BABEŞ-BOLYAI UNIVERSITÄT
FAKULTÄT FÜR MATHEMATIK UND INFORMATIK

Mathe-Info-Wettbewerb - 26. März 2023

Schriftliche Prüfung in Informatik

WICHTIGER HINWEIS:
Falls nicht anders angegeben:

- Wir gehen davon aus, dass alle arithmetischen Operationen mit unbegrenzten Datentypen durchgeführt werden (kein *Über-/Unterlauf*).
- die Indexnummerierung aller Zeichenketten beginnt bei 1
- alle Einschränkungen beziehen sich auf die aktuellen Parameterwerte zum Zeitpunkt des ersten Aufrufs.

1. Gegeben sei der Algorithmus `f(a, b)`, wobei a und b natürliche Zahlen ungleich Null sind ($1 \leq a, b \leq 10^9$).

```
1: Algorithm f(a, b):
2:     If a = b then
3:         Return a
4:     EndIf
5:     If a > b then
6:         Return f(a - b, b)
7:     EndIf
8:     Return f(a, b - a)
9: EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

A. Nach dem Aufruf `f(2000, 21)` gibt der Algorithmus 1 zurück.
B. Im Fall des Aufrufs `f(2000, 21)` bricht der Algorithmus seine Ausführung aufgrund der Bedingung in Zeile 2 nicht ab.
C. Damit der Algorithmus den größten gemeinsamen Teiler von a und b zurückgibt, sollte Zeile 8 in `Return f(b - a, b)` geändert werden.
D. Damit der Aufruf des Algorithmus `f(2000, 21)` den Wert 1 zurückgibt, sollte Zeile 8 in `Return f(b - a, b - a)` geändert werden.

2. Gegeben sei die folgende Algorithmus Sequenz, wobei a ein Vektor bestehend aus n natürlichen Zahlen ist ($a[1], a[2], ..., a[n]$, $1 \leq a[i] \leq 10^4$, für $i = 1, 2, ..., n$) und n eine natürliche, von Null verschiedene Zahl ist ($1 \leq n \leq 10^4$):

```
For i ← 1, n - 1 execute
    poz ← i
    For j ← i + 1, n execute
        If a[j] < a[poz] then
            poz ← j
        EndIf
    EndFor
    If poz ≠ i then
        temp ← a[i]
        a[i] ← a[poz]
        a[poz] ← temp
    EndIf
EndFor
```

Welche der folgenden Aussagen sind wahr, wenn i zu 2 wird?

A. $a[1] \leq a[k]$ für jedes $k \in \{1, 2, ..., n\}$
B. $a[n] \leq a[k]$ für jedes $k \in \{1, 2, ..., n\}$
C. $a[1] \geq a[k]$ für jedes $k \in \{1, 2, ..., n\}$
D. $a[k] \leq a[k + 1]$ für jedes $k \in \{1, 2, ..., n - 1\}$

3. Gegeben sei der Algorithmus `alg(n)`, wobei n eine natürliche Zahl ist ($0 \leq n \leq 10^9$).

```
Algorithm alg(n):
    If n MOD 2 = 0 then
        Return n + alg(n - 1)
    Else
        Return n
    EndIf
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

A. Wenn $n = 4$ ist, liefert der Algorithmus den Wert 7.
B. Der Algorithmus gibt die Summe der natürlichen Zahlen kleiner als n zurück.
C. Der Algorithmus liefert die Summe der natürlichen Zahlen, die kleiner oder gleich n sind.
D. Wenn $n = 7$ ist, liefert der Algorithmus den Wert 7.

4. Gegeben sei der Algorithmus `f(nr)`, wobei ***nr*** eine ganze Zahl ist ($-10^4 \le$ ***nr*** $\le 10^4$).

```
Algorithm f(nr):
    If nr < 0 then
        Return f(-nr)
    EndIf
    If (nr = 0) OR (nr = 7) then
        Return 1
    EndIf
    If nr < 10 then
        Return 0
    EndIf
    Return f((nr DIV 10) - 2 * (nr MOD 10))
EndAlgorithm
```

Für welche Werte von ***nr*** liefert der Algorithmus den Wert 1?

