BABEŞ-BOLYAI-UNIVERSITÄT
FAKULTÄT FÜR MATHEMATIK UND INFORMATIK

Zulassung - 9. September 2025

Schriftliche Prüfung in Informatik

WICHTIGE NOTE:
Sofern nicht anders angegeben:

- Alle arithmetischen Operationen werden mit unbegrenzten Datentypen durchgeführt (kein *Über-/Unterlauf*).
- Die Indexnummerierung aller Vektoren, Matrizen und Strings beginnt bei 1.
- Alle Beschränkungen beziehen sich auf die aktuellen Parameterwerte zum Zeitpunkt des ersten Aufrufs.
- Eine Teilfolge eines Vektors oder einer Zeichenkette besteht aus Elementen, die aufeinanderfolgende Positionen im Vektor bzw. in der Zeichenkette einnehmen.
- Treten mehrere aufeinanderfolgende Zuweisungsanweisungen in derselben Zeile auf, werden sie durch "; " voneinander getrennt.

1. Gegeben sei der Algorithmus `cauta(a, n, b, m)`, wobei ***a*** und ***b*** zwei Zeichenketten mit ***n*** bzw. ***m*** Zeichen sind (***a***[1], ***a***[2], ..., ***a***[***n***], ***b***[1], ***b***[2], ..., ***b***[***m***], $1 \leq n, m \leq 100$ und $m \leq n$). Welche der folgenden Implementierungen des Algorithmus `cauta(a, n, b, m)` gibt die Position in der Zeichenkette ***a*** zurück, ab der die Zeichenkette ***b*** zum ersten Mal als Teilfolge in der Zeichenkette ***a*** erscheint, oder -1, wenn die Zeichenkette ***b*** nicht in der Zeichenkette ***a*** erscheint?

A.
```
Algorithm cauta(a, n, b, m):
    i ← 1
    While i < n – m + 2 execute
        j ← 1
        While j ≤ m AND a[i + j - 1] = b[j] execute
            j ← j + 1
        EndWhile
        If j > m then
            Return i
        EndIf
        i ← i + 1
    EndWhile
    Return -1
EndAlgorithm
```

B.
```
Algorithm cauta(a, n, b, m):
    If n = m then
        Return 1
    EndIf
    For i ← 1, n - m + 1 execute
        If a[i] = b[i] AND a[i + m - 1] = b[m] then
            Return i
        EndIf
    EndFor
    Return -1
EndAlgorithm
```

C.
```
Algorithm cauta(a, n, b, m):
    For i ← 1, n - m + 1 execute
        k ← True; j ← 1
        While k AND j ≤ m execute
            If a[i + j - 1] ≠ b[j] then
                k ← False
            EndIf
            j ← j + 1
        EndWhile
        If k then
            Return i
        EndIf
    EndFor
    Return -1
EndAlgorithm
```

D.
```
Algorithm cauta(a, n, b, m):
    i ← 1
    While i ≤ n - m execute
        j ← 1
        While j ≤ m AND a[i + j - 1] = b[j] execute
            j ← j + 1
        EndWhile
        If j > m then
            Return i
        EndIf
        i ← i + 1
    EndWhile
    Return -1
EndAlgorithm
```

2. Gegeben sei der Algorithmus `S(n)`, wobei ***n*** eine ganze Zahl ist ($0 \leq n \leq 10^4$).

```
Algorithm S(n):
    If n = 0 then
        Return 0
    EndIf
    Return n + S(n DIV 2)
EndAlgorithm
```

Welche der folgenden Aussagen sind richtig?

A. Die Zeitkomplexität des Algorithmus ist $O(\log n)$.
B. Der Algorithmus berechnet die Summe der ersten ***n*** natürlichen Zahlen.
C. Nach dem Aufruf `S(4)` liefert der Algorithmus den Wert 12.
D. Der Ausdruck `S(2) + S(13) = S(7) + S(8)` hat den Wert *True*.

3. Welche der folgenden Implementierungen des Algorithmus `gcd(a, b)` liefert den größten gemeinsamen Teiler der natürlichen Zahlen $\boldsymbol{a}$ und $\boldsymbol{b}$ ($1 \leq \boldsymbol{a}, \boldsymbol{b} \leq 100$)?

