

Mate-Info-Wettbewerb – 21. März 2026
Schriftliche Prüfung in Informatik

WICHTIGER HINWEIS:

Sofern nicht anders angegeben:

- Alle arithmetischen Operationen werden mit unbegrenzten Datentypen durchgeführt (kein *Überlauf/Unterlauf*).
- Die Indizierung aller Vektoren, Matrizen und Zeichenfolgen beginnt bei 1.
- Alle Einschränkungen beziehen sich auf die Werte der aktuellen Parameter zum Zeitpunkt des ursprünglichen Aufrufs.
- Eine Teilsequenz eines Vektors oder einer Zeichenfolge besteht aus Elementen, die aufeinanderfolgende Positionen im Vektor bzw. in der Zeichenfolge einnehmen.
- Wenn mehrere aufeinanderfolgende Zuweisungsanweisungen in derselben Zeile auftreten, werden sie durch ";" voneinander getrennt.

1. Seien die Variablen $a = 5$, $b = 6$, $c = 7$ und $d = 8$.
Wir bezeichnen mit $|$ die **Bit-OR-Operation** und mit
& die **Bit-AND-Operation**.

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

Welche der folgenden Ausdrücke haben den Wert 8?

2. Betrachten wir den Algorithmus `cevaFace(n)`, wobei n eine natürliche Zahl ist ($10 \leq n \leq 10^6$):

```
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
```

Welche der folgenden Aussagen sind wahr?

- A. Nach dem Aufruf von `cevaFace(12345)` gibt der Algorithmus 5 zurück.
- B. Der Algorithmus `cevaFace(n)` gibt die niedrigstwertige Ziffer der Zahl n zur Potenz ihrer höchstwertigen Ziffer zurück.
- C. Nach dem Aufruf von `cevaFace(175439)` gibt der Algorithmus 1 zurück.
- D. Der Algorithmus `cevaFace(n)` gibt die höchstwertige Ziffer der Zahl n zur Potenz ihrer niedrigstwertigen Ziffer zurück.

3. Betrachten Sie die natürliche Zahl n ($1 \leq n \leq 10$).

Welcher der Algorithmen $F(n, i, P)$ gibt $n!$ zurück, wenn $F(n, 1, 1)$ aufgerufen wird?

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. Betrachten wir den Algorithmus $f(A, n)$, wobei n eine natürliche Zahl ist ($1 \leq n \leq 10^3$) und A ein Vektor mit n ganzzahligen Elementen ist ($0 \leq A[i] \leq 10^5, 1 \leq i \leq n$).

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Nach dem Aufruf von $f([23, 213, 123, 53, 97, 546, 768], 7)$ gibt der Algorithmus 3 zurück.
- B. Nach dem Aufruf von $f([77, 3, 657, 345, 123, 789, 45], 7)$ gibt der Algorithmus 3 zurück.
- C. Der Algorithmus gibt die Anzahl der Elemente im Vektor zurück, die mindestens 2 Ziffern haben.
- D. Nach dem Aufruf von $f([11, 31, 5, 37, 9], 5)$ gibt der Algorithmus 3 zurück.

5. Ein Informatikbuch hat Seiten, die von 1 bis n ($100 < n < 500$) nummeriert sind. Auf den Seiten, die Vielfache von 3, aber nicht von 4 sind, befinden sich 2 gelöste Aufgaben. Auf den Seiten, die Vielfache von 4, aber nicht von 3 sind, befinden sich 3 gelöste Aufgaben. Alle anderen Seiten enthalten nur Theorie.

Welche Formelpaare berechnen die Anzahl der Seiten, die nur Theorie enthalten, bezeichnet mit T , und die Gesamtzahl der gelösten Aufgaben im Buch, bezeichnet mit 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. Betrachten wir den Algorithmus $f(n)$, wobei n eine natürliche Zahl ist ($1 \leq n < 20$).