A. 308
B. -7
C. 7098
D. 57

5. Gegeben sei der Algorithmus `afis(n)`, wobei ***n*** eine natürliche Zahl ist ($1 \le$ ***n*** $\le 10^4$):

```
Algorithm afis(n):
    If n > 9 then
        If n MOD 2 = 0 then
            afis(n DIV 100)
            Write n MOD 10, " "
        Else
            afis(n DIV 10)
        EndIf
    EndIf
EndAlgorithm
```

Für welche der folgenden Anrufe werden die Werte **`2 4`**, in dieser Reihenfolge angezeigt?

A. `afis(1234)`
B. `afis(1224)`
C. `afis(4224)`
D. `afis(4321)`

6. Gegeben sei der Algorithmus `Afisare(a)`, wobei ***a*** eine natürliche Zahl ist ($1 \le$ ***a*** $\le 10^4$).

```
Algorithm Afișare(a):
    If a < 9000 then
        Write a, " "
        Afișare(3 * a)
        Write a, " "
    EndIf
EndAlgorithm
```

Was wird bei dem Aufruf von `Afisare(1000)` angezeigt?

A. `1000 3000 9000 9000 3000 1000`
B. `1000 3000 9000 3000 1000`
C. `1000 3000 3000 1000`
D. `1000 3000 9000`

7. Gegeben sei der Algorithmus `f(n, x)`, wobei ***n*** eine natürliche Zahl ist ($3 \le$ ***n*** $\le 10^4$) und ***x*** ein Vektor bestehend aus ***n*** natürlichen Zahlen ist (***x***[1], ***x***[2], ..., ***x***[***n***], $1 \le$ ***x***[***i***] $\le 10^4$, für ***i*** = 1, 2, ..., ***n***):

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

Bei welchem der folgenden Aufrufe gibt der Algorithmus *True* zurück?

A. `f(3, [10, 15, 25])`
B. `f(4, [0, 0, 0, 0])`
C. `f(5, [100, 535, 635, 1170, 1805])`
D. `f(4, [0, 1, 0, 1])`

8. Was ist das Ergebnis der Umrechnung der Dezimalzahl $2^{10} - 2^5 - 1$ zur Basis 2?

A. `1111011111`
B. `1010011001`
C. `1000011001`
D. Keine der Antworten A, B, C

9. Gegeben seien die Algorithmen `one(a, b)` und `two(n, m)`, bei denen die Eingabeparameter ***a***, ***b***, ***n*** und ***m*** natürliche Zahlen sind ($2 \leq \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{n}, \boldsymbol{m} \leq 10^6$, $\boldsymbol{n} < \boldsymbol{m}$).

```
Algorithm one(a, b):
    s ← 0
    For i ← 1, a execute
        If a MOD i = 0 then
            s ← s + i
        EndIf
    EndFor
    For i ← 1, b execute
        If b MOD i = 0 then
            s ← s + i
        EndIf
    EndFor
    Return s
EndAlgorithm
```

```
Algorithm two(n, m):
    For i ← n, m execute
        If one(i, i) = 2 * i + 2 then
            Write i, " "
        EndIf
    EndFor
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

A. Der Algorithmus `two(n, m)` zeigt nichts an, unabhängig vom Wert der Eingabeparameter.
B. Der Algorithmus `two(n, m)` zeigt die Primzahlen im Bereich [***n*** , ***m***] an.
C. Der Algorithmus `two(n, m)` zeigt durch 2 teilbare Zahlen aus dem Bereich [***n*** , ***m***] an.
D. Keine der anderen Varianten ist korrekt.

10. Gegeben sei der Algorithmus `decide(n, x)`, wobei ***n*** eine natürliche Zahl ungleich Null ist ($1 \leq \boldsymbol{n} \leq 10^4$) und ***x*** ein Vektor bestehend aus ***n*** natürlichen Zahlen ist ($\boldsymbol{x}[1], \boldsymbol{x}[2], ..., \boldsymbol{x}[\boldsymbol{n}]$, $0 \leq \boldsymbol{x}[\boldsymbol{i}] \leq 100$, für $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$).

```
Algorithm decide(n, x):
    i ← 1
    j ← n
    While i < j AND x[i] = x[j] execute
        i ← i + 1
        j ← j - 1
    EndWhile
    If i ≥ j then
        Return True
    Else
        Return False
    EndIf
EndAlgorithm
```

Wann gibt der Algorithmus `decide(n, x)` *True* zurück?

A. Immer
B. Wenn die Elemente des Vektors ***x*** [1, 2, 3] sind
C. Wenn die Elemente des Vektors ***x*** [1, 1, 1] sind
D. Wenn die Elemente des Vektors ***x*** ein Palindrom bilden, d. h. $\boldsymbol{x}[\boldsymbol{i}] = \boldsymbol{x}[\boldsymbol{n} - \boldsymbol{i} + 1]$ für jedes $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$

11. Gegeben sei der Algorithmus `alg(a, b)`, wobei ***a*** und ***b*** natürliche Zahlen sind ($1 \leq \boldsymbol{a}, \boldsymbol{b} \leq 10^3$).