A.
```
Algorithm gcd(a, b):
    If b = 0 then
        Return b
    EndIf
    If a > b then
        Return gcd(a - b, b)
    Else
        Return gcd(a, b - a)
    EndIf
EndAlgorithm
```

B.
```
Algorithm gcd(a, b):
    If b = 0 then
        Return 0
    EndIf
    Return gcd(b, a MOD b)
EndAlgorithm
```

C.
```
Algorithm gcd(a, b):
    While b > 0 execute
        c ← a; a ← b; b ← c MOD a
    EndWhile
    Return a
EndAlgorithm
```

D.
```
Algorithm gcd(a, b):
    While a MOD b > 0 execute
        c ← a; a ← b; b ← c MOD a
    EndWhile
    Return b
EndAlgorithm
```

4. Gegeben sei der Algorithmus `f(n, x, k)`, wobei $\boldsymbol{n}$ eine natürliche Zahl ist ($1 \leq \boldsymbol{n} \leq 10^4$), $\boldsymbol{x}$ ein Vektor mit $\boldsymbol{n}$ ganzzahligen Elementen ist ($\boldsymbol{x}[1], \boldsymbol{x}[2], ..., \boldsymbol{x}[\boldsymbol{n}]$, $-10^3 \leq \boldsymbol{x}[\boldsymbol{i}] \leq 10^3$, für $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$), und $\boldsymbol{k}$ eine natürliche Zahl ist ($1 \leq \boldsymbol{k} \leq 10^4$) :

```
Algorithm f(n, x, k):
    For i ← 1, k execute
        a ← x[1]
        j ← 1
        While j < n execute
            x[j] ← x[j + 1]
            j ← j + 1
        EndWhile
        x[n] ← a
    EndFor
    Return x[1]
EndAlgorithm
```

Welche der folgenden Aussagen sind wahr?

A. Wenn $\boldsymbol{n} = 10$, $\boldsymbol{x} = [1, 2, 3, ..., 10]$ und $\boldsymbol{k} = 3$, gibt der Algorithmus 6 zurück.

B. Wenn $\boldsymbol{n} = 10$, $\boldsymbol{x} = [1, 2, 3, ..., 10]$ und $\boldsymbol{k} = 117$, liefert der Algorithmus den Wert 8.

C. Wenn $\boldsymbol{n} = 5$, $\boldsymbol{x} = [4, 2, 5, 11, 13]$ und $\boldsymbol{k} = 117$, gibt der Algorithmus 5 zurück.

D. Wenn $\boldsymbol{n} = 5$, $\boldsymbol{x} = [1, 2, 3, 4, 5]$, gibt der Algorithmus denselben Wert zurück, falls $\boldsymbol{k} = 3$ oder $\boldsymbol{k} = 318$.

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

```
Algorithm ceFace(n):
    a ← 0; b ← 0
    For i ← 0, 9 execute
        x ← n
        c ← 0
        While x > 0 execute
            If x MOD 10 = i then
                c ← c + 1
            EndIf
            x ← x DIV 10
        EndWhile
        If c ≠ 0 then
            a ← a + 1
            If i MOD 2 = c MOD 2 then
                b ← b + 1
            EndIf
        EndIf
    EndFor
    Return a = b
EndAlgorithm
```

Welche der folgenden Aussagen sind wahr?

A. Der Algorithmus `ceFace(n)` liefert *True*, wenn bei der Zahl $\boldsymbol{n}$ die Anzahl der paarweise verschiedenen geraden Ziffern gleich der Anzahl der paarweise verschiedenen ungeraden Ziffern ist.

B. Wenn $\boldsymbol{n} = 12235$ ist, liefert der Algorithmus `ceFace(n)` den Wert *True*.

C. Wenn $\boldsymbol{n}$ nur aus ungeraden Ziffern besteht, gibt der Algorithmus `ceFace(n)` *True* zurück.

D. Wenn $\boldsymbol{n} = 10^{\boldsymbol{k}}$ ist, wobei $\boldsymbol{k}$ eine gerade Zahl ist, gibt der Algorithmus `ceFace(n)` *True* zurück.

6. Gegeben sei der Algorithmus `numar(n)`, wobei n eine natürliche Zahl ist ($1 \le n \le 10^6$).

```
Algorithm numar(n):
    While n > 9 execute
        n ← n DIV 10 + n MOD 10
    EndWhile
    Return n
EndAlgorithm
```

Welche der folgenden Aussagen sind wahr?

A. Der vom Algorithmus zurückgegebene Wert liegt für jeden Wert von n in der Menge {1, 2, ..., 9}.
B. Der Algorithmus liefert für keine Zahl n den Wert 0 zurück.
C. Es gibt 3 verschiedene Werte v, für die, wenn `numar(n) = v ist`, dann ist n durch v teilbar, unabhängig vom Wert von n.
D. Der Algorithmus liefert die Summe der Ziffern der Zahl n.

7. Gegeben sei die Binärzahl $b = 110101011_{(2)}$. Welche der folgenden Aussagen über die Zahl b sind wahr?

A. Zur Basis 4 hat sie den Wert $12223_{(4)}$.
B. Zur Basis 8 ist sie eine palindromische Zahl.
C. Sie ist eine ungerade Zahl.
D. Es handelt sich um eine durch 3 teilbare Zahl.

