leq n \leq 10^6$).

```
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
```

A következő állítások közül melyek igazak?

- A. A `cevaFace(12345)` hívás eredményeképpen az algoritmus 5-öt térít vissza.
- B. A `cevaFace(n)` algoritmus visszatéríti az n szám legkevesbé értékes számjegyét felemelve a legértékesebb számjegyének hatványára.
- C. A `cevaFace(175439)` hívás eredményeképpen az algoritmus 1-et térít vissza.
- D. A `cevaFace(n)` algoritmus visszatéríti az n szám legértékesebb számjegyét felemelve a legkevesbé értékes számjegyének hatványára.

3. Legyen az n természetes szám ($1 \leq n \leq 10$).

Az $F(n, i, P)$ algoritmusok közül melyik téríti vissza $n!$ értékét az $F(n, 1, 1)$ hívás eredményeképpen?

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 ← i + 1  
    Q ← P * i  
    Return F(n, j, Q)  
EndAlgorithm
```

D.

```
Algorithm F(n, i, P):  
    If i = n then  
        Return P  
    EndIf  
    j ← i + 1  
    Q ← P * j  
    Return F(n, j, Q)  
EndAlgorithm
```

4. Legyen az $f(A, n)$ algoritmus, ahol n egy természetes szám ($1 \leq n \leq 10^3$) és A egy n egész típusú számból álló vektor ($0 \leq A[i] \leq 10^5, 1 \leq i \leq n$).

```
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
```

A következő állítások közül melyek igazak?

- A. Az $f([23, 213, 123, 53, 97, 546, 768], 7)$ hívás eredményeképpen az algoritmus 3-at térít vissza.
- B. Az $f([77, 3, 657, 345, 123, 789, 45], 7)$ hívás eredményeképpen az algoritmus 3-at térít vissza.
- C. Az algoritmus visszatéríti a vektor azon elemeinek darabszámát, amelyek legalább 2 számjegyből állnak.
- D. Az $f([11, 31, 5, 37, 9], 5)$ hívás eredményeképpen az algoritmus 3-at térít vissza.

5. Egy informatika könyv oldalai 1-től n -ig ($100 < n < 500$) vannak megszámozva. Azokon az oldalakon, amelyeknek sorszáma 3-nak a többszöröse, de 4-nek nem, két megoldott feladat található. Azokon az oldalakon, amelyeknek sorszáma 4-nek a többszöröse, de 3-nak nem, három megoldott feladat található. Az összes többi oldal csak elméletet tartalmaz.

Mely képlet-pár számítja ki azoknak az oldalaknak a darabszámát, T -vel jelölve, amelyek csak elméletet tartalmaznak és a könyvben található összes megoldott feladat darabszámát, P -vel jelölve?

- 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. Legyen az $f(n)$ algoritmus, ahol n egy természetes szám ($1 \leq n < 20$).

```
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
```

A következő állítások közül melyek igazak?

- A. Ha n osztható 3-mal, az $f(n)$ algoritmus egy 3-mal osztható számot térít vissza.
- B. Az algoritmus mindig egy n számjegyű számot térít vissza.
- C. Nem létezik olyan n szám, amelyre az $f(n)$ algoritmus által visszatérített érték osztható legyen 3-mal és 5-tel is.
- D. Az $f(n)$ algoritmus által visszatérített érték párossága azonos az n szám párosságával.

7. Legyen a $h(n, k)$ algoritmus, ahol n és k természetes számok ($1 \leq n, k \leq 10^5$).