```
Algorithm alg(a, b):
    If b = 0 then
        Return 1
    Else
        Return a * alg(a, b - 1)
    EndIf
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

A. Für den Aufruf `alg(2, 3)` liefert der Algorithmus 7.
B. Bei dem Aufruf `alg(2, 3)` wird der Algorithmus unter Berücksichtigung des ersten Aufrufs 4 Mal aufgerufen.
C. Der Algorithmus berechnet den Wert von $\boldsymbol{a}^{\boldsymbol{b}-1}$ und gibt ihn zurück.
D. Der Algorithmus berechnet den Wert von $\boldsymbol{a}^{\boldsymbol{b}}$ und gibt ihn zurück.

12. Gegeben sei der Algorithmus `ceFace(a, b)`, wobei $\boldsymbol{a}$ und $\boldsymbol{b}$ natürliche Zahlen sind ($1 < \boldsymbol{a}, \boldsymbol{b} \leq 10^5$). Der Algorithmus `prim(n)` gibt *True* zurück, wenn die Zahl $\boldsymbol{n} > 1$ eine Primzahl ist, und andernfalls *False*.

```
Algorithm ceFace(a, b):
    If prim(a) = True then
        Write a, " "
    Else
        If prim(b) ≠ True then
            ceFace(a, b + 1)
        Else
            If b > a then
                Write a, " "
            Else
                If a MOD b = 0 then
                    Write b, " "
                    ceFace(a DIV b, b)
                Else
                    ceFace(a, b + 1)
                Endif
            EndIf
        EndIf
    EndIf
EndAlgorithm
```

Was wird bei dem Aufruf `ceFace(100, 2)` angezeigt?

A. `2 5 5 5`
B. `5 5 2 2`
C. `2 2 2 5`
D. `2 2 5 5`

13. Gegeben sei der Algorithmus `f(n, p)`, wobei $\boldsymbol{n}$ eine natürliche, von Null verschiedene Zahl ($1 \leq \boldsymbol{n} \leq 10^9$) und $\boldsymbol{p}$ eine natürliche Zahl ($0 \leq \boldsymbol{p} \leq 10^9$) ist:

```
Algorithm f(n, p):
    If n ≤ 9 then
        If n MOD 2 = 0 then
            Return 10 * p + n
        Else
            Return p
        EndIf
    Else
        If n MOD 2 = 0 then
            p ← p * 10 + n MOD 10
        EndIf
        Return f(n DIV 10, p)
    EndIf
EndAlgorithm
```

Welcher der folgenden Aufrufe ergibt 22?

A. `f(23572, 0)`
B. `f(23527, 0)`
C. `f(2, 0)`
D. `f(1242, 0)`

14. Gegeben sei der Algorithmus `cifre(n)`, wobei $\boldsymbol{n}$ eine natürliche Zahl ist ($0 \leq \boldsymbol{n} \leq 10^3$).

```
Algorithm cifre(n):
    If n ≥ 1 then
        If (n * 5) MOD 10 = 0 then
            Return cifre(n DIV 10)
        Else
            Return n MOD 10
        EndIf
    Else
        Return -1
    EndIf
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

A. Der Algorithmus liefert immer eine Zahl kleiner als 10.
B. Der Algorithmus gibt -1 zurück, wenn und nur wenn der Anfangswert von $\boldsymbol{n}$ 0 ist.
C. Für $\boldsymbol{n} \geq 1$ liefert der Algorithmus die niedrigstwertige Ziffer von $\boldsymbol{n}$, die ungerade ist, oder -1, wenn es sie nicht gibt.
D. Für $\boldsymbol{n} \geq 1$ liefert der Algorithmus die höchstwertige Ziffer von $\boldsymbol{n}$, die ungerade ist, oder -1, wenn es sie nicht gibt.

15. Gegeben sei der Algorithmus `ceFace(a, b)`, wobei ***a*** und ***b*** zwei natürliche Zahlen sind ($0 \leq \boldsymbol{a}, \boldsymbol{b} \leq 10^6$).