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Für Werte von n , die durch 3 teilbar sind, gibt der Algorithmus $f(n)$ einen durch 3 teilbaren Wert zurück.
- B. Der Algorithmus gibt immer eine Zahl mit n Ziffern zurück.
- C. Es gibt kein n , für das der vom Algorithmus $f(n)$ zurückgegebene Wert sowohl durch 3 als auch durch 5 teilbar ist.
- D. Der vom Algorithmus $f(n)$ zurückgegebene Wert hat dieselbe Parität wie n .

7. Betrachten wir den Algorithmus $h(n, k)$, wobei n und k natürliche Zahlen sind ($1 \leq n, k \leq 10^5$).

```

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

```

Was ist der Rückgabewert nach dem Aufruf von $h(2000, 21)$?

- A. 2000
- B. 21
- C. 5
- D. Nach dem Aufruf von $h(2000, 21)$ beendet der Algorithmus seine Ausführung nicht.

8. Betrachten wir den Algorithmus $f(k)$, wobei k eine natürliche Zahl ist ($1 \leq k \leq 10^4$).

```

Algorithm f(k):
  x ← 0; i ← 1
  While i ≤ 2 * k + 1 execute
    x ← x + i
    i ← i + 2
  EndWhile
  Return x
EndAlgorithm

```

Welche der folgenden Aussagen sind wahr?

- A. Nach dem Aufruf von $f(10)$ wird der Wert 123 zurückgegeben.
- B. Für jeden Wert k wird eine gerade Zahl zurückgegeben.
- C. Nach dem Aufruf von $f(3)$ wird der Wert 15 zurückgegeben.
- D. Nach dem Aufruf von $f(21)$ wird der Wert 484 zurückgegeben.

9. Welche der folgenden Zahlen in der Basis 10 sind Palindrome, wenn wir sie in der Basis 2 darstellen?

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

10. Betrachten wir den Algorithmus $s(arr, n, x)$, wobei n eine natürliche Zahl ist ($1 \leq n \leq 10^3$), arr ein Vektor mit n Elementen aus den natürlichen Zahlen ist ($arr[1], arr[2], \dots, arr[n], 0 \leq arr[i] \leq 10^3$, für $i = 1, 2, \dots, n$) und x eine natürliche Zahl ($0 \leq x \leq 10^3$) ist:

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Der Algorithmus $s(arr, n, x)$ hat eine Zeitkomplexität von $O(\log \log n)$.
- B. Nach dem Aufruf von $s([10, 20, 30, 40, 50, 60, 70], 7, 20)$ gibt der Algorithmus 2 zurück.
- C. Nach dem Aufruf von $s([10, 20, 30, 40, 50, 60, 70], 7, 35)$ gibt der Algorithmus -1 zurück.
- D. Nach dem Aufruf $s([1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 4], 11, 4)$ gibt der Algorithmus 11 zurück.

11. Betrachten wir den Algorithmus $eCrescatoare(a, n, p, q)$, wobei a ein Vektor aus n natürlichen Zahlen ist ($2 \leq n \leq 10^9, 0 \leq a[i] \leq 10^9$ für $1 \leq i \leq n$) und p und q natürliche Zahlen sind ($1 \leq p \leq q \leq n$).

```

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

```

Welche der folgenden Befehlssequenzen kann anstelle der Auslassungspunkte eingefügt werden, damit der Algorithmus „True“ zurückgibt, wenn die Teilsequenz $a[p], \dots, a[q]$ streng monoton steigend ist?