```
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
```

Mely értéket téríti vissza az algoritmus a $h(2000, 21)$ hívás eredményeképpen?

- A. 2000
- B. 21
- C. 5
- D. A $h(2000, 21)$ hívás eredményeképpen az algoritmus végrehajtása nem ér véget.

8. Legyen az $f(k)$ algoritmus, ahol k egy természetes szám ($1 \leq k \leq 10^4$).

Algorithm $f(k)$:

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

A következő állítások közül melyek igazak?

- A. Az $f(10)$ hívás eredményeképpen az algoritmus 123-at térít vissza.
- B. Bármely k érték esetében az algoritmus egy páros számot térít vissza.
- C. Az $f(3)$ hívás eredményeképpen az algoritmus 15-öt térít vissza.
- D. Az $f(21)$ hívás eredményeképpen az algoritmus 484-et, térít vissza.

9. A következő, 10-es számrendszerben megadott számok közül melyiknek palindrom a bináris (kettes számrendszerbeli) alakja?

- A. $515_{(10)}$ B. $1967_{(10)}$ C. $1057_{(10)}$ D. $2973_{(10)}$

10. Legyen az $s(arr, n, x)$ algoritmus, ahol n egy természetes szám ($1 \leq n \leq 10^3$), arr egy n természetes számból álló vektor ($arr[1], arr[2], \dots, arr[n], 0 \leq arr[i] \leq 10^3, i = 1, 2, \dots, n$), és x egy természetes szám ($0 \leq x \leq 10^3$).

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
```

A következő állítások közül melyek igazak?

- A. Az $s(arr, n, x)$ algoritmus időbonyolultsága $O(\log \log n)$.
- B. Az $s([10, 20, 30, 40, 50, 60, 70], 7, 20)$ hívás eredményeképpen az algoritmus 2-t térít vissza.
- C. Az $s([10, 20, 30, 40, 50, 60, 70], 7, 35)$ hívás eredményeképpen az algoritmus -1-et térít vissza.
- D. Az $s([1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 4], 11, 4)$ hívás eredményeképpen az algoritmus 11-et térít vissza.

11. Legyen az $eCrescatoare(a, n, p, q)$ algoritmus, ahol a egy n természetes számból álló vektor ($2 \leq n \leq 10^9, 0 \leq a[i] \leq 10^9, 1 \leq i \leq n$), p és q természetes számok ($1 \leq p \leq q \leq n$).

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

A következő utasításcsoportok közül melyiket lehet beszúrni a "... " helyére ahhoz, hogy az algoritmus *True*-t térítsen vissza, ha az $a[p], \dots, a[q]$ tömbszakasz szigorúan növekvő?

- 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. Legyen az $\text{altceva}(n, k)$ algoritmus, ahol n és k természetes számok ($1 \leq n \leq 10^4$, $0 \leq k \leq 10^4$).

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

```

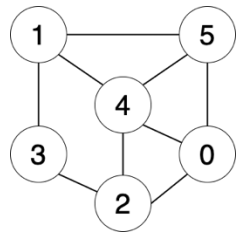
  If  $n < k$  then
    Return False
  EndIf
  If  $n = k$  then
    Return True
  EndIf
  Return  $\text{altceva}(n - k, k + 2)$ 
EndAlgorithm

```

A következő állítások közül melyek igazak?

- A. Az $\text{altceva}(256, 0)$ hívás eredményeképpen az algoritmus *True*-t térít vissza.
- B. Ha az n szám felírható két egymásutáni szám szorzataként és $k = 0$, az $\text{altceva}(n, k)$ hívás eredményeképpen az algoritmus *True*-t térít vissza.
- C. Ha n páratlan szám, az $\text{altceva}(n, 0)$ hívás eredményeképpen az algoritmus *False*-t térít vissza.
- D. Az $\text{altceva}(272, 0)$ hívás eredményeképpen és az $\text{altceva}(380, 0)$ hívás eredményeképpen az algoritmus *True*-t térít vissza.

13. Legyen a következő gráf:



A.

Szélességi bejárás: 3 2 4 1 0 5

Mélységi bejárás: 3 2 4 1 0 5

C.

Szélességi bejárás: 3 2 1 4 0 5

Mélységi bejárás: 3 1 5 2 4 0

B.