```
Algorithm ceFace(a, b):
    c ← 0
    p ← 1
    While a * b ≠ 0 execute
        If (a MOD 10) = (b MOD 10) then
            c ← (a MOD 10) * p + c
        Else
            If (a MOD 10) < (b MOD 10) then
                c ← ((b MOD 10 - a MOD 10) DIV 2) * p + c
            Else
                c ← ((a MOD 10 - b MOD 10) DIV 2) * p + c
            EndIf
        EndIf
        p ← p * 10
        a ← a DIV 10
        b ← b DIV 10
    EndWhile
    Return c
EndAlgorithm
```

Welche der folgenden Aussagen sind richtig?

A. Wenn ***a*** = 0 und ***b*** = 0 ist, liefert der Algorithmus 1.
B. Wenn ***a*** = 11 und ***b*** = 111 ist, liefert der Algorithmus 11.
C. Wenn ***a*** = 5678 und ***b*** = 5162738 ist, liefert der Algorithmus 1024.
D. Wenn ***a*** = 112233 und ***b*** = 331122 ist, liefert der Algorithmus 110000.

16. Gegeben seien die Algorithmen `ceva(n, m)` und `altceva(n, m)`, wobei ***n*** und ***m*** natürliche Zahlen ungleich Null sind ($1 \leq \boldsymbol{n}, \boldsymbol{m} \leq 10^{12}$ und $\boldsymbol{m} \leq \boldsymbol{n}$).

```
Algorithm ceva(n, m):
    nc ← n
    mc ← m
    While nc > 0 AND mc > 0 execute
        nc ← nc DIV 10
        mc ← mc DIV 10
    EndWhile
    If nc = mc then
        Return True
    Else
        Return False
    EndIf
EndAlgorithm
```

```
Algorithm altceva(n, m):
    c ← 0
    While ceva(n, m) = False execute
        m ← m * 10 + 1
        c ← c + 1
    EndWhile
    Write n, " ", m
    Return c
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

A. Die Zeitkomplexität des Algorithmus `ceva(n, m)` ist $O(\log \boldsymbol{m})$.
B. Der Algorithmus `altceva(n, m)` gibt 0 zurück, wenn und nur wenn ***n*** = ***m***.
C. Wir brauchen die Vorbedingung $\boldsymbol{m} \leq \boldsymbol{n}$ aus der Anweisung, denn wenn $\boldsymbol{m} > \boldsymbol{n}$ ist, tritt der Algorithmus `altceva(n, m)` immer in einer endlos Schleife ein.
D. Es gibt Zahlen ***n*** und ***m*** (die die Vorbedingung erfüllen), für die der Algorithmus `altceva(n, m)` zwei Werte in aufsteigender Reihenfolge anzeigt.

17. Gegeben sei der Algorithmus `h(s, d, A)`, wobei $\boldsymbol{s}$ und $\boldsymbol{d}$ natürliche Zahlen ungleich Null sind ($1 \leq \boldsymbol{s}, \boldsymbol{d} \leq 10^3$) und $\boldsymbol{A}$ ein Vektor bestehend aus $\boldsymbol{n}$ natürlichen Zahlen ungleich Null ist ($\boldsymbol{A}[1], \boldsymbol{A}[2], ..., \boldsymbol{A}[\boldsymbol{n}]$, $1 \leq \boldsymbol{A}[\boldsymbol{i}] \leq 10^3$, für $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$).

```
Algorithm h(s, d, A):
    If s = d then
        x ← A[s]
        y ← x MOD 10
        x ← x DIV 10
        While x > 0 execute
            z ← x MOD 10
            If z - y ≠ 2 then
                Return 0
            EndIf
            y ← z
            x ← x DIV 10
        EndWhile
        Return 1
    Else
        Return h(s, (s + d) DIV 2, A) + h((s + d) DIV 2 + 1, d, A)
    EndIf
EndAlgorithm
```

Für welche Werte der Zahl $\boldsymbol{n}$ und des Vektors $\boldsymbol{A}$ wird der Aufruf `h(1, n, A)` den Wert 5 liefern?

