BABEŞ-BOLYAI UNIVERSITY
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Admission exam – July 19th 2024
Written Exam for Computer Science

IMPORTANT NOTE:
Unless otherwise specified:

- All arithmetic operations are performed on unlimited data types (there is no *overflow* / *underflow*).
- Arrays, matrices and strings are indexed starting from 1.
- All restrictions apply to the values of the actual parameters at the time of the initial call.
- A subarray consists of elements occupying consecutive positions in the array.
- A subsequence of an array consists of elements not necessarily occupying consecutive positions in the array, but in the order in which they appear in the given array.
- If on the same row there are several consecutive assignment statements, they are separated by "; ".

1. Consider the algorithm `ceFace(A, m, n)`, where m is a natural number ($1 \leq m \leq 100$), and A is an array of m integer elements ($A[1], A[2], ..., A[m]$, $-10^5 \leq A[i] \leq 10^5$, for $i = 1, 2, ..., m$), and n is a natural number ($n \leq m$):

```
Algorithm ceFace(A, m, n):
    For i ← 1, n execute
        min_idx ← i
        For j ← i + 1, m execute
            If A[min_idx] > A[j] then
                min_idx ← j
            EndIf
        EndFor
        aux ← A[i]
        A[i] ← A[min_idx]
        A[min_idx] ← aux
    EndFor
EndAlgorithm
```

State which of the following statements are true:

A. If $n = m$, then after executing the algorithm `ceFace(A, m, n)` the elements of the array will be ordered ascendingly.
B. If $n = m$, then after executing the algorithm `ceFace(A, m, n)` the elements of the array will be ordered descendingly.
C. If $A = [4, 64, 1, 25, 12, 22, 2, 11]$, $n = 2$ and $m = 8$, after executing the `ceFace(A, m, n)` algorithm at least the first 3 elements of array A will be ordered ascendingly.
D. If $n < m$, after executing the `ceFace(A, m, n)` algorithm at least the first $n + 1$ elements of array A will be ordered ascendingly.

2. Consider the algorithm `h(n, a)`, where n is a natural number ($1 \leq n \leq 10^3$) and a is an array of n integer elements ($a[1], a[2], ..., a[n]$), where $-100 \leq a[i] \leq 100$, for $i = 1, 2, ..., n$):

```
Algorithm h(n, a):
    If n = 1 then
        Return a[n]
    Else
        If a[n] > a[n - 1] then
            a[n - 1] ← a[n] - a[n - 1]
        Else
            a[n - 1] ← a[n] + a[n - 1]
        EndIf
        Return h(n - 1, a)
    EndIf
EndAlgorithm
```

For what values of the number n and array a, will the call `h(n, a)` return the value 1?

A. $n = 6$, $a = [1, 2, 3, 4, 5, 6]$
B. $n = 6$, $a = [6, 5, 4, 3, 2, 1]$
C. $n = 5$, $a = [1, 5, 4, 2, 3]$
D. $n = 2$, $a = [1, 2]$

3. Consider the expression `E = (x MOD 3 = 0) OR ((y < x) OR NOT ((y * 3) MOD 7 ≤ 3))`.

What is the value of the expression, if $x = 10$ and $y = 41$?

A. *True*
B. *False*
C. Same value as expression `E1`, where `E1 = NOT ((y MOD 3 = 0) OR ((x < y) OR NOT ((x * 3) MOD 7 ≤ 3)))`
D. Same value as expression `E2`, where `E2 = (x MOD 3 = 0) OR ((x < y) AND ((y * x) MOD 3 ≤ 7))`

4. Ion implements the following algorithm to check if the natural number ***nr*** ($0 < nr < 10^6$) is prime.

```
Algorithm prim(nr):
    If nr < 2 then
        Return False
    EndIf
    If (nr > 2) AND (nr MOD 2 = 0) then
        Return False
    EndIf
    d ← 3
    While d * d < nr execute
        If nr MOD d = 0 then
            Return False
        EndIf
        d ← d + 2
    EndWhile
    Return True
EndAlgorithm
```

Ion tests the correctness of the algorithm on the numbers in the set ***M*** = {2, 3, 4, 5, 10, 11, 13}. Which of the following statements are true?