Szélességi bejárás: 3 1 4 2 5 0

Mélységi bejárás: 3 2 4 1 5 0

D.

Szélességi bejárás: 3 1 2 4 5 0

Mélységi bejárás: 3 1 5 0 4 2

14. Legyen a $\text{fill}(\text{mat}, n, \text{first}, d)$ algoritmus, ahol n egy természetes szám ($1 \leq n \leq 200$), mat egy n soros és n oszlopos mátrix ($\text{mat}[1][1], \text{mat}[1][2], \dots, \text{mat}[1][n], \dots, \text{mat}[n][n]$), first és d két természetes szám ($0 \leq \text{first}, d \leq 200$).

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

```

A következő állítások közül melyek igazak?

- A. A $\text{fill}(\text{mat}, 10, 1, 1)$ hívás eredményeképpen a mátrix utolsó oszlopában levő elemek összege 505.
- B. A $\text{fill}(\text{mat}, 10, 1, 1)$ hívás eredményeképpen a mátrix utolsó sorában levő elemek összege 955.
- C. A $\text{fill}(\text{mat}, n, \text{first}, d)$ hívás eredményeképpen a visszatérített érték a d szám többszöröse.
- D. A $\text{fill}(\text{mat}, 101, 0, 5)$ hívás eredményeképpen a mátrix jobb alsó sarkában levő elem értéke 51000.

15. Legyenek az $\text{Alg1}(n, x)$ és az $\text{Alg2}(n, x)$ algoritmusok, ahol n egy természetes szám ($3 \leq n \leq 10^4$), és x egy n egész típusú számból álló vektor ($x[1], x[2], \dots, x[n], -100 \leq x[i] \leq 100, i = 1, 2, \dots, n$).

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
     $x[j + 1] \leftarrow$  key
  EndFor
EndAlgorithm

```

A következő állítások közül melyek igazak?

- A. Bármely növekvően rendezett x vektor esetében, az Alg1(n, x) algoritmus (*)-gal jelölt utasítása többször hajtódik végre, mint az Alg2(n, x) algoritmus (*)-gal jelölt utasítása.
- B. Létezik egy 5 elemű bemeneti x vektor, amelynek esetében az Alg1(n, x) algoritmus (+)-szal jelölt ciklusának 3 iterációja után az x vektor [2, 1, 3, 7, 8] lesz és az Alg2(n, x) algoritmus (+)-szal jelölt ciklusának 3 iterációja után az x vektor [2, 3, 7, 8, 1] lesz.
- C. Ha az Alg1(n, x) algoritmus (+)-szal jelölt ciklusának egyik iterációjában nem történik egyetlen felcserélés sem, akkor a vektor már növekvően rendezett és a következő iterációk nem változtatják meg a vektort.
- D. Az Alg1(n, x) algoritmus időbonyolultsága $O(n^2)$, és az Alg2(n, x) algoritmus időbonyolultsága $O(n \log n)$.

16. Legyen az $f(n, x)$ algoritmus, ahol n természetes szám ($1 \leq n \leq 10^3$), és x egy n egész típusú számból álló vektor ($x[1], x[2], \dots, x[n], -10^3 \leq x[i] \leq 10^3, i = 1, 2, \dots, n$).

