BABEŞ-BOLYAI UNIVERSITY
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Mate-Info Contest – March 26th, 2023

Written Test in Computer Science

IMPORTANT NOTE:
In the absence of other specifications:
- Assume that all arithmetic operations are performed on unlimited data types (there is no *overflow*/*underflow*).
- Arrays are indexed starting from 1.
- All constraints refer to the values of the actual parameters at the time of the initial call.

1. Let us consider the algorithm `f(a, b)`, where $\boldsymbol{a}$ and $\boldsymbol{b}$ are non-zero natural numbers ($1 \leq \boldsymbol{a}, \boldsymbol{b} \leq 10^9$).

```
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
```

Which of the following statements are true?

A. For the call `f(2000, 21)`, the algorithm returns 1.
B. For the call `f(2000, 21)` the algorithm does not finish its execution due to the condition on line 2.
C. In order for the algorithm to return the greatest common divisor of $\boldsymbol{a}$ and $\boldsymbol{b}$, line 8 should be changed to: **Return** `f(b - a, b)`.
D. In order for the call `f(2000, 21)` to return the value 1, line 8 should be changed to: **Return** `f(b - a, b - a)`.

2. Let us consider the following algorithm sequence, where $\boldsymbol{a}$ is an array of $\boldsymbol{n}$ natural numbers ($\boldsymbol{a}[1], \boldsymbol{a}[2], ..., \boldsymbol{a}[\boldsymbol{n}]$, $1 \leq \boldsymbol{a}[\boldsymbol{i}] \leq 10^4$, for $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$), and $\boldsymbol{n}$ is a non-zero natural number ($1 \leq \boldsymbol{n} \leq 10^4$):

```
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
```

Which of the following statements are true in the moment when $\boldsymbol{i}$ becomes 2?

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

3. Let us consider the algorithm `alg(n)`, where $\boldsymbol{n}$ is a natural number ($0 \leq \boldsymbol{n} \leq 10^9$).

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

Which of the following statements are true?

A. If $\boldsymbol{n} = 4$, the value returned by the algorithm is 7.
B. The algorithm returns the sum of all natural numbers that are smaller than $\boldsymbol{n}$.
C. The algorithm returns the sum of all natural numbers that are smaller or equal to $\boldsymbol{n}$.
D. If $\boldsymbol{n} = 7$, the algorithm returns the value 7.

4. Let us consider the algorithm `f(nr)`, where ***nr*** is an integer ($-10^4 \le nr \le 10^4$).

```
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
```

For what values of ***nr*** will the algorithm return the value 1?

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

5. Let us consider the algorithm `afis(n)`, where ***n*** is a natural number ($1 \le n \le 10^4$):

```
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
```

For which of the following calls will the values **`2 4`** be printed, in this exact order?

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

6. Let us consider the algorithm `Afișare(a)`, where ***a*** is a natural number ($1 \le a \le 10^4$).

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

What will be displayed for the call `Afișare(1000)`?

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

7. Let us consider the algorithm `f(n, x)`, where ***n*** is a natural number ($3 \le n \le 10^4$), and ***x*** is an array of ***n*** natural numbers ($x[1], x[2], ..., x[n]$, $1 \le x[i] \le 10^4$, for $i = 1, 2, ..., n$):

```
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
```

For which of the following calls will the algorithm return *True*?

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. What is the result of converting the base 10 number $2^{10} - 2^5 - 1$ to base 2?

A. `1111011111`
B. `1010011001`
C. `1000011001`
D. None of the answers A, B, or C

9. Let us consider the algorithms `one(a, b)` and `two(n, m)` where the input parameters $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{n}$ and $\boldsymbol{m}$ are natural numbers ($2 \leq \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{n}, \boldsymbol{m} \leq 10^6$, $\boldsymbol{n} < \boldsymbol{m}$).

```
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
```

Which of the following statements are true?

