LAB EXERCISE 3

TOPIC: FUNCTIONS

NAME: KAVINESH REDDY A/L GOPALAKRISHNAN
MATRIC NO: A24CS0092
SECTION: 02

QUESTION 1

Describe the difference between predefined function and programmer-defined function?

Predefined Function	Programmer-defined Function
Predefined function aka "built-in" functions come with the compiler.	Programmer-defined function aka User-defined functions are created by you, the programmer. Commonly used to break a problem down into small manageable pieces.
The source code (definition) for library functions does NOT appear in your program.	Programmer-defined function require explicit definition and implementation in the program.
To use a predefined function, you simply need to include the proper header file and know the name of the function that you wish to use.	Program must include either prototype or full function definition before any call to the function – compiler error otherwise.

QUESTION 2

Write a statement to calculate the equation or to convert the statement below using function from library.

a) Square root of y.

```
#include <cmath>
double result = sqrt(y);
```

b) x to the power of y.

```
#include <cmath>
double result = pow(x,y);
```

c) cos x.

```
#include <cmath>
double result = cos(x);
```

d) Change character to uppercase.

```
#include <cctype>
char change=toupper(character);
```

e) Copy the string of x into string y.

```
#include <cstring>
Strcpy(y,x);
```

QUESTION 3

What is the difference between local variable, global variable, global constant and static local variable?

	DIfference
Local Variable	Variables defined inside a function are local to that function. They are hidden from the statements in other functions, which normally cannot access them.
Global Variable	A global variable is any variable defined outside all the functions in a program. Global variable can be accessed by all functions that are defined after the global variable is defined
Global Constant	Global constants defined for values that do not change throughout the program's execution. The constants are then used for those values throughout the program.
Static Local Variable	Local variables only exist while the function is executing. When the function terminates, the contents of local variables are lost. static local variables retain their contents between function calls. static local variables are defined and initialized only the first time the function is executed. 0 is the default initialization value.

QUESTION 4

Given the following coding, fill in the blank with the “terms” of function as a comment.

```
#include <iostream>
using namespace std;
int average(int, int, int); //Function Prototype
int main()
{
      int x, y, z, avrg;
      cout << "Please enter three numbers:" << endl;
      cin >> x >> y >> z;
      avrg = average (x, y, z); //Function Call
      cout << "The average of the given three numbers is: " <<
     avrg << endl;
      return 0;
}
int average(int a, int b, int c) //Function Header
{
      int sum, avrg2;
      sum = a + b + c;
      avrg2 = sum / 3;
      return avrg2; //return type
}
```

QUESTION 5

Find the errors in the following given code.

```
#include <iostream>
#include <cmath> //error1
using namespace std;
int average(int,int,int);//error2
int power (int p); //error3
int main()
{
 int x, y, z, avrg, powerOf;
 cout << "Please enter three numbers:" << endl;
 cin >> x >> y >> z;
 avrg = average (x,y,z); //error 4
 cout<<"The average of the given three numbers is:"<<avrg<<endl;
 powerOf=power (avrg); //error5
 cout << "The average number to the power of two is: " << powerOf
<< endl; //error6
return 0;
}
int average(int a, int b, int c) //error 7
{
 int sum, avrg2;
 sum = a + b + c;
 avrg2 = sum / 3;
 return avrg2; //error8
}
int power (int p)
{
 int pOf;
 pOf = pow(p,2);
 return pOf; //error 9
}
```

QUESTION 6

Write a C++ program to calculate a rectangle's area. The program consists of the following function:

