|  |  |  |  |
| --- | --- | --- | --- |
| **INTRODUCTION TO PROGRAMMING IN C (IS-FEE-10061S)**  **WEEK 07** | | | |
| **First Name** | **Last Name** | **Date** | **Points** |
|  |  | **11.04.2024** |  |

**Comments:**

* complete the data in the table above
* paste the program codes in the designed places
* send the file by the end of the day on which the next class will take place

|  |
| --- |
| **Program no. 1** |
| Declare a **N×M**-element array-of-int (**N, M** - symbolic constants defined using **#define** preprocessor directive). Write a program that performs the following operations:   1. write subsequent integers (1, 2, 3,…) in the array (row by row); print elements of the array; 2. write subsequent integers (1, 2, 3,…) in the array (column by column); print elements of the array; 3. write pseudo-randomly generated integers in the range of -9, 9 in the array; print elements of the array; 4. calculate and print the number of elements of the array: positive, negative, equal to zero; 5. calculate and print the number of array elements with even and odd values; 6. replace all negative elements of the array with the value zero; print elements of the array; 7. calculate and display the arithmetic mean of only those elements of  the array that are more than zero.   Example of program execution:  **1 2 3 4 5 6 7 8**  **9 10 11 12 13 14 15 16**  **17 18 19 20 21 22 23 24**  **1 4 7 10 13 16 19 22**  **2 5 8 11 14 17 20 23**  **3 6 9 12 15 18 21 24**  **4 1 9 1 -8 -1 4 0**  **-7 5 2 -5 -7 -7 -1 -3**  **-6 -9 -2 7 4 7 -2 8**  **The number of positive elements: 11**  **The number of negative elements: 12**  **The number of elements equal to zero: 1**  **The number of even elements: 10**  **The number of odd elements: 14**  **4 1 9 1 0 0 4 0**  **0 5 2 0 0 0 0 0**  **0 0 0 7 4 7 0 8**  **The arithmetic mean: 4.72727** |
|  |
| **Teacher's comments:** |
|  |

|  |
| --- |
| **Program no. 2** |
| Write a program that performs matrix multiplication. Use two matrices of order **N×M** and **M×K** respectively and find a resultant matrix of the order **N×K** (**N, M, K** - symbolic constants defined using **#define** preprocessor directive). Generate matrix elements pseudo-randomly. Print elements of all array.  Example of program execution:  **Matrix A(2 x 3):**  **1 2 2**  **3 2 1**  **Matrix B(3 x 4):**  **2 3 4 2**  **3 3 1 2**  **4 1 3 2**  **Matrix C(2 x 4) = A\*B:**  **16 11 12 10**  **16 16 17 12** |
| **Program code:** |
|  |
| **Teacher's comments:** |
|  |

|  |
| --- |
| **Program no. 3** |
| Declare a **N×M**-element array-of-int (**N, M** - symbolic constants defined using **#define** preprocessor directive). Write pseudo-randomly generated integers in the range of 0, 10 in the array. Print elements of the array. Find the array element whose sum of neighbors is the largest. The neighbors are the elements located on the left, on the right, above and below the element. Print indexes of this element.  Example of program execution:  **Matrix:**  **2 9 7 2 10 7**  **4 3 1 10 8 3**  **8 5 9 4 7 6**  **4 10 8 0 3 4**  **Sum of neighbors:**  **13 12 12 27 17 13**  **13 19 29 15 30 21**  **13 30 18 26 21 14**  **18 17 19 15 11 9**  **Max sum of neighbors: 30**  **Indexes: 1,4**  **Indexes: 2,1** |
| **Program code:** |
|  |
| **Teacher's comments:** |
|  |