8. Gegeben sei der Algorithmus `f(a, b, n, d)`, wobei a und b natürliche Zahlen ungleich Null sind ($1 \le a, b \le 10^9$), n = 3, und d = [5, 7, 11]:

```
Algorithm f(a, b, n, d):
    c ← 0
    p ← 1
    g ← False
    While NOT g execute
        For i ← 1, n execute
            If (a MOD d[i] = 0) OR (b MOD d[i] = 0) then
                c ← c * 10 + 1
                p ← p * i
            EndIf
        EndFor
        g ← NOT g
    EndWhile
    Return c * 100 + p
EndAlgorithm
```

Welche der folgenden Aussagen sind wahr?

A. Wenn $a * b$ nicht durch 5 oder durch 7 oder durch 11 teilbar ist, liefert der Aufruf von `f(a, b, 3, [5, 7, 11])` den Wert 1 zurück.
B. Wenn $a * b$ weder durch 5, noch durch 7, noch durch 11 teilbar ist, liefert der Aufruf des Algorithmus `f(a, b, 3, [5, 7, 11])` den letzten Wert, der c zugewiesen wurde, nämlich 10.
C. Wenn $a * b$ durch 5 und durch 7 und durch 11 teilbar ist, liefert der Aufruf `von f(a, b, 3, [5, 7, 11])` den Wert 11106.
D. Der Aufruf `f(112233, 331122, 3, [5, 7, 11])` gibt 1003 zurück.

9. Gegeben sei der Algorithmus `f(x, n)`, wobei n eine natürliche Zahl ist ($3 \le n \le 10^4$) und x ein Vektor 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(x, n):
    For i ← 1, n - 1 execute
        If x[i + 1] < x[i] then
            tmp ← x[i + 1]
            x[i + 1] ← x[i]
            x[i] ← tmp
        EndIf
    EndFor
EndAlgorithm
```

Welche der folgenden Aussagen sind wahr?

A. Nach dem Aufruf `f(x, n)` befindet sich das größte Element des Vektors x an der Position n.
B. Nach dem Aufruf `f(x, n)` befindet sich das kleinste Element des Vektors x an der Position n.
C. Nach dem Aufruf `f(x, n)` wird der Vektor x aufsteigend sortiert.
D. Nach dem Aufruf `f(x, n)` wird der Vektor x absteigend sortiert.

10. Gegeben sei der Algorithmus `F(n)`, wobei n eine ganze Zahl ist ($0 < n < 10^5$).

```
Algorithm F(n):
    If n = 0 then
        Return 0
    Else
        If n MOD 2 = 0 then
            Return F(n DIV 10) + 1
        Else
            Return F(n DIV 10) - 1
        EndIf
    EndIf
EndAlgorithm
```

Welche der folgenden Aussagen sind richtig?

A. Wenn n = 543 ist, gibt der Algorithmus 2 zurück.
B. Wenn n = 18 ist, gibt der Algorithmus 0 zurück.
C. Wenn n = 41173 ist, gibt der Algorithmus 3 zurück.
D. Für keine Zahl n im Intervall [111, 999] gibt der Algorithmus `F(n)` den Wert 0 zurück.

11. Gegeben sei der Algorithmus `ceFace(x, n)`, wobei n eine natürliche Zahl ist ($1 \leq n \leq 10^4$) und x ein Vektor mit n ganzzahligen Elementen ist ($x[1], x[2], ..., x[n]$, $0 \leq x[i] \leq 10^4$, für $i = 1, 2, ..., n$):

```
Algorithm ceFace(x, n):
    i ← 1
    c ← 0
    d ← 0
    While i ≤ n execute
        If (x[i] MOD 10) MOD 2 = 0 then
            c ← c + 1
        Else
            If (x[i] MOD 10) MOD 3 = 0 then
                d ← d + 1
            EndIf
        EndIf
        i ← i + 1
    EndWhile
    Return c = d
EndAlgorithm
```

Unter welchen der folgenden Bedingungen gibt der Algorithmus `ceFace(x, n)` *True* zurück?

A. Wenn die Anzahl der geraden Zahlen im Vektor gleich der Anzahl der ungeraden Zahlen im Vektor ist.
B. Wenn die Anzahl der Zahlen im Vektor, die durch 2 teilbar sind, gleich der Anzahl der Zahlen im Vektor, die ein Vielfaches von 6 sind, ist.
C. Wenn die Anzahl der geraden Zahlen im Vektor gleich der Anzahl der ungeraden Zahlen im Vektor ist, bei denen die letzte Ziffer ein Vielfaches von 3 ist.
D. Wenn die Anzahl der Zahlen im Vektor, die durch 2 teilbar sind, gleich der Anzahl der Zahlen im Vektor, die ein Vielfaches von 3 sind, ist.

12. Gegeben sei der Algorithmus `f(n, x, c)`, wobei n eine natürliche Zahl ist ($1 \leq n \leq 10^4$) , x ein Vektor mit n ganzzahligen Elementen ($x[1], x[2], ..., x[n]$, $-10^3 \leq x[i] \leq 10^3$, für $i = 1, 2, ..., n$), und c eine natürliche Zahl ist ($1 \leq c \leq$ n):