- 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. Betrachten wir den Algorithmus $\text{altceva}(n, k)$, wobei n und k natürliche Zahlen sind ($1 \leq n \leq 10^4, 0 \leq k \leq 10^4$):

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

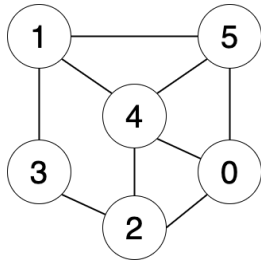
```

Welche der folgenden Aussagen sind wahr?

- A. Nach dem Aufruf von $\text{altceva}(256, 0)$ gibt der Algorithmus *True* zurück.
- B. Wenn die Zahl n als Produkt zweier aufeinanderfolgender Zahlen geschrieben werden kann und $k = 0$ ist, gibt der Aufruf von $\text{altceva}(n, k)$ *True* zurück.
- C. Wenn n eine ungerade Zahl ist, wird nach dem Aufruf von $\text{altceva}(n, 0)$ *False* zurückgegeben.
- D. Nach dem Aufruf von $\text{altceva}(272, 0)$ und $\text{altceva}(380, 0)$ wird *True* zurückgegeben.

13. Betrachten wir das folgenden Graph:

Welche der folgenden Breiten- und Tiefentraversierungen sind korrekt, wenn man vom Knoten 3 aus startet?



- | | |
|---|---|
| <p>A.</p> <p>Breitentersierung: 3 2 4 1 0 5</p> <p>Tiefentrersierung: 3 2 4 1 0 5</p> | <p>B.</p> <p>Breitentersierung: 3 1 4 2 5 0</p> <p>Tiefentrersierung: 3 2 4 1 5 0</p> |
| <p>C.</p> <p>Breitentersierung: 3 2 1 4 0 5</p> <p>Tiefentrersierung: 3 1 5 2 4 0</p> | <p>D.</p> <p>Breitentersierung: 3 1 2 4 5 0</p> <p>Tiefentrersierung: 3 1 5 0 4 2</p> |

14. Betrachten wir den Algorithmus $\text{fill}(\text{mat}, n, \text{first}, d)$, wobei n eine natürliche Zahl ist ($1 \leq n \leq 200$), mat eine Matrix mit n Zeilen und n Spalten ist ($\text{mat}[1][1], \text{mat}[1][2], \dots, \text{mat}[1][n], \dots, \text{mat}[n][n]$), und first und d zwei natürliche Zahlen sind ($0 \leq \text{first}, d \leq 200$).

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

```

Welche der folgenden Aussagen sind wahr?

- A. Nach dem Aufruf von $\text{fill}(\text{mat}, 10, 1, 1)$ beträgt die Summe der Elemente in der letzten Spalte der Matrix 505.
- B. Nach dem Aufruf von $\text{fill}(\text{mat}, 10, 1, 1)$ beträgt die Summe der Elemente in der letzten Zeile der Matrix 955.
- C. Nach dem Aufruf von $\text{fill}(\text{mat}, n, \text{first}, d)$ ist der Rückgabewert ein Vielfaches der Zahl d .
- D. Nach dem Aufruf von $\text{fill}(\text{mat}, 101, 0, 5)$ ist das Element in der rechten unteren Ecke der Matrix 51000.

15. Betrachten wir die Algorithmen $\text{Alg1}(n, x)$ und $\text{Alg2}(n, x)$, wobei n eine natürliche Zahl ist ($3 \leq n \leq 10^4$) und x ein Vektor mit n ganzzahligen Elementen ist ($x[1], x[2], \dots, x[n], -100 \leq x[i] \leq 100$, für $i = 1, 2, \dots, n$).