A. The algorithm is correct and returns the correct result for both the numbers in ***M*** and any other number within the specifications.
B. The algorithm is incorrect, but returns the correct result for the numbers in ***M***.
C. The algorithm is incorrect, and returns incorrect results for all numbers in ***M***.
D. The algorithm is incorrect, but returns the correct result for at least one number in ***M*** and an incorrect result for at least one other number in ***M***.

5. Consider the algorithm `f(n, x)`, where ***n*** is a natural number ($1 \leq n \leq 10^4$), and ***x*** is an array of ***n*** integer elements (***x***[1], ***x***[2], ..., ***x***[***n***], $-200 \leq x[i] \leq 200$, for ***i*** = 1, 2, ..., ***n***):

```
Algorithm f(n, x):
    a ← True
    i ← 1
    While a AND (i < n) execute
        a ← (x[i] > x[i + 1])
        i ← i + 1
    EndWhile
    Return a
EndAlgorithm
```

For which of the following input data does the `f(n, x)` algorithm return *True*?

A. For any array containing its positive elements followed by its negative elements
B. For any strictly descending array
C. For any array that does not contain positive elements
D. For array ***x*** = [5, 4, 3, 2, 1, 0, -1, -2, -3, -4, -5] and ***n*** = 11

6. Consider expression ***E*** = $AB_{(16)} + 120_{(3)} - 120_{(4)}$, where the notation $x_{(b)}$ signifies the number ***x*** written in base ***b***.

Which value corresponds to the expression ***E***?

A. $162_{(10)}$ B. $278_{(8)}$ C. $1000101_{(2)}$ D. $242_{(8)}$

7. Consider the algorithm `f(a, b)`, where ***a*** and ***b*** are non-zero natural numbers ($0 < a, b < 10^4$).

```
Algorithm f(a, b):
    If a = 0 then
        Return b
    EndIf
    x ← f(a - 1, b + 1)
    Return f(a - 1, x - 2)
EndAlgorithm
```

What is the smallest natural number ***a*** for which the call `f(a, 15)` returns a strictly negative number?

A. 3 B. 4 C. 5 D. 6

8. Consider the `compute(n)` algorithm, where ***n*** is a natural number ($1 < n \leq 10^4$).

```
Algorithm compute(n):
    x ← 0
    While n > 0 execute
        If n MOD 2 = 1 then
            x ← x + 1
        EndIf
        n ← n DIV 2
    EndWhile
    Return x
EndAlgorithm
```

Which of the following statements are true?

A. If ***n*** is odd, the `compute(n)` algorithm returns a value greater than 1.
B. The `compute(n)` algorithm returns the sum of the digits in the representation of ***n*** in base 2.
C. The `compute(n)` algorithm returns the number of odd divisors (proper and improper) of the natural number ***n***.
D. The `compute(n)` algorithm returns the number of bits equal to 1 in the representation of ***n*** in base 2.

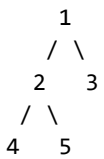
9. Consider the algorithm `f(p, q, r)`, where ***p***, ***q*** and ***r*** are Boolean values:

```
Algorithm f(p, q, r):
    While (p AND (NOT r)) OR (NOT q) execute
        Write (q AND (p OR r))
        p ← NOT p
        r ← q OR p
    EndWhile
EndAlgorithm
```

Which of the following statements are true for the call `f(True, False, True)`?

A. The algorithm enters an infinite loop, displaying *False* repeatedly.
B. The algorithm does not display anything.
C. The algorithm displays the value *False* only once.
D. The algorithm displays the values *False True False.*

10. Consider the following binary tree:



Which of the following sequences of nodes correspond to the tree traversal in preorder?

