|  |  |  |  |
| --- | --- | --- | --- |
| **INTRODUCTION TO PROGRAMMING IN C (IS-FEE-10061S)**  **WEEK 05** | | | |
| **First Name** | **Last Name** | **Date** | **Points** |
|  |  | **28.03.2024** |  |

**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 an **N**-element array-of-int (**N** **-** a symbolic constant defined using **#define** preprocessor directive). Write a program  that performs the following operations:   1. write subsequent integers (1, 2, 3, …, N) in the array; print elements of the array in one line; 2. write integers (N, N-1, …, 3, 2, 1) in the array; print elements of the array in one line; 3. write pseudo-randomly generated integers in the range of 0, 9 in the array; print elements of the array in one line; 4. print array elements with odd values; 5. print array elements with even values; 6. calculate and print the sum of all elements of the array; 7. calculate and print the arithmetic mean of all elements of the array; 8. find and print the largest and smallest value in the array; 9. using the scanf() function, enter the number **x**; check if **x** exists in the array; if so, display the index of the first element  equal to **x**; 10. print the number of occurrences of **x** in the array; 11. print the number of array elements less than **x** and the number of array elements more than **x**; 12. reverse the order of the array elements; print elements of the array; 13. sort the array elements in ascending order; print elements of the array in one line.   Example of program execution:  **1 2 3 4 5 6 7 8 9 10**  **10 9 8 7 6 5 4 3 2 1**  **5 7 8 2 4 4 2 7 1 9**  **Odd elements: 5 7 7 1 9**  **Even elements: 8 2 4 4 2**  **Sum: 49**  **Mean: 4.9**  **Min: 1**  **Max: 9**  **Enter x: 4**  **Index: 4**  **4 occurs 2 times**  **Less than x: 3**  **More than x: 5**  **9 1 7 2 4 4 2 8 7 5**  **1 2 2 4 4 5 7 7 8 9** |
|  |
| **Teacher's comments:** |
|  |

|  |
| --- |
| **Program no. 2** |
| Declare an **N**-element vector of integers. Write pseudorandom numbers from the range **[0, 10]** to the vector. Print vector elements on the screen. Calculate how many times each number appears in the vector.  Example of program execution:  **Vector elements:**  **10 0 2 0 7 1 5 10 3 3 0 4 2 1 1 8 7 3 7 6**  **Occurrences of each number:**  **0: 3**  **1: 3**  **2: 2**  **3: 3**  **4: 1**  **5: 1**  **6: 1**  **7: 3**  **8: 1**  **9: 0**  **10: 2** |
| **Program code:** |
|  |
| **Teacher's comments:** |
|  |

|  |
| --- |
| **Program no. 3** |
| There are two **N**-element arrays-of-int (**A** and **B**) storing pseudo-randomly generated integers in the range of **0, 99**:   1. create a **C** array storing the larger of the array elements from **A** and **B** in the corresponding position, 2. create a **D** array storing the sum of elements of arrays **A** and **B**, 3. calculate and print the **dot product** of the arrays **A** and **B**.   Declare the size of the arrays (**N**) as a symbolic constant defined using **#define** preprocessor directive. Print the elements of all arrays (**A**, **B**, **C**, **D**).  Example of program execution:  **A: 0 22 72 80 36**  **B: 33 30 59 98 70**  **C: 33 30 72 98 70**  **D: 33 52 131 178 106**  **Dot product: 15268** |
| **Program code:** |
|  |
| **Teacher's comments:** |
|  |