One-dimensional Arrays

A guide to prepare your submission

- For multiple-choice and essay questions: Present your answers to a PDF file and name the file as <StudentID>_Writing.pdf, where <StudentID> is your student number. Programming questions sometimes require brief explanation and that content should also presented here.
    - **Multiple-choice questions:** Pick the most probable choice and give your reasons (even for those you do not choose). If a question has multiple correct choices, it will be specified in the content.
    - **Essay questions:** Give a brief and focused answer that considers and explains every aspects of the problem
- For programming question: Write your source codes in C++/Python 3. Do not use any additional libraries/open source codes without permission.
    - **C++ programming language:** Name your Visual Studio solution such that the execution filename is <MSSV>_<Ci>.exe, where StudentID is your student number, C denotes programming questions, and i is the question index (e.g., C1, C2, etc.). Please submit the whole solution folder (after delete all bulky intermedia files). Do not include the execution file in your submission to avoid spreading dangerous virus.
    - **Python programming language:** name your Python files in a similar way to what described above
- TA will support solutions for multiple-choice and essay questions after the submission deadlines.
- Students can contact TA through email (Ms. Nguyễn Ngọc Thảo: nnthao@fit...).

A. Multiple-choice questions (5.0 marks)

A.1. Given a one-dimensional array of 9 elements. What is the index of the last element?

    a. 9

    b. 8

    c. 0

    d. It depends on specific contexts

A.2. Which of the following statements is used to access the seventh element in a one-dimensional array?

    a. a[6]

    b. a[7]

    c. a(7)

    d. a[8]

    e. None of the above

A.3. In the main function defined a one-dimensional array, `int A[100]`. This array is passed to a function. Which of the following statements is INAPPRORIATE for parameter declaration in a function prototype?

    a. int *P

    b. int P[]

    e. int P[int]

    c. int P[50]

    d. int P[100]

    f. None of the above

A.4. Which of the following one-dimensional array initialization is CORRECT?

    a. int my_array[5] = {5; 3; 4; 2; 7};

    b. int my_array[ ] = (5; 3; 4; 2; 7};

    c. int my_array[5] = {5, 3, 4, 2, 7};

    d. int my_array[ ] = (5, 3, 4, 2, 7};

    e. None of the above

A.5. What happens when accessing an element that is out of bound (of a one-dimensional array)? (Be aware that there may multiple correct options)

    a. This may accidentally modify the value of some other variable.

    b. Nothing because C/C++ will ignore all operations not inside the array

    c. Syntax error, the compiler will check this

    d. The array will automatically increase its size to adapt with the new access

    e. Run-time error, there may be an exception thrown, such as "access violation"

B. Essay questions (5.0 marks)

B.1. Given two one-dimensional arrays of integers, `int a[10]` and `int b[10]`. Is it possible to copy the content of a to b using the statement `b = a`?

B.2. A programmer writes a program that passes a one-dimensional array to a function and inside the function there are some operations that modify the values of the array. How should the array be passes such that

    a. The original content of the array is preserved after leaving the function.

    b. All modifications to the array are kept after leaving the function.

    c. The programmer reinitializes the array with a new size.

B.3. Tell me which IDE you are using to code C/C++ programs. If you use the statement `int a[10];` to declare a one-dimensional array, are all elements initially set to 0?

B.4. The following code segment aims to print the content of a one-dimensional array that has 100 elements. Identify all possible errors in the following code segment and fix them.

```
int a[100];
...    // assign values to array
for(int i=0; i<100; i++)
    cout <<  a(i+1);
```

B.5. What is the output of the following program?

```
void main(){
    int egArray[10] = { 2, 4, 6, 8, 10, 1, 3, 5, 7, 9 };
    int n = (sizeof(egArray) / sizeof(*egArray));
    for (int index = 0; index < n; index = index+2)
         cout << egArray[index];
}
```

3

C. Programming questions (10.0 marks)

Students do not have to check the validity of the input.

C.1. Write a function, `inFunc`, that asks the user to input 10 integers and stores these numbers in a one-dimensional array. Write two functions, `findMax` and `findMin`, that return the maximum and the minimum values in the array, respectively. Write a program that subsequently calls those functions and outputs the resulting findings.

C.2. Write a function that accepts a one-dimensional array of integers (and of course, its size) and then outputs the integer that has the highest number of occurrences (Output all if there are multiple such integers).

C.3. Write a function whose parameters include a one-dimensional array of integers (with size), an integer x, and a non-negative integer that is less than the array size. The function inserts x into the array at position pos. Inserting an integer at postion pos means moving elements from pos to the end of the array 1 unit to the right while dropping the last element. Output the array after insertion.

C.4. Write a function that checks whether a one-dimensional array satisfies either of the following orders

   a. Constant array: all elements are the same, a[0] = a[1] = …. = a[n-1]

   b. Increasing array: a[i] ≤ a[i+1] for all i and there exists at least one pair that a[i] < a[i+1]

   c. Decreasing array: a[i] ≥ a[i+1] for all i and there exists at least one pair that a[i] > a[i+1]

   d. [Advanced] Elements in the array can be divided into two subsequent sub-arrays such that the first sub-array is an increasing array while the other is a decreasing one.

   e. [Advanced] Elements at even positions constitute an increasing sub-array while elements at odd positions form a decreasing sub-array.

C.5. Write a function to sort elements in a one-dimensional array of integers in ascending order.