A. The algorithm `two(n, m)` will not print anything, regardless of the values of its input parameters.
B. The algorithm `two(n, m)` prints the prime numbers from the interval [$\boldsymbol{n}$, $\boldsymbol{m}$].
C. The algorithm `two(n, m)` prints the numbers that are divisible by 2 from the interval [$\boldsymbol{n}$, $\boldsymbol{m}$].
D. None of the other variants is correct.

10. Let us consider the algorithm `decide(n, x)`, where $\boldsymbol{n}$ is a non-zero natural number ($1 \leq \boldsymbol{n} \leq 10^4$), and $\boldsymbol{x}$ is an array of $\boldsymbol{n}$ natural numbers ($\boldsymbol{x}[1], \boldsymbol{x}[2], ..., \boldsymbol{x}[\boldsymbol{n}]$, $0 \leq \boldsymbol{x}[\boldsymbol{i}] \leq 100$, for $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$).

```
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
```

When will the algorithm `decide(n, x)` return *True* ?

A. Always
B. If the elements of array $\boldsymbol{x}$ are [1, 2, 3]
C. If the elements of array $\boldsymbol{x}$ are [1, 1, 1]
D. If the elements of array $\boldsymbol{x}$ form a palindrome, meaning that $\boldsymbol{x}[\boldsymbol{i}] = \boldsymbol{x}[\boldsymbol{n} - \boldsymbol{i} + 1]$ for all $\boldsymbol{i} = 1, 2, ..., \boldsymbol{n}$

11. Let us consider the algorithm `alg(a, b)`, where $\boldsymbol{a}$ and $\boldsymbol{b}$ are natural numbers ($1 \leq \boldsymbol{a}, \boldsymbol{b} \leq 10^3$).

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

Which of the following statements are true?

A. For the call `alg(2, 3)` the algorithm returns 7.
B. For the call `alg(2, 3)` the algorithm is called 4 times, taking into account the initial call.
C. The algorithm calculates and returns the value of $\boldsymbol{a}^{\boldsymbol{b}-1}$.
D. The algorithm calculates and returns the value of $\boldsymbol{a}^{\boldsymbol{b}}$.

12. Let us consider the algorithm `ceFace(a, b)`, where ***a*** and ***b*** are natural numbers ($1 < \boldsymbol{a}, \boldsymbol{b} \leq 10^5$). The algorithm `prim(n)` returns *True* if the number ***n*** > 1 is prime, otherwise it returns *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
```

What will be printed after the call `ceFace(100, 2)`?

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

13. Let us consider the algorithm `f(n, p)` where ***n*** is a non-zero natural number ($1 \leq \boldsymbol{n} \leq 10^9$), and ***p*** is a natural number ($0 \leq \boldsymbol{p} \leq 10^9$):

```
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
```

Which of the following calls will return the value 22?

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

14. Let us consider the algorithm `cifre(n)`, where ***n*** is a natural number ($0 \leq \boldsymbol{n} \leq 10^3$).

```
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
```

Which of the following statements are true?

A. The algorithm always returns a number smaller than 10.
B. The algorithm returns -1 if and only if the initial value of ***n*** is 0.
C. For ***n*** ≥ 1, the algorithm returns the least significant odd digit of ***n***, or -1, if this does not exist.
D. For ***n*** ≥ 1 the algorithm returns the most significant odd digit of ***n***, or -1, if this does not exist.

15. Let us consider the algorithm `ceFace(a, b)`, where ***a*** and ***b*** are natural numbers ($0 \le \boldsymbol{a}, \boldsymbol{b} \le 10^6$).

```
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
```

Which of the following statements are true?

A. If ***a*** = 0 and ***b*** = 0, the algorithm returns 1.
B. If ***a*** = 11 and ***b*** = 111, the algorithm returns 11.
C. If ***a*** = 5678 and ***b*** = 5162738, the algorithm returns 1024.
D. If ***a*** = 112233 and ***b*** = 331122, the algorithm returns 110000.

16. Let us consider the algorithms `ceva(n, m)` and `altceva(n, m)`, where ***n*** and ***m*** are non-zero natural numbers ($1 \le \boldsymbol{n}, \boldsymbol{m} \le 10^{12}$ and $\boldsymbol{m} \le \boldsymbol{n}$).