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

```

Welche der folgenden Aussagen sind wahr?

- A. Für jeden aufsteigend sortierten Vektor x wird die mit (*) markierte Anweisung im Algorithmus Alg1(n, x) öfter ausgeführt als die mit (*) markierte Anweisung im Algorithmus Alg2(n, x).
- B. Es gibt einen Eingabevektor x mit der Größe 5, sodass nach 3 Iterationen des mit (+) markierten Zyklus aus Alg1(n, x) der Vektor x genau [2, 1, 3, 7, 8] sei, und nach 3 Iterationen des mit (+) markierten Zyklus in Alg2(n, x) der Vektor x genau [2, 3, 7, 8, 1] sei.
- C. Im Algorithmus Alg1(n, x) gilt: Wenn bei einer Iteration der mit (+) markierten Schleife kein Austausch stattfindet, ist der Vektor bereits aufsteigend sortiert und die folgenden Iterationen ändern den Vektor nicht mehr.
- D. Der Algorithmus Alg1(n, x) hat eine Zeitkomplexität von $O(n^2)$, während der Algorithmus Alg2(n, x) eine Zeitkomplexität von $O(n \log n)$ hat.

16. Betrachten wir den Algorithmus $f(n, x)$, wobei n eine natürliche Zahl ist ($1 \leq n \leq 10^3$) und x ein Vektor mit n ganzzahligen Elementen ist ($x[1], x[2], \dots, x[n]$, $-10^3 \leq x[i] \leq 10^3$, für $i = 1, 2, \dots, n$):

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

```

Welche der folgenden Aussagen sind wahr?

- A. Nach den Aufrufen $f(5, [2, 5, 3, 6, 1])$ und $f(5, [2, 5, 3, 1, -1])$ gibt der Algorithmus denselben Wert zurück.
- B. Wenn der Vektor x aufsteigend sortiert ist und k den Index des Maximalwerts in diesem Vektor darstellt, ist der Rückgabewert nach dem Aufruf von $f(n, x)$ k .
- C. Für $n = 4$ gibt es einen Vektor x , der verschiedene gerade Werte enthält, für die der Rückgabewert nach dem Aufruf von $f(n, x)$ 2 ist.
- D. Wenn x ein Vektor ist, der nur ungerade Zahlen enthält, und m das Ergebnis des Aufrufs $f(n, x)$ ist, dann ist n gleich m .

17. Betrachten Sie den Algorithmus $\text{verif}(x, n, y, m)$, wobei n, m natürliche Zahlen sind ($1 \leq n, m \leq 200$), x ein Vektor mit n verschiedenen ganzen Zahlen ist ($x[1], x[2], \dots, x[n]$, $-10^3 \leq x[i] \leq 10^3$, für $i = 1, 2, \dots, n$) und y ein Vektor mit m verschiedenen ganzen Zahlen ($y[1], y[2], \dots, y[m]$, $-10^3 \leq y[i] \leq 10^3$, für $i = 1, 2, \dots, m$) ist.