A. 1, 2, 4, 5, 3
B. 4, 2, 5, 1, 3
C. 1, 2, 3, 4, 5
D. 4, 5, 2, 3, 1

11. Consider the algorithm `mark(n, m, a)`, where ***n*** and ***m*** are non-zero natural numbers ($1 \leq n, m \leq 10$), and ***a*** is an array of ***n*** natural numbers (***a***[1], ***a***[2], ..., ***a***[***n***]). The algorithm `tuple(i, j, k)`, where ***i***, ***j,*** and ***k*** are non-zero natural numbers ($1 \leq i, j, k, \leq 10$) returns *True* or *False*.

```
Algorithm mark(n, m, a):
   a[1] ← 1
   For i ← 2, n execute
      a[i] ← 0
   EndFor
   ready ← False
   While NOT ready execute
      ready ← True
      For i ← 1, n execute
         For j ← 1, n execute
            For s ← 1, m execute
               If a[i] = 1 AND tuple(i, s, j) AND a[j] = 0 then
                  a[j] ← 1
                  ready ← False
               EndIf
            EndFor
         EndFor
      EndFor
   EndWhile
EndAlgorithm
```

Assume that for all triplets below, the algorithm `tuple(i, j, k)` returns *True*. For which pairs of triplets will the effect of the call `mark(3, 3, a)` be that of setting all the elements of array ***a*** to the value 1?

A. (1, 1, 2) and (2, 2, 3)
B. (1, 1, 2) and (3, 2, 2)
C. (1, 2, 2) and (1, 3, 3)
D. (1, 2, 2) and (3, 3, 1)

12. Consider a matrix ***mat*** with ***n*** rows and ***n*** columns ($1 \leq n \leq 200$, ***mat***[1][1], ..., ***mat***[1][**n**], ***mat***[2][1], ..., ***mat***[2][***n***], ..., ***mat***[***n***][1], ..., ***mat***[***n***][***n***]) and the `matrice(mat, n)` algorithm.

```
Algorithm matrice(mat, n):
    k ← 1
    For i ← 1, n execute
        For j ← 1, n execute
            mat[i][j] ← k
            k ← k * (-1)
        EndFor
    EndFor
    Return mat
EndAlgorithm
```

Which of the following statements are true for the matrix returned by the `matrice(mat, n)` call?

A. If ***n*** = 31, the product of the elements on the main diagonal is 1.
B. If ***n*** = 32, the product of the elements on the first row is 1.
C. If ***n*** = 127, the element on the last row and the last column is -1.
D. If ***n*** = 128, the sum of the elements on the first column is 1.

13. Consider the algorithm `modifica(n, a)`, where n is a natural number ($1 \leq n \leq 10^3$), and a is an array of n integer elements ($a[1], a[2], ..., a[n], -100 \leq a[i] \leq 100, i = 1, ..., n$):

```
Algorithm modifica(n, a):
    x ← a[n]
    i ← 0
    For j ← 1, n - 1 execute
        If a[j] ≤ x then
            i ← i + 1
            t ← a[i]
            a[i] ← a[j]
            a[j] ← t
        EndIf
    EndFor
    t ← a[i + 1]
    a[i + 1] ← a[n]
    a[n] ← t
    Return a
EndAlgorithm
```

Which of the following statements are true?

A. If array a is sorted ascendingly, it will remain sorted ascendingly when the algorithm finishes.
B. If array a is sorted strictly descending, then in the array returned by the algorithm the maximum element will be on the last position.
C. The array returned by the algorithm will always have the maximum element in the last position.
D. If $n = 100$, and the elements of array a have the property that $a[i] = i$ **MOD** 2, for $i = 1, 2, ..., n$, then at the end of the algorithm's execution the array will be sorted ascendingly.