```
Algorithm f(n, x, c):
    For i ← 1, c execute
        For j ← 1, n - 1 execute
            If x[j] > x[j + 1] then
                aux ← x[j]
                x[j] ← x[j + 1]
                x[j + 1] ← aux
            EndIf
        EndFor
    EndFor
    Return x[n + 1 - c]
EndAlgorithm
```

Welche der folgenden Aussagen sind richtig?

A. Nach dem Aufruf `f(5, [5, 4, 30, 5, 1], 3)` wird der Algorithmus den Wert 5 zurückgeben.
B. Wenn x die Werte 100, 99, 98, ..., 1 enthält, $n = 100$ und $c = 50$, gibt der Algorithmus `f(n, x, c)` den Wert 51 zurück.
C. Wenn $c = n$ ist, gibt der Algorithmus `f(n, x, c)` das kleinste Element des Vektors x zurück.
D. Ist $c = 1$, gibt der Algorithmus `f(n, x, c)` das minimale Element des Vektors x zurück.

13. Gegeben sei der Algorithmus `afla(n, x)`, wobei n eine natürliche Zahl ist ($3 \leq n \leq 10^4$) und x ein Vektor mit n ganzzahligen Elementen ist ($x[1], x[2], ..., x[n]$, $-100 \leq x[i] \leq 100$, für $i = 1, 2, ..., n$).

```
Algorithm afla(n, x):
    M ← x[1]
    For i ← 1, n - 2 execute
        For j ← i + 1, n - 1 execute
            For k ← j + 1, n execute
                If M < x[i] + x[j] + x[k] then
                    M ← x[i] + x[j] + x[k]
                EndIf
            EndFor
        EndFor
    EndFor
    Return M
EndAlgorithm
```

Welche der folgenden Aussagen ist nach dem Aufruf `afla(n, x)` wahr?

A. Wenn alle Elemente des Vektors x negativ sind, wird der Wert des Elements $x[1]$ zurückgegeben.
B. Wenn alle Elemente des Vektors x positiv sind, aber kleiner oder gleich $x[1]$ sind, wird der Wert des Elements $x[1]$ zurückgegeben.
C. Wenn $x[1]$, $x[2]$ und $x[3]$ positive Zahlen sind, gibt der Algorithmus ihre Summe zurück.
D. Für jeden Vektor x mit n Elementen, bei dem das Produkt aller Elemente eine ungerade Zahl ist, gibt der Algorithmus `afla(n, x)` eine ungerade Zahl zurück.

14. Geben Sie an, wie viele verschiedene nicht-orientierte Graphen mit 5 Knoten, die von 1 bis 5 nummeriert sind, so konstruiert werden können, dass Knoten 3 den Grad 1 hat. Zwei Graphen sind verschieden, wenn ihre Adjazenzmatrizen unterschiedlich sind.

A. 2^8 B. 5! C. $5 * C_5^4$ D. 2^5

15. Gegeben sei der Algorithmus `ceFace(A, n)`, wobei n eine natürliche Zahl ist ($1 < n \leq 10$), A ist eine $n \times n$ Matrix deren Elemente natürliche Zahlen sind ($A[1][1], A[1][2], ..., A[n][n]$, $1 \leq A[i][j] \leq 10$, für $i, j = 1, 2, ..., n$).

```
Algorithm ceFace(A, n):
    c ← 0
    For i ← 2, 3 execute
        first ← modifica(A, n, i - 1, i, 2, 4)
        second ← modifica(A, n, i, i - 1, 3, 2)
        If first AND second then
            c ← c + 1
        EndIf
    EndFor
    Return c
EndAlgorithm
```