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
             $x[j] \leftarrow x[j + 1]$ 
        EndFor
        n ← n - 1
    Else
        i ← i + 1
    EndIf
EndWhile
Return n
EndAlgorithm

```

A következő állítások közül melyek igazak?

- A. Az $f(5, [2, 5, 3, 6, 1])$ és az $f(5, [2, 5, 3, 1, -1])$ hívások eredményeképpen az algoritmus ugyanazt az értéket téríti vissza.
- B. Ha az x vektor növekvően rendezett, és k a vektor maximális értékének indexe, az $f(n, x)$ hívás eredményeképpen az algoritmus k értékét téríti vissza.
- C. Ha $n = 4$, létezik egy olyan x vektor, amely különböző értékű páros számot tartalmaz és az $f(n, x)$ hívás eredményeképpen az algoritmus 2-t térít vissza.
- D. Ha az x vektor csak páratlan számokat tartalmaz, és m az $f(n, x)$ algoritmus hívásának eredménye, akkor n egyenlő m -mel.

17. Legyen a $\text{verif}(x, n, y, m)$ algoritmus, ahol n, m természetes számok ($1 \leq n, m \leq 200$), x egy n egész típusú különböző értékű számból álló vektor ($x[1], x[2], \dots, x[n], -10^3 \leq x[i] \leq 10^3, i = 1, 2, \dots, n$), és y egy m egész típusú különböző értékű számból álló vektor ($y[1], y[2], \dots, y[m], -10^3 \leq y[i] \leq 10^3, i = 1, 2, \dots, m$).

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

```

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

```

A következő állítások közül melyek igazak?

- A. Ha az x vektor hossza nagyobb, mint az y vektor hossza, akkor az algoritmus *False*-t térít vissza.
- B. Ha a $\text{verif}(x, n, y, m)$ hívás eredményeképpen az algoritmus *True*-t térít vissza, az x vektor összes eleme szorzatának utolsó számjegye azonos az y vektor összes eleme szorzatának utolsó számjegyével.
- C. A $\text{verif}([1, 2, 3, 4], 4, [1, 7, 2, 5, 3, 4, 10], 7)$ és a $\text{verif}([4, 2, 1, 3], 4, [1, 10, 2, 5, 3, 4, 7], 7)$ hívások eredményeképpen az algoritmus azonos értéket térít vissza.
- D. Ha az x vektor a $[-20, 100]$ intervallum összes páros számát tartalmazza, növekvően rendezve, és az y vektor a $[-20, 100]$ intervallum összes 4-gyel osztható számát tartalmazza, csökkenően rendezve, a $\text{verif}(x, n, y, m)$ hívás eredményeképpen az algoritmus *True*-t térít vissza.

18. Legyen az $\text{enigma}(x, n)$ algoritmus, ahol n természetes szám ($1 \leq n \leq 100$), és x egy n elemű, természetes számokból álló vektor ($x[1], x[2], \dots, x[n]$, $10 \leq x[i] \leq 10^5$, $i = 1, 2, \dots, n$).

```

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

```

A következő állítások közül melyek igazak?

- A. Ha az $\text{enigma}(x, n)$ hívás eredményeképpen az algoritmus n darab '*' karaktert ír ki, akkor az x vektor növekvően rendezett.
- B. Az $\text{enigma}([19, 28, 37, 46], 4)$ hívás eredményeképpen az algoritmus 10-et térít vissza.
- C. Ha az x vektor növekvően rendezett, akkor az $\text{enigma}(x, n)$ hívás eredményeképpen az algoritmus n darab '*' karaktert ír ki.
- D. Az $\text{enigma}([99, 95, 91, 11], 4)$ hívás eredményeképpen az algoritmus nem ír ki egyetlen '*' karaktert sem.

19. Legyen a $\text{modifica}(X, n)$ algoritmus, ahol n egy természetes szám ($1 \leq n \leq 100$), és X egy n elemű természetes számokból álló vektor ($X[1], X[2], \dots, X[n]$, $10 \leq X[i] \leq 10^5$, $i = 1, 2, \dots, n$).