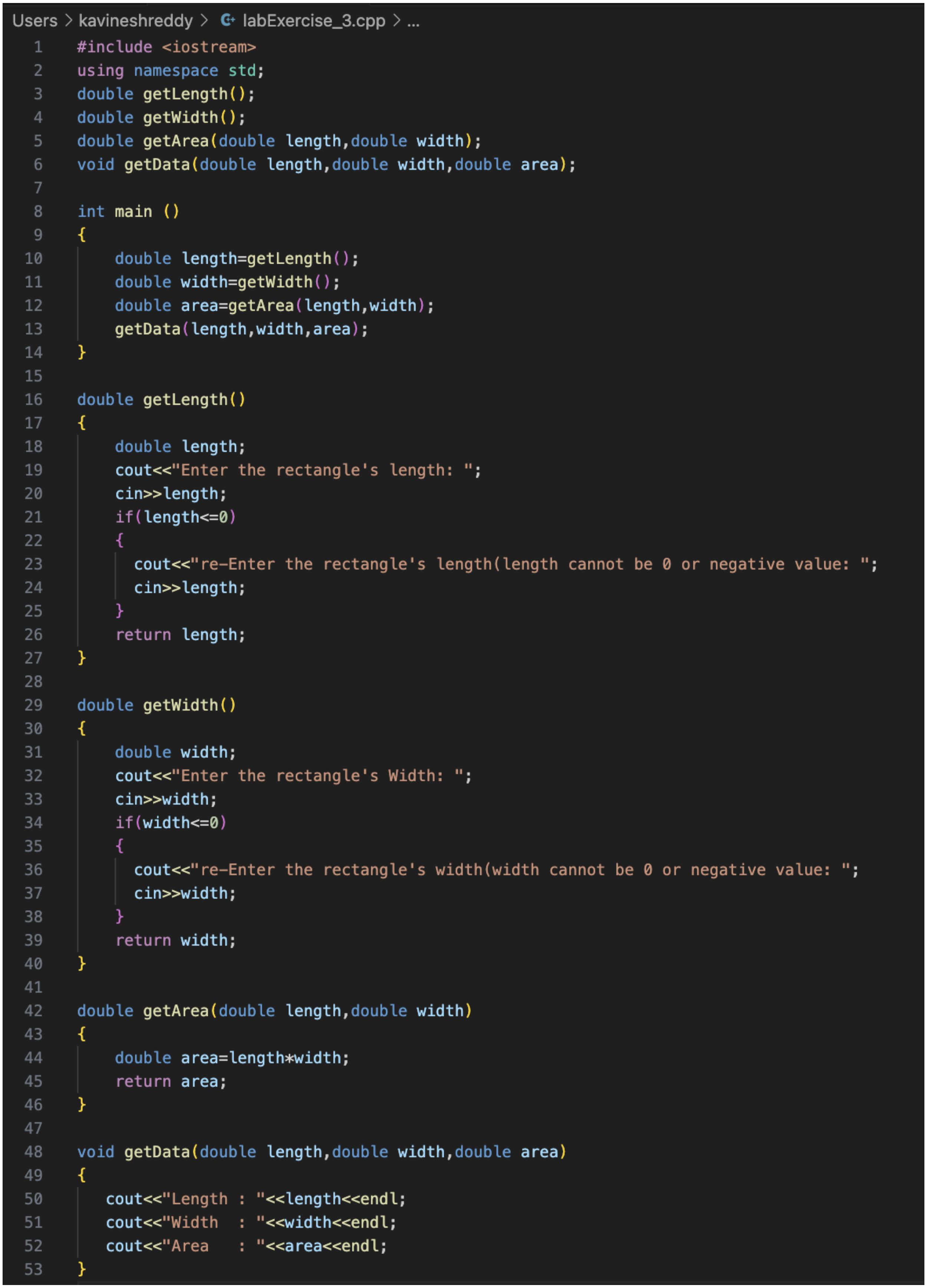
- `getLength` – This function should ask the user to enter the rectangle's length, and then returns that value as a double
- `getWidth` – This function should ask the user to enter the rectangle's width, and then returns that value as a double.
- `getArea` – This function should accept the rectangle's length and width as arguments and return the rectangle's area.
- `displayData` – This function should accept the rectangle's length, width and area as arguments, and display them in an appropriate message on the screen.
- `main` – This function consists of calls to the above functions.

For Question 6, provide the answer in .cpp file.