```
Algorithm modifica(A, n, r, c, x, nr):
    If (r + x - 1 > n) OR (c + x - 1 > n) then
        Return False
    EndIf
    For i ← r, r + x - 1 execute
        For j ← c, c + x - 1 execute
            A[i][j] ← A[i][j] * nr
        EndFor
    EndFor
    Return True
EndAlgorithm
```

Wenn $n = 4$ und die Ausgangsmatrix $A=\begin{pmatrix} 4 & 8 & 3 & 4 \\ 2 & 5 & 6 & 8 \\ 4 & 3 & 2 & 9 \\ 5 & 3 & 1 & 7 \end{pmatrix}$ ist, welche der folgenden Aussagen sind nach dem Aufruf von `ceFace(A, n)` wahr?

A. Die Summe der Elemente auf der Hauptdiagonalen in der modifizierten Matrix ist 95.
B. In der modifizierten Matrix gibt es doppelt so viele durch 2 teilbare Elemente wie in der ursprünglichen Matrix.
C. Der zurückgegebene Wert ist 1.
D. Die Summe der Elemente auf der Hauptdiagonalen in der modifizierten Matrix ist 67.

16. Gegeben sei der Algorithmus `buildMatrix(n, x, m, y)`, wobei n und m natürliche Zahlen sind ($1 \leq n, m \leq 100$), x und y sind Vektoren mit n und m ganze Zahlen ($x[1], x[2], ..., x[n]$, wobei $-10^3 \leq x[i] \leq 10^3$, für $i = 1, 2, ..., n$ und $y[1], y[2], ..., y[m]$, mit $-10^3 \leq y[j] \leq 10^3$, für $j = 1, 2, ..., m$). Der Algorithmus `max(a, b)` gibt den Maximalwert zwischen den Zahlen a und b zurück. Algorithmus `zeros(n, m)` gibt eine Matrix mit n Zeilen und m Spalten zurück, deren Elemente gleich 0 sind.

```
Algorithm buildMatrix(n, x, m, y):
    A ← zeros(n + 1, m + 1)
    For i ← 1, n execute
        For j ← 1, m execute
            If x[i] = y[j] then
                A[i + 1][j + 1] ← A[i][j] + 1
            Else
                A[i + 1][j + 1] ← max(A[i][j + 1], A[i + 1][j])
            EndIf
        EndFor
    EndFor
    Return A[n + 1][m + 1]
EndAlgorithm
```

Welche der folgenden Aussagen sind richtig?

A. Nach dem Aufruf `buildMatrix(3, [3,2,3], 3, [2,3,3])` wird der Wert 2 zurückgegeben.
B. Der Algorithmus `buildMatrix(n, x, m, y)` liefert nur dann den Wert 0, wenn x und y die gleiche Länge haben und keine gemeinsamen Elemente besitzen.
C. Wenn $n = m$ ist und die Elemente des Vektors y eine Permutation der Elemente des Vektors x darstellen, dann gibt der Algorithmus `buildMatrix(n, x, m, y)` den Wert n zurück.
D. Es gibt Eingabedaten, bei denen der vom Algorithmus zurückgegebene Wert gleich $n + m$ ist.

17. Gegeben sei der Algorithmus `h(A, n, p)`, wobei n und p natürliche Zahlen sind ($1 \leq n, p \leq 10^3$) , und A ein Vektor mit n Elementen natürliche Zahlen ist ($A[1], A[2], ..., A[n]$, wobei $1 \leq A[i] \leq 100$, für $i = 1, 2, ..., n$):

```
Algorithm h(A, n, p):
    t ← n + 1; j ← 1; s ← 0
    For i ← 1, n execute
        s ← s + A[i]
        While s > p execute
            If t > i – j + 1 then
                t ← i – j + 1
            EndIf
            s ← s - A[j]
            j ← j + 1
        EndWhile
    EndFor
    Return t
EndAlgorithm
```

Welche Aufrufe geben den Wert 3 zurück?

A. `h([2, 1, 5, 6, 2, 5], 6, 11)`
B. `h([2, 1, 5, 6, 2, 5], 6, 10)`
C. `h([7, 8, 1, 2], 4, 15)`
D. `h([7, 8, 1, 2], 4, 16)`

18. Gegeben sei der Algorithmus `matrice(mat, n)`, wobei ***mat*** eine Matrix mit ***n*** Zeilen und ***n*** Spalten ($3 \leq n \leq 200$) deren Elemente natürlichen Zahlen sind (***mat***[1][1], ***mat***[1][2], ..., ***mat***[***n***][***n***], $1 \leq mat[i][j] \leq 10^4$, für ***i*** = 1, 2, ..., ***n***, ***j*** = 1, 2, ..., ***n***). Der Operator "/" steht für die Division von reellen Zahlen; z.B. 7 / 2 = 3.5.