```

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

```

A következő állítások közül melyek igazak??

- A. A $\text{modifica}([51, 24, 65, 756, 38, 82, 13, 22], 8)$ hívás eredményeképpen az algoritmus az $[51, 82, 22, 13, 24, 65, 756, 38]$ vektort téríti vissza.
- B. A $\text{modifica}([1845, 9076, 3545, 242, 13, 512, 61, 86], 8)$ hívás eredményeképpen az algoritmus a $[9076, 512, 61, 13, 242, 3545, 86, 1845]$ vektort téríti vissza.
- C. A $\text{modifica}([372, 4932, 42, 5201, 8193, 4672, 195, 88, 57], 9)$ hívás eredményeképpen az algoritmus a $[42, 57, 88, 195, 372, 4932, 4672, 5201, 8193]$ vektort téríti vissza.
- D. A $\text{modifica}([87, 19, 28, 656, 943, 2061, 1121], 7)$ hívás eredményeképpen az algoritmus a $[2061, 1121, 943, 656, 87, 28, 19]$ vektort téríti vissza.

20. Legyen a $\text{ceface}(k, n, x)$ algoritmus, ahol n és k természetes számok ($1 \leq k \leq n \leq 10^3$), és x egy n természetes számból álló vektor ($0 \leq x[1], x[2], \dots, x[n] \leq 10^3$).

```

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

```

A következő állítások közül melyek igazak?

- A. A `ceface(1, 5, x)` hívás eredményeképpen az algoritmus az első sorban csak egyenlő értékeket ír ki.
- B. A `ceface(1, 5, x)` hívás eredményeképpen az ötödik sorban az algoritmus 2-t és 3-at ír ki.
- C. Az algoritmus kiírja az $\{1, 2, \dots, n\}$ halmaz összes k elemű részhalmazát.
- D. A `ceface(1, n, x)` hívás eredményeképpen az algoritmus 2^{n-1} sort ír ki.

21. Legyen a `calcul(n)` algoritmus, ahol n egy természetes szám ($2 < n < 10$). A `zerosMatrice(n)` algoritmus egy n soros és n oszlopos négyzetes mátrix minden elemét 0-val inicializálja. A `zerosVector(n)` algoritmus egy n elemű vektort térít vissza, melynek minden eleme 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

```

Mely értékeket tartalmazza a visszatérített vektor a `calcul(n)` hívás eredményeképpen?

- A. A 11 szám hatványait 0-tól $n - 2$ -ig
- B. A 11 szám első n többszörösét
- C. A $2 * x[i] - 1$ ($i > 0$) sor elemeit, ahol az x sor a következőképpen van definiálva: $x[1] = 1, x[2] = 6, x[k] = x[k - 1] * 10 + 1$ (ha $k > 2$)
- D. Az x sor elemeit, ahol az x sor a következőképpen van definiálva: $x[1] = 1, x[k] = x[k - 1] * 11$ (ha $k > 1$)

22. Legyen a `genereaza(n, a, k, p)` algoritmus, ahol n, k és p természetes számok ($1 \leq n \leq 12, 0 \leq p \leq n, 1 \leq k \leq 10^6$), és a egy n természetes számból álló vektor ($a[1], a[2], \dots, a[n], 0 \leq a[i] \leq 10^4, i = 1, 2, \dots, n$).

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

```

Mely értékeket tartalmazza az a vektor a `genereaza(6, [0, 0, 0, 0, 0, 0], 482, 0)` hívás eredményeképpen?

- 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. Legyen az $f(n, a)$ algoritmus, ahol n egy természetes szám ($2 \leq n \leq 10^6$) és a egy n természetes számból álló vektor ($a[1], a[2], \dots, a[n], 1 \leq a[i] \leq 10^6, i = 1, 2, \dots, n$). A $\text{zerosMat}(n)$ algoritmus egy n soros és n oszlopos négyzetes mátrixot térít vissza, melynek minden eleme 0.