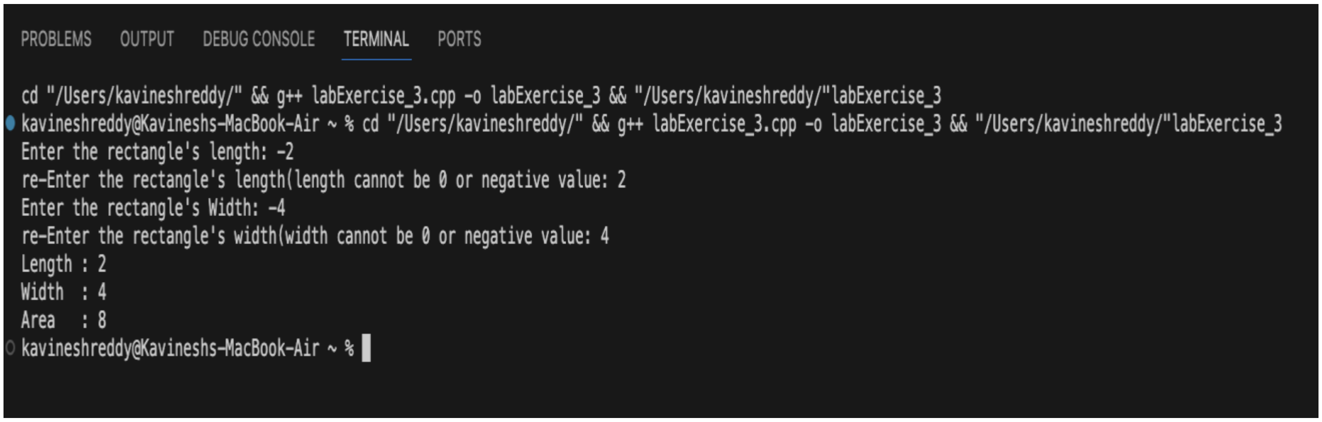
```
#include <iostream>
using namespace std;
double getLength();
double getWidth();
double getArea(double length,double width);
void getData(double length,double width,double area);

int main ()
{
   double length=getLength();
   double width=getWidth();
   double area=getArea(length,width);
   getData(length,width,area);
}

double getLength()
```

```
{
    double length;
    cout<<"Enter the rectangle's length: ";
    cin>>length;
    if(length<=0)
    {
      cout<<"re-Enter the rectangle's length(length cannot be 0 or negative value: ";
      cin>>length;
    }
    return length;
}

double getWidth()
{
    double width;
    cout<<"Enter the rectangle's Width: ";
    cin>>width;
    if(width<=0)
    {
      cout<<"re-Enter the rectangle's width(width cannot be 0 or negative value: ";
      cin>>width;
    }
    return width;
}

double getArea(double length,double width)
{
    double area=length*width;
    return area;
}
```

```
void getData(double length,double width,double area)
{
   cout<<"Length : "<<length<<endl;
   cout<<"Width  : "<<width<<endl;
   cout<<"Area   : "<<area<<endl;
}
```



```cpp
#include <iostream>
using namespace std;
double getLength();
double getWidth();
double getArea(double length,double width);
void getData(double length,double width,double area);

int main ()
{
    double length=getLength();
    double width=getWidth();
    double area=getArea(length,width);
    getData(length,width,area);
}

double getLength()
{
    double length;
    cout<<"Enter the rectangle's length: ";
    cin>>length;
    if(length<=0)
    {
      cout<<"re-Enter the rectangle's length(length cannot be 0 or negative value: ";
      cin>>length;
    }
    return length;
}

double getWidth()
{
    double width;
    cout<<"Enter the rectangle's Width: ";
    cin>>width;
    if(width<=0)
    {
      cout<<"re-Enter the rectangle's width(width cannot be 0 or negative value: ";
      cin>>width;
    }
    return width;
}

double getArea(double length,double width)
{
    double area=length*width;
    return area;
}

void getData(double length,double width,double area)
{
   cout<<"Length : "<<length<<endl;
   cout<<"Width  : "<<width<<endl;
   cout<<"Area   : "<<area<<endl;
}
```



```
cd "/Users/kavineshreddy/" && g++ labExercise_3.cpp -o labExercise_3 && "/Users/kavineshreddy/"labExercise_3
kavineshreddy@Kavineshs-MacBook-Air ~ % cd "/Users/kavineshreddy/" && g++ labExercise_3.cpp -o labExercise_3 && "/Users/kavineshreddy/"labExercise_3
Enter the rectangle's length: -2
re-Enter the rectangle's length(length cannot be 0 or negative value: 2
Enter the rectangle's Width: -4
re-Enter the rectangle's width(width cannot be 0 or negative value: 4
Length : 2
Width  : 4
Area   : 8
kavineshreddy@Kavineshs-MacBook-Air ~ % 
```