```
Algorithm matrice(mat, n):
   c ← 0
   For i ← 1, n execute
      For j ← 1, n execute
         ... // (1)
            c ← c + mat[i][j]
         EndIf
      EndFor
   EndFor
   ... // (2)
EndAlgorithm
```

Durch welche Anweisungen sollten die mit `(1)` und `(2)` bezeichneten Zeilen in dem Algorithmus `matrice(mat, n)` ersetzt werden, damit er den Mittelwert der Elemente liefert, die nicht auf den Diagonalen der Matrix liegen?

A.
```
(1) If i ≠ j AND i ≠ n – j - 1 then
(2) Return c / (n * n - 2 * n + 1)
```

B.
```
(1) If i ≠ j OR i + j ≠ n + 1 then
(2) Return c / (n * n - 2 * n)
```

C.
```
(1) If i ≠ j AND i + j ≠ n + 1 then
(2) Return c / (n * n - 2 * n + n MOD 2)
```

D.
```
(1) If i ≠ j AND i ≠ n – j - 1 then
(2) Return c / (n * n - 2 * n + n MOD 2)
```

19. Gegeben sei der Algorithmus `f(x, n)`, wobei ***n*** eine natürliche Zahl ($2 \leq n \leq 10^3$) und ***x*** ein Vektor mit ***n*** natürliche Zahlen (***x***[1], ***x***[2], ..., ***x***[***n***], $0 \leq x[i] \leq 100$, für ***i*** = 1, 2, ..., ***n***) ist.

```
Algorithm f(x, n):
    left ← 1
    right ← n
    g ← True
    While left < right execute
        max ← -1
        For i ← left, right execute
            If x[i] > max then
               max ← x[i]
               poz1 ← i
            EndIf
        EndFor
        If g then
            poz2 ← left
            left ← left + 1
        Else
            poz2 ← right
            right ← right - 1
        EndIf
        If poz1 ≠ poz2 then
            x[poz1] ← x[poz2] // (*)
            x[poz2] ← max
        EndIf
        g ← NOT g
    EndWhile
    Return x
EndAlgorithm
```

Welche der folgenden Aussagen sind richtig?

A. Bei einem gegebenen Vektor ***x*** mit ***n*** Elementen, liefert der Aufruf `f(x, n)` für jede Permutation der Vektorelemente das gleiche Ergebnis.
B. Wenn ***n*** = 10 und ***x*** = [1, 2, 3, …, 10] wird nach dem Aufruf `f(x, n)` die mit (*) markierte Zeile 9 Mal ausgeführt.
C. Wenn der Vektor ***x*** aufsteigend sortiert ist, wird nach dem Aufruf von `f(x, n)` die mit (*) markierte Zeile genauso oft ausgeführt wie im Fall, dass der Vektor ***x*** absteigend sortiert ist.
D. Wenn ***n*** = 100 und ***x*** = [1, 2, 3, …, 100] ist, gibt es nur eine Permutation der Elemente des Vektors ***x***, für die nach dem Aufruf von `f(x, n)` die mit (*) markierte Zeile nicht ausgeführt wird.

20. Gegeben sei der Algorithmus `rezultat(azi, zile)`, wobei ***azi*** ($1 \leq azi \leq 5$), eine natürliche Zahl ist, die einen Arbeitstag unter der Woche angibt (der Wert 1 entspricht Montag, der Wert 2 entspricht Dienstag, ..., der Wert 5 entspricht Freitag) und ***zile*** ($1 \leq zile \leq 100$) eine natürliche Zahl ist.

Der Algorithmus `rezultat(azi, zile)` gibt den Arbeitstag zurück, an dem das Ergebnis einer Analyse veröffentlicht wird, unter Berücksichtigung der erforderlichen Verarbeitungszeit, ausgedrückt in Kalendertagen durch den Wert der Variablen ***zile***. Die Analysen werden an jedem Tag der Woche verarbeitet, und die Ergebnisse werden am ersten Arbeitstag, an dem sie verfügbar sind, veröffentlicht. Wenn die Verarbeitung an einem Wochenende (Samstag oder Sonntag) endet, wird das Ergebnis am folgenden Montag veröffentlicht.

Beispielsweise wird nach dem Aufruf `rezultat(2, 1)` der Wert 2 zurückgegeben (das Ergebnis wird am selben Tag veröffentlicht), und nach dem Aufruf `rezultat(3, 4)` wird der Wert 1 zurückgegeben (das Ergebnis wird am Montag veröffentlicht).

```
Algorithm rezultat(azi, zile):
    For i ← 2, zile execute
        azi ← azi + 1
        If azi > 7 then
            azi ← 1
        EndIf
    EndFor
    If azi ≥ 6 then
        azi ← 1
    EndIf
    Return azi
EndAlgorithm
```

Welche der folgenden Aussagen sind wahr?

A. Der Algorithmus `rezultatRec(azi, zile)` gibt für alle Eingabedaten denselben Wert zurück wie der Algorithmus `rezultat(azi, zile)` zurück.