```
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
```

Which of the following statements are true?

A. The time complexity of algorithm `ceva(n, m)` is $O(\log \boldsymbol{m})$.
B. The algorithm `altceva(n, m)` returns 0 if and only if ***n*** = ***m***.
C. The precondition $\boldsymbol{m} \le \boldsymbol{n}$ is required, since if $\boldsymbol{m} > \boldsymbol{n}$ the algorithm `altceva(n, m)` will always enter an infinite loop.
D. There exist ***n*** and ***m*** ($\boldsymbol{m} \le \boldsymbol{n}$) for which `altceva(n, m)` displays two values in ascending order.

17. Let us consider the algorithm `h(s, d, A)`, where ***s*** and ***d*** are non-zero natural numbers ($1 \leq s, d \leq 10^3$) and ***A*** is an array of ***n*** non-zero natural numbers (***A***[1], ***A***[2], ..., ***A***[***n***], $1 \leq A[i] \leq 10^3$, for ***i*** = 1, 2, ..., ***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
```

For which values of the number ***n*** and array ***A*** will the call `h(1, n, A)` return the value 5?

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

18. Let us consider the algorithm `f(a, x)`, where ***x*** is a non-zero natural number ($1 \leq x \leq 10^4$) and ***a*** is an array of 10 non-zero natural numbers (***a***[1], ***a***[2], ..., ***a***[10]) .

```
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
```

For which of the following input values will the algorithm enter an infinite loop?

A. ***a*** = [3, 3, 3, 3, 3, 3, 3, 3, 3, 3] and ***x*** > 3
B. ***a*** = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] and ***x*** < 10
C. ***a*** = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20] and $1 < x < 20$, ***x*** – odd number
D. ***a*** = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20] and $1 < x < 20$, ***x*** – even number

19. Let us consider the algorithm `f(a)`, where ***a*** is a natural number ($1 \leq a \leq 10^9$).

```
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
```

Which of the following statements are true?

A. For ***a*** = 253401976 the algorithm `f(a)` is called 8 times. The initial call is also counted.
B. For ***a*** = 253401976 the algorithm `f(a)` is called 9 times. The initial call is also counted.
C. For ***a*** = 253401976 the result returned by the algorithm is 2406.
D. The result returned by the algorithm `f(a)` for the number ***a*** formed using only even digits equals ***a***.

20. Let us consider the algorithm `A(k)`, where parameter ***k*** is a non-zero natural number ($1 \leq k \leq 10^9$).

```
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
```

- Where [***gr***] is the integer part of ***gr***.
- The algorithm `radical(x)` returns the square root of ***x***.
- The / operator represents real number division, for example: 7 / 2 = 3.5

Which of the following statements are correct?

A. The algorithm `A1(k)` defined below is equivalent with algorithm `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. The algorithm `A2(k)` defined below is equivalent with algorithm `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. The algorithm `A(k)` returns the ***k***-th element of the sequence formed from concatenating the arrays in the form of [1, 2, ..., ***i***], for each ***i*** = 1, 2, ..., ***k***, in this order (that is [1, 1, 2, 1, 2, 3, 1, 2, 3, 4, ...]).

D. The algorithm `A(k)` returns the ***k***-th element of the sequence formed from concatenating the arrays in the form of [***i***, ..., 2, 1], for each ***i*** = 1, 2, ..., ***k***, in this order (that is [1, 2, 1, 3, 2, 1, 4, 3, 2, 1, ...])

21. Let us consider the algorithm `ceFace(a, lung)`, where ***lung*** is a natural number ($1 \leq lung \leq 10^5$), and ***a*** is an array of ***lung*** integers (***a***[1], ***a***[2], ..., ***a***[***lung***]). The array ***a*** contains at least one positive number.

