

CMPE110 - Experiment 7 (Functions and Arrays Revision)

Q1) Let **A** be a 1-D array which contains 100 **integer** elements. Complete the following code to compute the norm of **A** which is defined as

$$\text{Norm_A} = \sqrt{A_0^2 + A_1^2 + \dots + A_{99}^2} = \sqrt{\sum_{i=0}^{99} A_i^2}$$

```
#include<iostream>
#include<cmath>
using namespace std;
int main(){//declarations
double Norm_A,Sum;

int i;

.....
// read the elements of the array A

.....

.....

//compute the sum of the square of the elements, i. e,  $\text{Sum} = \sum_{i=0}^{99} A_i^2$ 
.....
.....
.....

// compute the norm of A: Norm_A

.....
//print the norm of A on the monitor

.....
return 0;}
```

Q2) Complete the missing statements in the following program that defines a function to find the sum of all the integers between x and y (including x and y).

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

```
int Function2(int x,int y)
```

```
{
.....
.....
.....
.....
}
```

```
int main( )
```

```
{
    int a, b, result;
    cout<<"Enter two integer numbers:";
    cin>> a >> b;
```

```
// Call the function with a and b values and assign the returned
// value to 'result'
```

```
.....
    cout<<"Result = "<< result <<endl;
return 0;
}
```

Q3) a) Trace the following code and show its output. Use the following value of N: 5

```
#include<iostream>
using namespace std;
//function prototype
int F1(int);
// function definition
int F1(int M)
{
int r,i;
r = 0;
for (i = 1; i <= M; i++)
    {if (i%2==1)
        r+= i;
    }
return r;
}
// main function
int main()
{int N,Result;
cout << "Enter an integer number: ";
cin >> N;
Result=F1(N);
cout << "Result= " << Result << endl;
return 0;}
```

Trace section

| <u>N</u> | <u>M</u> | <u>i</u> | <u>i<=M</u> | <u>i%2==1</u> | <u>r</u> | <u>Result</u> |
|----------|----------|----------|----------------|---------------|----------|---------------|
|----------|----------|----------|----------------|---------------|----------|---------------|

Output section

b) Explain what does this code do?

.....

Q4) Complete the following code to convert a given temperature from Celsius (C) to Fahrenheit (F) or from Fahrenheit (F) to Celsius (C). Some useful formulas:

$$C = (F - 32) \times \frac{5}{9} \quad \text{and} \quad F = \frac{9}{5} \times C + 32$$

The function DisplayMessage should display the following messages:

Enter 1 to Convert from Celsius to Fahrenheit.

Enter 2 to convert from Fahrenheit to Celsius.

```
#include<iostream>
using namespace std;
void DisplayMessage(void);
double Convert_CtoF(double);
double Convert_FtoC(double);
int main()
{double C, F;
//activate the function Display message
.....
cin>>Choice;
if (choice == 1){
    {cout<<"Enter Celsius temperature:";
    cin>>C;
    //activate Convert_CtoF to compute Fahrenheit temperature
    .....
    cout<<"The equivalent Fahrenheit temperature is " << F;}
else if (choice == 2){
    {cout<<"Enter Fahrenheit temperature:";
    cin>>F;
    //activate Convert_FtoC to compute Celsius temperature
    .....
    cout<<"The equivalent Celsius temperature is " << C;}
else
    cout<<"Wrong Choice is entered.";
return 0;}
//DisplayMessage Function definition
.....
.....
.....
.....
// Convert_CtoF Function definition
.....
.....
.....
.....
// Convert_FtoC Function definition
.....
.....
```

Q5) Consider the following code which reads from the keyboard the value of an integer variable n and the value of a double variable x and then computes the result of the following formula:

$$f = 1 + x + x^2 + \dots + x^n$$

```

#include<iostream>
using namespace std;
int main()
{
int n,i;

double x,f=1.0,t=1.0;

cin >> n >> x ;

for (i = 1; i <= n; i++)
{
t*=x;
f+=t;
}

cout << "f=" << f << endl;
return 0;}

```

```

#include<iostream>
using namespace std;
// function definition
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
// main function
int main(){int n,m,k;
double x,y,z;
..... //
// Compute f
.....;
cout<<"f="<< f;
return 0;}

```

It is required to write a code to compute

$$f = \frac{(1 + x + x^2 + \dots + x^n)(1 + y + y^2 + \dots + y^m)}{1 + z + z^2 + \dots + z^k}$$

Where n, m, and k are integers and x, y, and z are double variables. Complete the given C++ code to do this task.

Note: To compute the series, it is required to use a function with the following prototype:

```
double Series(double , int);
```

Q6) Let **A** be a 1-D integer array which contains 5 elements, fill in the following code to find the maximum element and its position.

```
#include<iostream>
using namespace std;
int main(){//declarations
int MaxVlau, MaxPos,i;
```

| |
|---|
| <p>A sample run of the code can be</p> <pre>Enter the array elements: 15 5 25 10 7 The Maximum is 25 The position of the maximum is 3</pre> |
|---|

```
.....
// read the array elements from the Keyboard
```

```
.....
.....
```

```
// Assume that Maxvalue be the first element and its position is
// the MaxPos
```

```
.....
.....
```

```
// find the maximum element and its position
```

```
.....
.....
.....
.....
```

```
// print the maximum element and its position
```

```
.....
.....
```

```
return 0;}
```

Q7) The following code is designed to find the maximum of three numbers. Complete the missing parts to achieve this task.

```
// Finding the maximum of three floating-point numbers.
#include <iostream>
using namespace std;
// function prototype
double maximum( double, double, double );
int main(){double number1, number2, number3;
cout << "Enter three floating-point numbers: ";
cin >> number1 >> number2 >> number3;
//Activate the function maximum to find the largest number
cout<<"Maximum is:"<< .....;
return 0; }
// function maximum definition;
double maximum( double x, double y, double z )
{ double max;
//find the largest number
.....
.....
.....
return max; }
```

Q8) Consider the following C++ code that computes the average of each row in a 2-D array with 10 rows and 10 columns. Complete the missing statements to achieve this task.

```
#include <iostream>
using namespace std;
int main() {
const int Nrow = 4;
const int Ncolumn = 4;
// provide the 2-D array declaration with the name A
// and with the given size
int A[Nrow][Ncolumn];
int Sum;
// read the elements of the matrix A
.....
.....
.....
// Compute the average of each row
for ( int i = 0; i < Nrow; i++ )
{
.....
.....
.....
.....
.....
.....
}
return 0; }
```