```

Algorithm  $f(n, a)$ :
   $m \leftarrow \text{zerosMat}(n)$ 
   $i \leftarrow 1$ 
   $j \leftarrow n$ 
  While  $i \leq n$  AND  $j \geq 1$  execute
    If  $a[i] < a[j]$  then
       $m[i][j] \leftarrow 1$ 
       $i \leftarrow i + 1$ 
    Else
       $m[i][j] \leftarrow 2$ 
       $j \leftarrow j - 1$ 
    EndIf
  EndWhile
  Return  $m$ 
EndAlgorithm

```

A következő állítások közül melyek igazak?

- A. Ha az a vektor növekvően rendezett, akkor az $f(n, a)$ hívás eredményeként a 2-es érték n -szer jelenik meg a mátrixban.
- B. Ha az a vektor csökkenően rendezett, akkor az $f(n, a)$ hívás eredményeként 2-es érték egyszer jelenik meg a mátrixban.
- C. Ha az a vektor minden eleme azonos értékű, akkor az $f(n, a)$ hívás eredményeként a mátrix minden sorában lesz legalább egy nullától különböző érték.
- D. Az $f(5, [1, 1, 4, 6, 9])$ hívás eredményeként a mátrix utolsó sorában csak 2-es értékek lesznek.

24. Három színű gyöngyünk van, amelyeket a "0", "1" és "2" karakterláncokkal ábrázoljuk. A $\text{colier}(n)$ algoritmust alkalmazva, ahol n egy természetes szám ($4 \leq n \leq 3 \cdot 2^{10}$), egy olyan n hosszúságú gyöngysort szeretnénk létrehozni, amely nem tartalmaz két egymásutáni azonos tömbszakaszt. Például, ha $n = 4$, a gyöngysor ábrázolása nem lehet "0211" (kétszer egymás után előfordul az "1" tömbszakasz), de "0202" sem (a "02" előfordul kétszer egymás után). Egy lehetséges megoldás a "0201". A + operátor a karakterláncok összefűzését jelöli, a $\text{len}(s)$ algoritmus pedig az s karakterlánc hosszát téríti vissza.

```

Algorithm  $\text{extinde}(cuv)$ :
   $res \leftarrow ""$ 
   $i \leftarrow 1$ 
  While  $i \leq \text{len}(cuv)$  execute
     $c \leftarrow cuv[i]$ 
    If  $c = "0"$  then
       $res \leftarrow res + "012"$ 
    Else
      If  $c = "1"$  then
         $res \leftarrow res + "02"$ 
      Else
         $res \leftarrow res + "1"$ 
      EndIf
    EndIf
     $i \leftarrow i + 1$ 
  EndWhile
  Return  $res$ 
EndAlgorithm

```

```

Algorithm  $\text{colier}(n)$ :
   $cuv \leftarrow "0"$ 
  While  $\text{len}(cuv) < n$  execute
     $cuv \leftarrow \text{extinde}(cuv)$ 
  EndWhile
   $res \leftarrow ""$ 
   $i \leftarrow 1$ 
  While  $i \leq n$  execute
     $res \leftarrow res + cuv[i]$ 
     $i \leftarrow i + 1$ 
  EndWhile
  Return  $res$ 
EndAlgorithm

```

A következő állítások közül melyek igazak?

- A. A $\text{colier}(n)$ algoritmus, n értékétől függetlenül, a követelményeknek megfelelő gyöngysorábrázolást téríti vissza.
- B. A $\text{colier}(8)$ hívás eredményeképpen az algoritmus a "01202101" gyöngysorábrázolást téríti vissza.
- C. A $\text{colier}(10)$ és a $\text{colier}(12)$ hívások végrehajtásának során az $\text{extinde}(cuv)$ algoritmus azonos számszor hívódik meg.
- D. A $\text{colier}(n)$ algoritmus akkor és csakis akkor térít vissza a követelményeknek megfelelő gyöngysorábrázolást, ha n felírható $3 \cdot 2^k$ alakban ($1 \leq k \leq 10$).

BABEŞ-BOLYAI TUDOMÁNYEGYETEM
MATEMATIKA ÉS INFORMATIKA KAR

MATEK-INFO verseny – 2026 március 21

Informatika írásbeli

JAVÍTÁSI KULCS & MEGOLDÁSOK

HIVATALBÓL: 10 pont

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