

DEPARTMENT OF COMPUTER ENGINEERING
CMPE110: Fundamentals of Computing and Programming
EXPERIMENT 6

Introduction to C++ Programming: One and two dimensional array

Objectives:

- 1) Understand how to edit, compile and execute C++ computer codes.
- 2) Understand C++ programming: One and two - dimensional array.

Task I:

A/Show the contents of the following array declarations.

- a) `int N1[5], N2[5];`
 `N1[2] = 4;`
 `N1[0] = N1[2];`
 `N2[4] = N1[0];`
- b) `float prices[5]={1.25, 3.50, 1.20, 2.99, 0.75};`
- c) `float prices[5]={1.25, 3.50};`
- d) `int months[]={1,2,3,4,5,6,7,8,9,10,11,12};`
- e) `int x[2][3]={1,2,3,3,2,1};`
- f) `int y[2][3]={{1,2,3},{3,2,1}};`

B)what is the output of

```
#include<iostream>
using namespace std;
int main(){
int M1[2][3]={1,5,3,2,1,3};
int i, j, s;

for (i=0; i<2; i++)
{s=0;
  for (j=0; j<3; j++)
    {s+=M1[i][j];}
  cout<<"A=["<<i<<"]="<<(s/3)<<endl;
}
return 0;}
```

Task II: Programming tasks

1) Consider the following code that finds the sum of elements of the integer array A:

```
#include<iostream>
#include<cmath>
using namespace std;
int main(){
#define MAX 5
int A[MAX],i,Sum=0;
cout << "Enter the element of the array A (5 elements): ";
for (i = 0; i < MAX; i++)
  cin >> A[i];
Sum = 0;
for (i = 0; i < MAX; i++)
  Sum += A[i];

cout << "The sum is " << Sum<< endl;
return 0;}
```

- a) Edit, compile and execute this code. Use the following input values for array A elements: 3 2 4 1 5.
- b) Modify the given code to find the average of the array A. **Note:** The average is computed by as sum of elements/number of elements
A sample run of the program must be as follows:

```
*** Program to calculate find the average of elements of an integer array ***
Enter the elements of the array A: 3 2 4 1 5
The average of the array A is 3.0.
```

- 2) The dot-product of the two arrays x[m] and y[m] is computed as:

$$\text{dot_product} = x[0]*y[0] + x[1]*y[1] + \dots + x[m-1]*y[m-1]$$

Write a C++ code to read two 1-D integer arrays with size m (x[m], y[m]) and computes their dot-product.

- 3) Write one C++ program to do the following tasks:
- Read the grades of 10 students from the keyboard. Store the grades in a one-dimensional array called "grade" of type integer.
 - Find the average of the grades
 - Find and print the maximum grade
 - Add to each students grades the following Bonus value: 100-maximum
 - Print from the monitor the students new grades

- 4) The following code finds the sum of two matrices M1 and M2 with the size of two rows and three columns:

```
#include<iostream>
#include<cmath>
using namespace std;
int main(){

float M1[2][3],M2[2][3],Result[2][3];
int row, col;

// Read matrix 1
for (row=0; row<2; row++)
    for (col=0; col<3; col++)
        cin >> M1[row][col];

// Read matrix 2
for (row=0; row<2; row++)
    for (col=0; col<3; col++)
        cin >> M2[row][col];

// Add the two matrices
for (row=0; row<2; row++)
    for (col=0; col<3; col++)
        Result[row][col]= M1[row][col] + M2[row][col];

// Display the result
for (row=0; row<2; row++)
```

```

{
    for (col=0; col<3; col++)
        cout << Result[row][col] << '\t';
    cout << endl;
}

return 0;}

```

- a) Edit, compile and execute this code. Use the following input values for the elements of M1 and M2:

$$M1 = \begin{pmatrix} 1 & -5 & 3 \\ 2 & -1 & -3 \end{pmatrix}, M2 = \begin{pmatrix} 3 & 2 & 4 \\ -2 & -3 & 7 \end{pmatrix}$$

- b) Modify this code to find the maximum element of the matrix Result.
c) How can you find the average of each row of the matrix Result?

- 5) The expectation value $\langle X \rangle$ of M numbers X_i ($i = 1, \dots, M$) is defined by

$$\langle X \rangle = \frac{1}{M} \sum_{i=1}^M X_i$$

Write a **C++** code that reads integer numbers from the keyboard as input and then computes and prints the expectation value of those numbers that are in the following range [0:100], numbers that are not in this range will not be included [use `continue` statement]. Also, the program should be terminated if the number -1000 is entered [use `break` statement].

- 6) **Exercise** - Write a C code that will read the age of 10 students in a one dimensional array and then order the array in ascending order.

Use the following input: 23, 25, 18, 23, 16, 19, 22, 21, 27, and 18