```
Algorithm rezultatRec(azi, zile):
    If zile = 0 then
        If azi > 5 then
            Return 1
        EndIf
        Return azi
    EndIf
    Return rezultatRec(azi + 1, zile - 1)
EndAlgorithm
```

B. Nach dem Aufruf von `rezultat(4, 25)` wird der Wert 1 zurückgegeben.

C. Der Algorithmus `rezultat2(azi, zile)` gibt für alle Eingabedaten denselben Wert wie der Algorithmus `rezultat(azi, zile)` zurück und hat eine geringere Zeitkomplexität:

```
Algorithm rezultat2(azi, zile):
    v ← ((azi + zile - 2) MOD 7) + 1
    If v > 5 then
        v ← 1
    EndIf
    Return v
EndAlgorithm
```

D. Die Zeitkomplexität des Algorithmus `rezultat(azi, zile)` ist $O(1)$.

21. Gegeben sei der Algorithmus `suma(A, n, m, x1, y1, x2, y2)`, der die Summe aller Elemente eines Teils der Matrix $\boldsymbol{A}$ berechnet. Die Matrix $\boldsymbol{A}$ ist eine $\boldsymbol{n} \times \boldsymbol{m}$ Matrix ($\boldsymbol{A}[1][1], \boldsymbol{A}[1][2], ..., \boldsymbol{A}[1][\boldsymbol{m}], \boldsymbol{A}[2][1], ..., \boldsymbol{A}[\boldsymbol{n}][\boldsymbol{m}]$, wobei $2 < \boldsymbol{n}, \boldsymbol{m} < 100$, $0 \le \boldsymbol{A}[\boldsymbol{i}][\boldsymbol{j}] \le 10^3$ für $\boldsymbol{i} = 1, 2, ... \boldsymbol{n}, \boldsymbol{j} = 1, 2, ..., \boldsymbol{m}$). Der rechteckige Bereich wird durch die Koordinaten der linken oberen Ecke ($\boldsymbol{x1}, \boldsymbol{y1}$) und der rechten unteren Ecke ($\boldsymbol{x2}, \boldsymbol{y2}$) definiert, wobei $1 \le \boldsymbol{x1} < \boldsymbol{x2} \le \boldsymbol{n}$, $1 \le \boldsymbol{y1} < \boldsymbol{y2} \le \boldsymbol{m}$. Der Algorithmus `zeros(a, b)` gibt eine Matrix mit $\boldsymbol{a}$ Zeilen und $\boldsymbol{b}$ Spalten zurück, in der alle Elemente auf 0 initialisiert sind.

```
Algorithm compute(A, n, m):
    aux ← zeros(n, m)
    For i ← 1, n execute
        For j ← 1, m execute
            aux[i][j] ← A[i][j]
            If j > 1 then
                aux[i][j] ← aux[i][j] + aux[i][j - 1]
            EndIf
            If i > 1 then
                aux[i][j] ← aux[i][j] + aux[i - 1][j]
            EndIf
            If j > 1 AND i > 1 then
                aux[i][j] ← aux[i][j] - aux[i - 1][j - 1]
            EndIf
        EndFor
    EndFor
    Return aux
EndAlgorithm
```

```
1. Algorithm suma(A, n, m, x1, y1, x2, y2):
2.     aux ← compute(A, n, m)
3.     S1 ← 0; S2 ← 0; S3 ← 0; S4 ← 0
4.     If x1 > 1 AND y1 > 1 then
5.         S1 ← aux[x1 - 1][y1 - 1]
6.     EndIf
7.     If x1 > 1 then
8.         S2 ← aux[x1 - 1][y2]
9.     EndIf
10.    If y1 > 1 then
11.        S3 ← aux[x2][y1 - 1]
12.    EndIf
13.    S4 ← aux[x2][y2]
14.    ...............
15. EndAlgorithm
```

Welche Anweisung muss in Zeile 14 eingefügt werden, damit der Algorithmus `suma(A, n, m, x1, y1, x2, y2)` die gewünschte Summe liefert?

A. `Return S4 - S2 - S3 + S1`

B. `Return S4 + S2 + S3 + S1`

C. `Return S4 - S2 + S3 + S1`

D. `Return S4 - S2 - S3 - S1`

22. Gegeben sei der Algorithmus `f(n, x)`, mit $\boldsymbol{n}$ eine natürliche Zahl ($3 \leq \boldsymbol{n} \leq 10^4$) und $\boldsymbol{x}$ ein Vektor aus $\boldsymbol{n}$ natürlichen Zahlen ($\boldsymbol{x}[1], \boldsymbol{x}[2], ..., \boldsymbol{x}[\boldsymbol{n}]$, $1 \leq \boldsymbol{x}[\boldsymbol{i}] \leq 10^4$, für $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$).