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

```

Welche der folgenden Aussagen sind wahr?

- A. Wenn die Länge des Vektors x größer ist als die Länge des Vektors y , gibt der Algorithmus *False* zurück.
- B. Wenn der Algorithmus nach dem Aufruf von $\text{verif}(x, n, y, m)$ *True* zurückgibt, hat das Produkt aller Elemente im Vektor x dieselbe letzte Ziffer wie das Produkt aller Elemente im Vektor y .
- C. Nach den Aufrufen $\text{verif}([1, 2, 3, 4], 4, [1, 7, 2, 5, 3, 4, 10], 7)$ und $\text{verif}([4, 2, 1, 3], 4, [1, 10, 2, 5, 3, 4, 7], 7)$ gibt der Algorithmus das gleiche Ergebnis zurück.
- D. Wenn der Vektor x aus allen geraden Zahlen im Intervall $[-20, 100]$ in aufsteigender Reihenfolge besteht und der Vektor y aus allen durch 4 teilbaren Zahlen im Intervall $[-20, 100]$ in absteigender Reihenfolge besteht, gibt der Algorithmus nach dem Aufruf von $\text{verif}(x, n, y, m)$ *True* zurück.

18. Betrachten wir den Algorithmus $\text{enigma}(x, n)$, wobei n eine natürliche Zahl ist ($1 \leq n \leq 100$) und x ein Vektor mit n Elementen natürlicher Zahlen ($x[1], x[2], \dots, x[n]$, $10 \leq x[i] \leq 10^5$, für $i = 1, 2, \dots, n$) ist.

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Wenn nach dem Aufruf von $\text{enigma}(x, n)$ n Zeichen '*' angezeigt werden, ist der Vektor x in aufsteigender Reihenfolge sortiert.
- B. Nach dem Aufruf von $\text{enigma}([19, 28, 37, 46], 4)$ gibt der Algorithmus 10 zurück.
- C. Wenn der Vektor x aufsteigend sortiert ist, werden nach dem Aufruf von $\text{enigma}(x, n)$ n Zeichen '*' angezeigt.
- D. Nach dem Aufruf von $\text{enigma}([99, 95, 91, 11], 4)$ zeigt der Algorithmus kein Zeichen '*' an.

19. Betrachten wir den Algorithmus $\text{modifica}(X, n)$, wobei n eine natürliche Zahl ist ($1 \leq n \leq 100$) und X ein Vektor mit n Elementen natürlicher Zahlen ist ($X[1], X[2], \dots, X[n]$, $10 \leq X[i] \leq 10^5$, für $i = 1, 2, \dots, n$):

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Nach dem Aufruf von $\text{modifica}([51, 24, 65, 756, 38, 82, 13, 22], 8)$ wird der Vektor $[51, 82, 22, 13, 24, 65, 756, 38]$ zurückgegeben.
- B. Nach dem Aufruf von $\text{modifica}([1845, 9076, 3545, 242, 13, 512, 61, 86], 8)$ wird der Vektor $[9076, 512, 61, 13, 242, 3545, 86, 1845]$ zurückgegeben.
- C. Nach dem Aufruf von $\text{modifica}([372, 4932, 42, 5201, 8193, 4672, 195, 88, 57], 9)$ wird der Vektor $[42, 57, 88, 195, 372, 4932, 4672, 5201, 8193]$ zurückgegeben.
- D. Nach dem Aufruf von $\text{modifica}([87, 19, 28, 656, 943, 2061, 1121], 7)$ wird der Vektor $[2061, 943, 2061, 1121, 943, 656, 87, 28, 19]$ zurückgegeben.

20. Betrachten wir den Algorithmus $\text{ceface}(k, n, x)$, wobei n und k natürliche Zahlen sind ($1 \leq k \leq n \leq 10^3$) und x ein Vektor aus n natürlichen Zahlen ist ($0 \leq x[1], x[2], \dots, x[n] \leq 10^3$).

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Nach dem Aufruf von `ceface(1, 5, x)` werden in der ersten Zeile nur gleiche Werte angezeigt.
- B. Nach dem Aufruf von `ceface(1, 5, x)` werden in der fünften Zeile die Werte 2 und 3 angezeigt.
- C. Der Algorithmus zeigt alle Teilmengen mit k Elementen der Menge $\{1, 2, \dots, n\}$ an.
- D. Nach dem Aufruf von `ceface(1, n, x)` werden 2^{n-1} Zeilen angezeigt.

21. Betrachten wir den Algorithmus `calcul(n)`, wobei n eine natürliche Zahl ist ($2 < n < 10$). Der Algorithmus `zerosMatrice(n)` gibt eine quadratische Matrix mit n Zeilen und n Spalten zurück, wobei alle Elemente gleich 0 sind. Der Algorithmus `zerosVector(n)` gibt einen Vektor mit n Elementen zurück, die alle gleich 0 sind.

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

```

Welche Werte enthält der nach dem Aufruf von `calcul(n)` zurückgegebene Vektor?