A. $\boldsymbol{n} = 7$, $\boldsymbol{A}$ = (20, 53, 10, 42, 31, 131, 42)
B. $\boldsymbol{n} = 10$, $\boldsymbol{A}$ = (420, 75, 68, 86, 97, 975, 53, 64, 24, 57)
C. $\boldsymbol{n} = 10$, $\boldsymbol{A}$ = (402, 75, 6, 86, 7, 9, 35, 46, 24, 57)
D. $\boldsymbol{n} = 10$, $\boldsymbol{A}$ = (642, 97, 6, 64, 7, 9, 75, 4, 53, 31)

18. Gegeben sei der Algorithmus `f(a, x)`, wobei $\boldsymbol{x}$ eine natürliche Zahl ungleich Null ($1 \leq \boldsymbol{x} \leq 10^4$) und $\boldsymbol{a}$ ein Vektor aus 10 natürlichen Zahlen ungleich Null ($\boldsymbol{a}[1], \boldsymbol{a}[2], ..., \boldsymbol{a}[10]$) ist.

```
Algorithm f(a, x):
    i ← 1, j ← 10
    k ← 1
    While a[k] ≠ x AND i < j execute
        k ← (i + j) DIV 2
        If a[k] < x then
            i ← k
        Else
            j ← k
        EndIf
    EndWhile
    If a[k] = x then
        Return True
    Else
        Return False
    EndIf
EndAlgorithm
```

Für welche der folgenden Eingaben wird der Algorithmus unendlich oft wiederholt?

A. $\boldsymbol{a}$ = [3, 3, 3, 3, 3, 3, 3, 3, 3, 3] und $\boldsymbol{x} > 3$
B. $\boldsymbol{a}$ = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] und $\boldsymbol{x} < 10$
C. $\boldsymbol{a}$ = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20] und $1 < \boldsymbol{x} < 20$, $\boldsymbol{x}$ - ungerade Zahl
D. $\boldsymbol{a}$ = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20] und $1 < \boldsymbol{x} < 20$, $\boldsymbol{x}$ - gerade Zahl

19. Gegeben sei der Algorithmus `f(a)`, wobei $\boldsymbol{a}$ eine natürliche Zahl ist ($1 \leq \boldsymbol{a} \leq 10^9$).

```
Algorithm f(a):
    x ← a MOD 10
    If x = a then
        If x MOD 2 = 0 then
            Return a
        Else
            Return 0
        EndIf
    EndIf
    If x MOD 2 = 0 then
        Return 10 * f(a DIV 10) + x
    EndIf
    Return f(a DIV 10)
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

A. Für $\boldsymbol{a}$ = 253401976 wird der Algorithmus `f(a)` 8 Mal aufgerufen. Der erste Aufruf wird ebenfalls gezählt.
B. Für $\boldsymbol{a}$ = 253401976 wird der Algorithmus `f(a)` 9 Mal aufgerufen. Der erste Aufruf wird ebenfalls gezählt.
C. Für $\boldsymbol{a}$ = 253401976 ist das Ergebnis des Algorithmus 2406.
D. Das Ergebnis des Algorithmus `f(a)` für die Zahl $\boldsymbol{a}$, die nur aus geraden Ziffern besteht, ist gleich $\boldsymbol{a}$.

20. Gegeben sei der Algorithmus `A(k)`, wobei der Parameter ***k*** eine natürliche von Null verschiedene Zahl ist ($1 \leq \boldsymbol{k} \leq 10^9$).

```
Algorithm A(k):
    gr ← (-1 + radical(1 + 8 * k)) / 2
    If gr = [gr] then
        p ← gr
    Else
        p ← [gr] + 1
    EndIf
    Return p - (k – p * (p - 1) DIV 2 - 1)
EndAlgorithm
```

- Mit [***gr***] wird der ganze Teil von ***gr*** bezeichnet.
- Der Algorithmus `radical(x)` liefert den Wert des Radikals von ***x***.
- Der Operator / steht für die Division von reellen Zahlen, zum Beispiel: 7 / 2 = 3,5

Welche der folgenden Aussagen sind richtig?

A. Der unten definierte Algorithmus `A1(k)` ist äquivalent zu Algorithmus `A(k)`.

```
Algorithm A1(k):
    c ← 0
    i ← 1
    While c < k execute
        j ← 1
        While j ≤ i execute
            If c < k then
                c ← c + 1
                If c = k then
                    Return j
                Else
                    j ← j + 1
                EndIf
            Else
                Return j
            EndIf
        EndWhile
        i ← i + 1
    EndWhile
EndAlgorithm
```

B. Der unten definierte Algorithmus `A2(k)` ist äquivalent zu Algorithmus `A(k)`.