14. Consider the algorithm `f(v, n)`, where n is a natural number ($2 \leq n \leq 10^4$) and v is an array of n natural numbers ($v[1], v[2], ..., v[n], 1 \leq v[i] \leq 10^3$, for $i = 1, 2, ..., n$).

```
Algorithm f(v, n):
    a ← 0; b ← n; i ← 1
    While i < n execute
        If v[i] MOD 3 = 0 then
            a ← a + v[i]
            b ← b + 1
        EndIf
        i ← i + 1
        b ← b - 1
    EndWhile
    If b = 0 then
        Return 0
    EndIf
    i ← 0
    While a ≥ b execute
        a ← a – b
        i ← i + 1
    EndWhile
    Return i
EndAlgorithm
```

Which of the following statements are true?

A. The algorithm returns the arithmetic mean of the elements that are multiples of 3 in array v, or 0 if the array contains no multiples of 3.
B. The algorithm returns the greatest common divisor of the elements that are multiples of 3 in array v, or 0 if the array contains no multiples of 3.
C. The algorithm returns the number of elements that are multiples of 3 in array v, or 0 if the array contains no multiples of 3.
D. None of the answers A., B., C is true.

15. To determine all the subsets of the set $A = \{4, 8, 9, 12, 15\}$ with 5 elements, a student wrote the algorithm `generare(i, n, x, A)`. The set is represented using array A of n natural number elements. The generated subsets are displayed using the algorithm `afis(m, x, A)`, x being an auxiliary array indexed from 0 and m a natural number representing the length of the current array x. Before the `generare(1, 5, x, A)` call, the element $x[0]$ was initialized with 0.

```
Algorithm generare(i, n, x, A):
    For j ← n, x[i - 1] + 1, -1 execute
        x[i] ← j
        afis(i, x, A)
        generare(i + 1, n, x, A)
    EndFor
EndAlgorithm
```

```
Algorithm afis(m, x, A):
    Write "{", a[x[1]]
    For i ← 2, m execute
        Write ", ", a[x[i]]
    EndFor
    Write "}", newline
EndAlgorithm
```

Knowing that the first 4 subsets displayed are, in this order: {15}, {12}, {12, 15}, {9} which will be the 8th generated subset (the empty subset is not considered)?

A. {9, 12} B. {8} C. {9, 12, 15} D. {8, 15}

16. Consider the algorithm `f(x, n, k)` where ***n*** and ***k*** are natural numbers ($3 \leq n \leq 10^4$, $1 \leq k \leq 10^4$), and ***x*** is an array of ***n*** natural numbers (***x***[1], ***x***[2], ..., ***x***[***n***], $1 \leq x[i] \leq 10^4$, for $i = 1, 2, ..., n$):

```
Algorithm f(x, n, k):
    If k > n then
        Return 0
    EndIf
    For i ← 1, n - 1 execute
        x[i + 1] ← x[i + 1] + x[i]
    EndFor
    Return x[k]
EndAlgorithm
```

For which of the following calls will the algorithm return the value 10?

A. `f([1, 4, 6], 3, 3)`
B. `f([1, 2, 3, 4, 5], 5, 3)`
C. `f([1, 2, 3, 4], 4, 4)`
D. `f([10, 15, 25], 3, 1)`

17. Consider the algorithm `decide(n)`, where ***n*** is a natural number ($10^4 \leq n \leq 10^7$):

```
Algorithm decide(n):
    m ← 10
    abc ← n DIV m
    While abc ≥ 1000 execute
        m ← m * 10
        abc ← n DIV m
    EndWhile
    bc ← abc MOD 100
    f ← (bc < 2)
    i ← 2
    While i ≤ bc DIV 2 execute
        If bc MOD i = 0 then
            f ← True
            i ← bc
        EndIf
        i ← i + 1
    EndWhile
    Return f
EndAlgorithm
```

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

A. `decide(865756)`
B. `decide(72387)`
C. `decide(103983)`
D. `decide(10405)`

18. Consider the algorithm `ceFace(n)`, where ***n*** is a non-zero natural number ($1 \leq n < 10^3$).

```
Algorithm ceFace(n):
    Return ceFaceRecursiv(n, 1, 1)
EndAlgorithm

Algorithm ceFaceRecursiv(n, a, b):
    If n = 0 then
        Return 1
    Else
        If n < 0 OR b > n then
            Return 0
        Else
            Return ceFaceRecursiv(n, a + b, a) + ceFaceRecursiv(n - a, a + b, a)
        EndIf
    EndIf
EndAlgorithm
```

Which of the following statements are true?