- A. Die Potenzen von 0 bis $n - 2$ der Zahl 11
- B. Die ersten n Vielfachen der Zahl 11
- C. Elemente aus der Folge $2 * x[i] - 1$ ($i > 0$), wobei die Folge x wie folgt definiert ist: $x[1] = 1, x[2] = 6, x[k] = x[k - 1] * 10 + 1$ (für $k > 2$)
- D. Elemente aus der Folge x , wobei die Folge x wie folgt definiert ist: $x[1] = 1, x[k] = x[k - 1] * 11$ (für $k > 1$)

22. Sei der Algorithmus `genereaza(n, a, k, p)`, wobei n, k und p natürliche Zahlen sind ($1 \leq n \leq 12, 0 \leq p \leq n, 1 \leq k \leq 10^6$) und a ein Vektor aus n natürlichen Zahlen ist ($a[1], a[2], \dots, a[n], 0 \leq a[i] \leq 10^4$, für $i = 1, 2, \dots, n$).

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

```

Was ist der Wert des Vektors a nach dem Aufruf von `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. Betrachten wir den Algorithmus $f(n, a)$, wobei n eine natürliche Zahl ist ($2 \leq n \leq 10^6$) und a ein Vektor ($a[1]$, $a[2]$, ..., $a[n]$, $1 \leq a[i] \leq 10^6$ für $i = 1, 2, \dots, n$) mit n natürlichen Elementen ist. Der Algorithmus $\text{zerosMat}(n)$ gibt eine Matrix mit n Zeilen und n Spalten zurück, deren Elemente alle gleich 0 sind.

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Wenn der Vektor a aufsteigend sortiert ist, dann erscheint nach dem Aufruf von $f(n, a)$ der Wert 2 n -mal in der Matrix.
- B. Wenn der Vektor a absteigend sortiert ist, dann erscheint nach dem Aufruf von $f(n, a)$ der Wert 2 nur einmal in der Matrix.
- C. Wenn der Vektor a aus identischen Werten besteht, dann enthält nach dem Aufruf von $f(n, a)$ jede Zeile der Matrix mindestens einen Wert ungleich Null.
- D. Nach dem Aufruf von $f(5, [1, 1, 4, 6, 9])$ befinden sich in der letzten Zeile der Matrix nur Werte gleich 2.

24. Wir haben Perlen in drei Farben (die wir mit den Zeichenfolgen "0", "1" und "2" bezeichnen) und, mithilfe des Algorithmus $\text{colier}(n)$, wobei n eine natürliche Zahl ist ($4 \leq n \leq 3 \cdot 2^{10}$), möchten wir eine Perlenkette der Länge n konstruieren, die keine zwei identischen aufeinanderfolgenden Teilsequenzen enthält. Beispielsweise, für $n = 4$ sind weder "0211" (die Sequenz "1" würde zweimal hintereinander auftreten) noch "0202" (die Sequenz "02" würde zweimal hintereinander auftreten) gute Lösungen. Eine mögliche Lösung ist "0201". Der Operator $+$ steht für die Verkettung von Zeichenfolgen, und der Algorithmus $\text{len}(s)$ gibt die Länge der Zeichenfolge s zurück.

```

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

```

Welche der folgenden Aussagen sind wahr?

- A. Der Algorithmus $\text{colier}(n)$ gibt die Darstellung einer Perlenkette gemäß den Anforderungen zurück, unabhängig vom Wert von n .
- B. Nach dem Aufruf von $\text{colier}(8)$ wird die Darstellung "01202101" zurückgegeben.
- C. Bei den Aufrufen $\text{colier}(10)$ und $\text{colier}(12)$ wird der Algorithmus $\text{extinde}(cuv)$ gleich oft aufgerufen.
- D. Der Algorithmus $\text{colier}(n)$ gibt die Darstellung einer Perlenkette gemäß den Anforderungen zurück, wenn und nur wenn n von der Form $3 \cdot 2^k$ ($1 \leq k \leq 10$) ist.

BABEŞ-BOLYAI UNIVERSITÄT
FAKULTÄT FÜR MATHEMATIK UND INFORMATIK

Mathe-Info-Wettbewerb - 21. März 2026

Schriftliche Prüfung in Informatik

PUNKTEANZAHL & LÖSUNGEN

ANFANGSPUNKTEANZAHL: 10 punkte

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