```
Algorithm A2(k):
    c ← 0
    i ← 1
    While c < k execute
        j ← i
        While j ≥ 1 execute
            If c < k then
                c ← c + 1
                If c = k then
                    Return j
                Else
                    j ← j - 1
                EndIf
            Else
                Return j
            EndIf
        EndWhile
        i ← i + 1
    EndWhile
EndAlgorithm
```

C. Algorithmus `A(k)` gibt den ***k-ten*** Term der Zeichenkette zurück, die durch Verkettung von Zeichenketten der Form [1, 2, ..., ***i***] für jedes ***i*** = 1, 2, ..., ***k*** in dieser Reihenfolge gebildet wird (d.h. Zeichenkette [1, 1, 2, 1, 2, 3, 1, 2, 3, 4, ...]).

D. Algorithmus `A(k)` gibt den ***k-ten*** Term der Zeichenkette zurück, die durch Verkettung von Zeichenketten der Form [***i***, ..., 2, 1] für jedes ***i*** = 1, 2, ..., ***k*** in dieser Reihenfolge gebildet wird (d.h. Zeichenkette [1, 2, 1, 3, 2, 1, 4, 3, 2, 1, ...]).

21. Gegeben sei der Algorithmus `ceFace(a, long)`, wobei ***long*** eine natürliche Zahl ist ($1 \leq \boldsymbol{long} \leq 10^5$) und ***a*** ein Vektor mit ***long*** ganzzahligen Elementen ist (***a***[1], ***a***[2], ..., ***a***[***long***]). Es gibt mindestens eine positive Zahl im Vektor ***a.***

```
Algorithm ceFace(a, long):
    value1 ← 0
    value2 ← 0
    For i ← 1, long execute
        value2 ← value2 + a[i]
        If value1 < value2 then
            value1 ← value2
        EndIf
        If value2 < 0 then
            value2 ← 0
        EndIf
    EndFor
    Return value1
EndAlgorithm
```

Wenn man weiß, dass eine Teilsequenz des Vektors ***x*** = [***x***[1], ***x***[2], ..., ***x***[***n***]] aus Elementen des Vektors ***x*** besteht, die aufeinanderfolgende Positionen einnehmen (z.B. ***y*** = [***x***[3], ***x***[4], ***x***[5], ***x***[6]] ist eine Teilsequenz des Vektors ***x*** der Länge 4), welche der folgenden Aussagen sind dann wahr?

A. Wenn der Vektor ***a*** nur eine positive Zahl enthält, gibt der Algorithmus seinen Wert zurück.

B. Der Algorithmus gibt die Länge einer der Teilsequenzen zurück, die die maximale Summe im Vektor ***a*** hat.

C. Der Algorithmus gibt die Summe einer der Teilsequenzen zurück, die die maximale Summe im Vektor ***a*** haben.

D. Der Algorithmus liefert die Summe der positiven Zahlen an aufeinanderfolgenden Stellen am Ende des Vektors ***a***.

22. Gegeben sei der Algorithmus `ceFace(sir, a, b)`, wobei ***sir*** ein Vektor ist, der aus ***n*** ($1 \leq n \leq 100$) verschiedenen natürlichen Zahlen besteht, die in aufsteigender Reihenfolge angeordnet sind (***sir***[1], ***sir***[2], ..., ***sir***[***n***]), ***a*** und ***b*** sind natürliche Zahlen ($1 \leq a, b \leq n$).

```
Algorithm ceFace(sir, a, b):
    If a > b then
        Return a
    EndIf
    c ← a + (b - a) DIV 2
    If sir[c] = c then
        Return ceFace(sir, c + 1, b)
    Else
        Return ceFace(sir, a, c - 1)
    EndIf
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend, wenn man davon ausgeht, dass der erste Aufruf `ceFace(sir, 1, n)` lautet?

A. Wenn der Vektor ***sir*** aus den ersten ***n*** verschiedenen natürlichen Zahlen besteht, gibt der Algorithmus den Wert ***n*** + 1 zurück.
B. Der Algorithmus gibt die größte Position ***p*** kleiner oder gleich ***n*** **DIV** 2 zurück, für die ***sir***[***p***] = ***p*** , oder 1, wenn es keine solche Position gibt ($1 \leq p \leq n$).
C. Der Algorithmus gibt die größte Position ***p*** kleiner oder gleich ***n*** **DIV** 2 zurück, für die ***sir***[***p***] ≠ ***p*** , oder ***n*** + 1, wenn es keine solche Position gibt ($1 \leq p \leq n$).
D. Der Algorithmus liefert die kleinste natürliche Zahl ungleich Null, die nicht im Vektor ***sir*** vorkommt.