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

Knowing that a subarray of array ***x*** = [***x***[1], ***x***[2], ..., ***x***[***n***]] is formed by elements of the array ***x*** that occupy consecutive positions (for example ***y*** = [***x***[3], ***x***[4], ***x***[5], ***x***[6]]) is a length 4 subarray of array ***x***), specify which of the following statements are true:

A. If there is only one positive number in array ***a***, the algorithm returns its value.
B. The algorithm returns the length of one of the subarrays that have the maximum sum in array ***a***.
C. The algorithm returns the sum of one of the subarrays that have the maximum sum in array ***a***.
D. The algorithm returns the sum of the positive numbers that are on consecutive positions at the end of array ***a***.

22. Let us consider the algorithm `ceFace(sir, a, b)`, where ***sir*** is an array of ***n*** ($1 \le n \le 100$) non-zero distinct natural numbers in ascending order (***sir***[1], ***sir***[2], ..., ***sir***[***n***]), ***a*** and ***b*** are natural numbers ($1 \le a, b \le n$) .

```
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
```

Which of the following statements are true, considering the initial call `ceFace(sir, 1, n)`?

A. If array ***sir*** is comprised of the first ***n*** distinct natural numbers, then the algorithm returns ***n*** + 1.

B. The algorithm returns the greatest position ***p*** that is less than or equal to ***n*** **DIV** 2 for which ***sir***[***p***] = ***p*** or 1, if such a position does not exist ($1 \le p \le n$).

C. The algorithm returns the greatest position ***p*** that is less than or equal to ***n*** **DIV** 2 for which ***sir***[***p***] ≠ ***p*** or ***n*** + 1, if such a position does not exist ($1 \le p \le n$).

D. The algorithm returns the smallest non-zero natural number that does not appear in the array ***sir***.

23. Let us consider the algorithm `ceFace(s, x, c, y, n, m, k)`, where ***s*** is an array of characters (***s***[1], ***s***[2], ..., ***s***[***x***]) of length ***x***, and ***c*** is an array of characters (***c***[1], ***c***[2], ..., ***c***[***y***]) of length ***y***. The identifiers ***x***, ***y***, ***n***, ***m*** and ***k*** memorize non-zero natural numbers ($1 \le x, y, n, m, k \le 100$).

```
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
```

By calling `ceFace("+-", 2, "123", 3, 2, 2, 4)`, we aim to obtain a valid arithmetic expression (that is an arithmetic expression created by alternating one operator with one operand; it can start with one of the operators '+' or '-' and must end with an operand). Which of the following statements are **NOT** true?

A. Lines 5 and 10 can be filled in with the instruction `k ← k + 7`.

B. Line 5 can be filled in with the instruction `k ← k + 2`, and line 10 with the instruction `k ← k + 5`.

C. Lines 5 and 10 can be filled in with the instruction `k ← k + 2`.

D. Line 5 can be filled in with the instruction `k ← k + 7`, and line 10 with the instruction `k ← k - 1`.

24. Let us consider the natural number ***n*** ($1 \le n \le 50$) and the array ***x*** having ***n*** integer elements (***x***[1], ***x***[2], ..., ***x***[***n***]). Which of the following statements are true, regardless of the value of ***n*** and the values of the array's elements?

A. There exists a natural number ***k*** ($1 \le k \le n$), so that ***x***[1] + ***x***[2] + ... + ***x***[***k***] is divisible by ***n***.

B. There exist (***i***, ***j***), $0 \le i < j \le n$, so that the sum ***x***[***i*** + 1] + ***x***[***i*** + 2] + ... + ***x***[***j***] is divisible by ***n***.

C. Neither of statements A and B is true.

D. Knowing that a subarray of array ***x*** = [***x***[1], ***x***[2], ..., ***x***[***n***]] is comprised of elements of array ***x*** that occupy consecutive positions (for example, ***y*** = [***x***[3], ***x***[4], ***x***[5], ***x***[6]] is a length 4 subarray of ***x***), there exists a natural number ***k***, ($1 \le k \le n$), so that in the array ***x*** there exists a subarray of ***k*** elements ($1 \le k \le n$) whose sum is divisible by ***n***.

BABEŞ-BOLYAI UNIVERSITY

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Mate-Info Contest – March 26th, 2023

Written Exam for Computer Science

GRADING AND SOLUTIONS

DEFAULT: 10 points

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