GECG10069 (561085) F25: Introduction to Programming (C++)

Lab 8 : Arrays

What you will learn from Lab 8

In this laboratory, you will understand how to use array.

TASK 8-1 : ARRAY AS ARGUMENT

✓ *Sorting* is used to sort an array with increasing order or decreasing order.

```
#include <iostream>
using namespace std;

void DisplayArray(int [],int);
void InsertionSort(int [],int);

int main() {
    const int MaxSize = 10;
     int array[MaxSize] = {21,15,12,24,9,30,27,6,3,18};
     DisplayArray(array,MaxSize);
     InsertionSort(array,MaxSize);
     DisplayArray(array,MaxSize);
}

void DisplayArray(int vec[], int size) {
    for (int idx = 0; idx < size; idx++)
        cout << vec[idx] << " ";
    cout << endl;
}

void InsertionSort(int vec[],int size) {
    for (int jdx = 1; jdx < size; jdx++) {
        int key = vec[jdx];
        int idx = jdx - 1;

        while ( idx >= 0 && vec[idx] > key) {
            vec[idx+1] = vec[idx];
            --idx;
        }

        vec[idx+1] = key;
    }
}
```

TASK 8-2 : 2D ARRAY

```
#include <iostream>
#include <iomanip>
using namespace std;

const int numrows = 3;
const int numcols = 4;
void showarray(int [][numcols]);
int main()
{

    int val[numrows][numcols]={ { 1, 2, 3, 4},
                                { 5, 6, 7, 8},
                                { 9,10,11,12} };
    cout << "Display the multiplied elements" << endl;
    showarray(val);
    return 0;
}
void showarray(int array[][numcols]){
    for(int row = 0; row < numrows; row++){
        for(int col = 0; col < numcols; col++){
            cout << setw(5)<< array[row][col] << " ";
       }
       cout << endl;
   }
}
```

EXERCISE 8-1: CYCLIC ROTATION

Description

Write a C++ program that performs a cyclic right rotation on an array of integers.

Given an array of size 5 and an integer K, rotate all elements to the right by K positions.

Elements shifted beyond the end of the array wrap around to the beginning.

Input

- The first line contains an integer, K.
- The second line contains 5 integers, representing the array elements.

Output

- Print the array after rotation as 5 integers separated by spaces.

Requirements / Notes

- The valid range is: $0 \leq K \leq 100$.
- You must use **basic arrays** (no STL containers).

Sample Input - 1
2 10 20 30 40 50
Sample Output - 1
40 50 10 20 30
Sample Input - 2
14 1 2 3 4 5
Sample Output - 2
2 3 4 5 1
Sample Input - 3
36 6 8 10 3 5
Sample Output - 3
5 6 8 10 3

EXERCISE 8-2: MATRIX PRODUCT SUM

Write a C++ program that reads two integers m and n, constructs two matrices: A and B.

$$A = \begin{bmatrix} m & m+1 & m+2 \\ m+3 & m+4 & m+5 \end{bmatrix}$$

$$B = \begin{bmatrix} n & n+3 & n+6 & n+9 \\ n+1 & n+4 & n+7 & n+10 \\ n+2 & n+5 & n+8 & n+11 \end{bmatrix}$$

computes the product C=A×B (matrix multiplication), and **outputs a single integer** equal to the sum of all entries of C.

Hint:

$$C_{ij} = \sum_{k=1}^{3} A_{ik} B_{kj}, \qquad C = A B \in \mathbb{Z}^{2\times 4}.$$

Rules / Notes

- Input: two integers: $-200 \leq m, n \leq 200$.
- Use only basic arrays (no STL containers).

Sample Input - 1
1 2
Sample Output - 1
646
Sample Input - 2
0 0
Sample Output - 2
346
Sample Input - 3
-1 4
Sample Output - 3
358