23. Gegeben sei der Algorithmus `ceFace(s, x, c, y, n, m, k)`, wobei ***s*** eine Zeichenkette (***s***[1], ***s***[2], ..., ***s***[***x***]) der Länge ***x*** ist und ***c*** eine Zeichenkette (***c***[1], ***c***[2], ..., ***c***[***y***]) der Länge ***y.*** Die Bezeichner ***x***, ***y***, ***n***, ***m*** und ***k*** speichern natürliche Zahlen, die nicht Null sind ($1 \leq x, y, n, m, k \leq 100$).

```
1. Algorithm ceFace(s, x, c, y, n, m, k):
2.     If (n ≥ 0) AND (m ≥ 0) AND (n ≤ x) AND (m ≤ y) then
3.         If k MOD 2 = 0 then
4.             Write s[(n + k) MOD x + 1]
5.             ...
6.             ceFace(s, x, c, y, n - 1, m, k)
7.         EndIf
8.         If k MOD 2 = 1 then
9.             Write c[(m + k) MOD y + 1]
10.            ...
11.            ceFace(s, x, c, y, n, m - 1, k)
12.        EndIf
13.    EndIf
14. EndAlgorithm
```

Wir wollen nach dem Aufruf `ceFace("+-", 2, "123", 3, 2, 2, 4)` einen gültigen arithmetischen Ausdruck erhalten (d. h. einen arithmetischen Ausdruck, der eine Abwechslung von einem Operator und einem Operanden darstellt; er kann mit einem der Operatoren '+' oder '-' beginnen und muss mit einem Operanden enden). Welche der folgenden Aussagen sind **NICHT** zutreffend?

A. Die Zeilen 5 und 10 können mit der Anweisung `k ← k + 7` ergänzt werden.
B. Zeile 5 kann mit der Anweisung `k ← k + 2 und` Zeile 10 mit der Anweisung `k ← k + 5` ergänzt werden.
C. Die Zeilen 5 und 10 können mit der Anweisung `k ← k + 2` ergänzt werden.
D. Zeile 5 kann mit der Anweisung `k ← k + 7` und Zeile 10 mit der Anweisung `k ← k - 1` ergänzt werden.

24. Man betrachte die natürliche Zahl ***n*** ($1 \leq n \leq 50$) und den Vektor ***x*** bestehend aus ***n*** ganzzahligen Elementen (***x***[1], ***x***[2], ..., ***x***[***n***]). Welche der folgenden Aussagen sind wahr, unabhängig vom Wert von ***n*** und den Werten der Elemente des Vektors?

A. Es gibt eine natürliche Zahl ***k*** ($1 \leq k \leq n$), so dass die Summe ***x***[1] + ***x***[2] + ... + ***x***[***k***] durch ***n*** teilbar ist.
B. Es gibt (***i***, ***j***), $0 \leq i < j \leq n$, so dass die Summe ***x***[***i*** + 1] + ***x***[***i*** + 2] + ... + ***x***[***j***] durch ***n*** teilbar ist.
C. Keine der Aussagen A und B ist zutreffend.
D. Wenn man weiß, dass eine Teilsequenz des Vektors ***x*** = [***x***[1], ***x***[2], ...[x[1], x[2], ..., ***x***[***n***]] aus Elementen des Vektors ***x*** besteht, die aufeinanderfolgende Positionen einnehmen (z. B. ist ***y*** = [***x***[3], ***x***[4], ***x***[5], ***x***[6]] eine Teilsequenz des Vektors ***x*** der Länge 4), dann gibt es eine natürliche Zahl ***k***, ($1 \leq k \leq n$), so dass in dem Vektor ***x*** eine Teilsequenz von ***k*** Elementen ($1 \leq k \leq n$) existiert, deren Summe durch ***n*** teilbar ist.

BABEŞ-BOLYAI UNIVERSITÄT

FAKULTÄT FÜR MATHEMATIK UND INFORMATIK

Mathe-Info-Wettbewerb - 26. März 2023

Schriftliche Prüfung in Informatik

PUNKTEANZAHL & LÖSUNGEN

ANFANGSPUNKTEANZAHL: 10 punkte

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