```
Algorithm f(n, x):
    For i ← 1, n - 1 execute
        x[i + 1] ← (x[i + 1] * x[i]) DIV g(x[i], x[i + 1])
    EndFor
    Return x[n]
EndAlgorithm
```

```
Algorithm g(a, b):
    If a = b then
        Return a
    EndIf
    If a > b then
        Return g(a - b, b)
    EndIf
    Return g(a, b - a)
EndAlgorithm
```

Welche der folgenden Aussagen sind wahr?

A. Nach dem Aufruf `f(5, [12, 16, 8, 40, 24])` wird der Wert 240 zurückgegeben.
B. Die Gesamtzahl der Aufrufe des Algorithmus `g(a, b)` ist im Fall des Aufrufs `f(5, [8, 12, 16, 24, 40])` größer als im Fall des Aufrufs `f(5, [40, 24, 16, 12, 8])`.
C. Der Algorithmus berechnet den größten gemeinsamen Teiler aller Elemente des Vektors $\boldsymbol{x}$.
D. Für einen Vektor $\boldsymbol{x}$ mit $\boldsymbol{n} = 10$ Elementen, der einen einzigen Wert gleich 17 enthält, wobei alle anderen Werte gleich 1 sind, beträgt die maximale Anzahl der Aufrufe des Algorithmus `g(a, b)`, einschließlich des ersten Aufrufs, 153.

23. Bei einem Fußballspiel zwischen den Mannschaften **A** und **B**, bei dem 3 Tore erzielt werden und das 2 - 1 für die Mannschaft **A** endet, können die Tore in 3 Folgen erzielt werden:

1. 0 - 0, 0 - 1, 1 - 1, 2 - 1
2. 0 - 0, 1 - 0, 1 - 1, 2 - 1
3. 0 - 0, 1 - 0, 2 - 0, 2 - 1

Zum Beispiel, beginnt in Folge 1 das Spiel bei 0 - 0, Team **B** erzielt ein Tor (0 - 1), Team **A** gleicht aus (1 - 1) und Team **A** erzielt ein Tor (2 - 1).

Wie viele mögliche Reihenfolgen gibt es bei einem Spiel mit 4 Toren?

A. 4!
B. 16
C. 64
D. 1! + 2! + 3! + 2! + 1!

24. Gegeben sei der Algorithmus `litere(v, n)`, wobei $\boldsymbol{v}$ eine Zeichenkette mit $\boldsymbol{n}$ Elementen ($1 \leq \boldsymbol{n} \leq 100$) ist, die Großbuchstaben des englischen Alphabets sind. Für jedes Element an der Position $\boldsymbol{i}$ ($1 \leq \boldsymbol{i} \leq \boldsymbol{n}$) in der Zeichenkette wird ein Wert $\boldsymbol{S}[\boldsymbol{i}]$ berechnet, der die Anzahl der Positionen $\boldsymbol{j} \in \{1, 2, ..., \boldsymbol{i}\}$ darstellt, an denen es Buchstaben gibt, die im Alphabet vor $\boldsymbol{v}[\boldsymbol{i}]$ stehen. Wenn beispielsweise $\boldsymbol{v} =$ `['E', 'X', 'A', 'M', 'E', 'N']` ist der Wert $\boldsymbol{S}[6]$ gleich 4, da nur die Buchstaben `'E'`, `'A'` und `'M'` vor dem Buchstaben `'N'` im Alphabet stehen. Für den Zeichentyp werden die Vergleichs- und Subtraktionsoperationen mit dem ASCII-Code der Operanden durchgeführt.

Welcher der folgenden Algorithmen liefert die Summe aller Werte von $\boldsymbol{S}[\boldsymbol{i}]$ ($\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$)?

A.
```
Algorithm litere(v, n):
    sum ← 0
    For i ← 2, n execute
        For j ← 1, i execute
            If v[j] – 'A' > v[i] – 'A' then
                sum ← sum + 1
            EndIf
        EndFor
    EndFor
    Return n * (n - 1) DIV 2 – sum
EndAlgorithm
```

B.
```
Algorithm litere(v, n):
    sum ← 0
    For i ← 2, n execute
        For j ← 1, i execute
            If v[j] < v[i] then
                sum ← sum + 1
            EndIf
        EndFor
    EndFor
    Return sum
EndAlgorithm
```

C.
```
Algorithm litere(v, n):
    sum ← 0
    For i ← 2, n – 1 execute
        For j ← i + 1, n execute
            If v[j] – 'A' < v[i] – 'A' then
                sum ← sum + 1
            EndIf
        EndFor
    EndFor
    Return sum
EndAlgorithm
```

D.
```
Algorithm litere(v, n):
    sum ← 0
    For i ← 1, n execute
        For j ← 1, i execute
            If v[j] < v[i] – 'A' then
                sum ← sum + (v[i] – v[j])
            EndIf
        EndFor
    EndFor
    Return sum
EndAlgorithm
```

BABEŞ-BOLYAI UNIVERSITÄT

FAKULTÄT FÜR MATHEMATIK UND INFORMATIK

Zulassung - 9. September 2025

Schriftliche Prüfung in Informatik

PUNKTEANZAHL & LÖSUNGEN

ANFANGSPUNKTEANZAHL: 10 punkte

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