A. In the range [11, 16] there is only one ***x*** value for which the `ceFace(x)` algorithm returns 1.
B. For any number ***n***, the `ceFace(n)` algorithm will return the value 0 or 1.
C. The `ceFace(n)` algorithm returns the number of ways to write the number ***n*** as a sum of consecutive numbers.
D. The `ceFace(n)` algorithm returns the number of different sets whose elements are Fibonacci numbers other than 0 and which have the sum equal to ***n***.

19. Consider the algorithm `ceFace(x, n)`, where ***n*** is a natural number ($1 \leq n \leq 10^4$), ***x*** is an array of ***n*** elements that are digits (***x***[1], ***x***[2], ..., ***x***[***n***], $1 \leq x[i] \leq 9$, for $i = 1, 2, ..., n$), and the `Zero(k)` algorithm, which returns an array of ***k*** elements, all equal to zero:

```
Algorithm ceFace(x, n):
    f ← Zero(9)
    For i ← 1, n execute
        f[x[i]] ← f[x[i]] + 1
    EndFor
    i ← 9
    nr ← 0
    While i > 0 execute
        If f[i] = 0 then
            nr ← nr * 10 + i
        EndIf
        i ← i - 1
    EndWhile
    Return 10 * nr
EndAlgorithm
```

What does the given algorithm return?

A. A number formed from the digits of array ***x***
B. A number formed from the digits of array ***x***, with each digit used only once
C. The largest possible number formed using distinct digits that do not appear in array ***x***
D. The smallest possible number formed using distinct digits that do not appear in array ***x***

20. Consider the non-zero natural numbers ***n*** and ***m***, ($1 \leq n, m \leq 100$) and the ***matrix*** matrix with ***n*** rows and ***m*** columns, its elements being 0 or 1. Consider the algorithms `prelucrare(matrix, row, col, n, m)` and `num(matrix, n, m)`, where ***row*** and ***col*** are natural numbers ($1 \leq row \leq n$, $1 \leq col \leq m$).

```
Algorithm prelucrare(matrix, row, col, n, m):
    If row ≥ 1 AND row ≤ n AND col ≥ 1 AND col ≤ m AND matrix[row][col] = 1 then
        matrix[row][col] ← 0
        prelucrare(matrix, row - 1, col, n, m)
        prelucrare(matrix, row + 1, col, n, m)
        prelucrare(matrix, row, col – 1, n, m)
        prelucrare(matrix, row, col + 1, n, m)
    EndIf
EndAlgorithm
```

```
Algorithm num(matrix, n, m):
    c ← 0
    For row ← 1, n execute
        For col ← 1, m execute
            If matrix[row][col] = 1 then
                c ← c + 1
                prelucrare(matrix, row, col, n, m)
            EndIf
        EndFor
    EndFor
    Return c
EndAlgorithm
```

Considering that an island is made up of identical elements neighboring horizontally or vertically, which of the following statements are true?

A. If $n \neq m$ the algorithm `num(matrix, n, m)` does not check all the elements of the matrix.
B. For the matrix with 5 rows and 5 columns:
```
matrix =
1  1  0  0  0
1  1  0  0  0
0  0  1  0  0
0  0  0  1  1
0  0  0  1  1
```
the call `num(matrix, 5, 5)` returns 3.
C. The `num(matrix, n, m)` algorithm returns the number of islands consisting of zeros in the given matrix.
D. The `num(matrix, n, m)` algorithm returns the number of islands consisting of ones in the given matrix.

21. Consider two strings of characters ***r*** and ***s*** of length ***Lung*** ($1 \leq$ ***Lung*** ≤ 256). Consider the following algorithms:

- The `copiere(a, primul, ultimul)` algorithm returns the string consisting of the elements of string ***a***, starting from the ***primul*** position to the ***ultimul*** position inclusive.
- The `egale(a, b, k)` algorithm returns *True*, if strings ***a*** and ***b***, both of length ***k***, are identical, and *False* otherwise.
- The `lungime(a)` algorithm returns the length of string ***a***.
- The `concatenare(a, b)` algorithm returns the string obtained by concatenating string ***a*** with string ***b***, in this order.

State which of the following algorithms returns the value *True* if string ***r*** can be obtained by rotating string ***s*** 0, 1, or more times. For example, the string `"abcde"` can be obtained by rotating the string `"cdeab"`.

A.

```
Algorithm check(s, r, Lung):
    For i ← 1, Lung execute
        If egale(s, r, Lung) then
            Return True
        EndIf
        aux ← s[1]
        For j ← 2, Lung execute
            s[j - 1] ← s[j]
        EndFor
        s[Lung] ← aux
    EndFor
    Return False
EndAlgorithm
```

B.

```
Algorithm check(s, r, Lung):
    ss ← concatenare(s, s)
    i ← 1
    sf ← Lung + 1
    While i ≤ sf execute
        k ← i
        j ← 1
        While j ≤ Lung AND ss[k] = r[j] execute
            j ← j + 1
            k ← k + 1
        EndWhile
        If j > Lung then
            Return True
        EndIf
        i ← i + 1
    EndWhile
    Return False
EndAlgorithm
```

C.

```
Algorithm check(s, r, Lung):
    ss ← concatenare(r, s)
    i ← 1
    While i ≤ Lung execute
        k ← i
        j ← 1
        While j ≤ Lung AND ss[k] = r[j] execute
            j ← j + 1
            k ← k + 1
        EndWhile
        If j > Lung then
            Return True
        EndIf
        i ← i + 1
    EndWhile
    Return False
EndAlgorithm
```

D.

```
Algorithm check(s, r, Lung):
    pos1 ← 1
    ok ← False
    While r[pos1] ≠ s[1] execute
        pos1 ← pos1 + 1
    EndWhile
    If pos1 > 0 then
        ok ← egale(s, r, Lung)
    EndIf
    If NOT ok then
        pos2 ← Lung - pos1 + 1
        ok ← (r[1] = s[pos2])
        ss ← copiere(s, pos2, Lung)
        rr ← copiere(r, 1, pos1)
        ok ← ok AND egale(rr, ss, lungime(ss))
    EndIf
    Return ok
EndAlgorithm
```

22. Consider the algorithm `ceFace(a, n)` where ***n*** is a natural number ($2 <$ ***n*** $\leq 10^4$) and ***a*** is an array of ***n*** natural numbers (***a***[1], ***a***[2], ..., ***a***[***n***], $0 \leq$ ***a***[***i***] $\leq 10^4$ for ***i*** = 1, 2, ..., ***n***).

We consider the algorithm `nrPalindromuri(b, p, r)`, where ***b*** is an array of ***m*** natural numbers (***b***[1], ***b***[2], ..., ***b***[***m***], $0 \leq$ ***b***[***j***] $\leq 10^4$ for ***j*** = 1, 2, ..., ***m***, $2 <$ ***m*** $< 10^4$). The parameters ***p*** and ***r*** are natural numbers such that $1 \leq$ ***p*** $<$ ***r*** $\leq$ ***m***. The `nrPalindromuri(b, p, r)` algorithm returns the number of palindrome numbers in the ***b***[***p***], ..., ***b***[***r***] subarray of array ***b***.

```
Algorithm ceFace(a, n):
    b ← 0; c ← b; e ← 0; d ← 0
    For i ← 1, n - 2 execute
        If nrPalindromuri(a, i, i + 2) > 1 then
            If c = 0 then
                d ← i
            EndIf
            c ← c + 1
        Else
            If c > b then
                b ← c; e ← d
            EndIf
            c ← 0
        EndIf
    EndFor
    If c > b then
        b ← c; e ← d
    EndIf
    If b = 0 then
        Write 0, " ", 0
    Else
        Write e, " ", e + b + 1
    EndIf
EndAlgorithm
```

Which of the following statements are true?

A. If in the case of an array of length 10^4 the value 7381 7384 is displayed, it follows that among the 4 numbers located in the array in the range of positions [7381, ..., 7384] there are exactly two palindrome numbers.
B. If ***n*** = 12 and ***a*** = [11, 33, 45, 103, 121, 343, 33, 99, 100, 22, 44, 45] the `ceFace(a, n)` algorithm displays: 5 8
C. If at the end of the execution of the algorithm the value of ***b*** is 0, it follows that in array ***a*** there is no palindrome number.
D. If ***n*** = 12 and ***a*** = [11, 33, 45, 103, 121, 343, 33, 99, 100, 22, 44, 45] the `ceFace(a, n)` algorithm displays: 4 12

23. Consider algorithm `fun(a, b, len)`, where ***len*** is a natural number ($1 \leq$ ***len*** ≤ 100), and ***a*** and ***b*** are two arrays having the same length ***len*** (***a***[1], ***a***[2], ..., ***a***[***len***], ***b***[1], ***b***[2], ..., ***b***[***len***], $1 \leq$ ***a***[***i***], ***b***[***i***] $\leq$ ***len***, i = 1, 2, ..., ***len***).

```
Algorithm fun(a, b, len):
    For i ← 1, len execute
        k ← a[b[i]]
        a[b[i]] ← b[a[i]]
        b[a[i]] ← k
    EndFor
EndAlgorithm
```

Let ***len*** = 7, ***a*** = [6, 2, 5, 4, 1, 3, 4] and ***b*** = [1, 2, 3, 5, 6, 4, 4]. In the two arrays, before the execution of the algorithm `fun(a, b, len)` there are two elements having the same value, located on identical positions (***a***[2] = ***b***[2] and ***a***[7] = ***b***[7])).

Which of the following statements are true following the call `fun(a, b, len)`?

A. Arrays ***a*** and ***b*** will have identical elements at positions 3 and 6.
B. Arrays ***a*** and ***b*** will each have three elements having the same value, located on identical positions.
C. Array ***b*** will have the values: [1, 2, 3, 4, 6, 5, 4].
D. Array ***a*** will have the values: [4, 2, 6, 3, 6, 1, 4].

24. Consider the algorithm `calculeaza(v, b, n, i)`, where ***b***, ***n***, ***i*** are non-zero natural numbers ($1 \leq$ ***b***, ***n***, ***i*** $\leq 10^3$), and ***v*** is an array of ***n*** natural number elements (***v***[1], ***v***[2], ..., ***v***[***n***], $0 \leq$ ***v***[***i***] $\leq 10^3$, for ***i*** = 1, 2, ..., ***n***):

```
Algorithm calculeaza(v, b, n, i):
    If b = 0 then
        Return True
    EndIf
    If i = n then
        Return False
    EndIf
    Return calculeaza(v, b - v[i], n, i + 1) OR calculeaza(v, b, n, i + 1)
EndAlgorithm
```

For which of the following input data does the algorithm return *True*?

A. ***v*** = [3, 1, 7, 4, 2], ***b*** = 10, ***n*** = 5, ***i*** = 1
B. ***v*** = [2, 6, 4, 8, 12], ***b*** = 12, ***n*** = 5, ***i*** = 1
C. ***v*** = [3, 1, 7, 4, 2], ***b*** = 10, ***n*** = 5, ***i*** = 2
D. ***v*** = [2, 6, 4, 8, 12], ***b*** = 12, ***n*** = 5, ***i*** = 3

BABEŞ-BOLYAI UNIVERSITY
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Admission Exam – July 19th, 2024

Written Exam for Computer Science

GRADING AND SOLUTIONS

DEFAULT: 10 points

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