**Chapter 7: User-Defined Functions II**

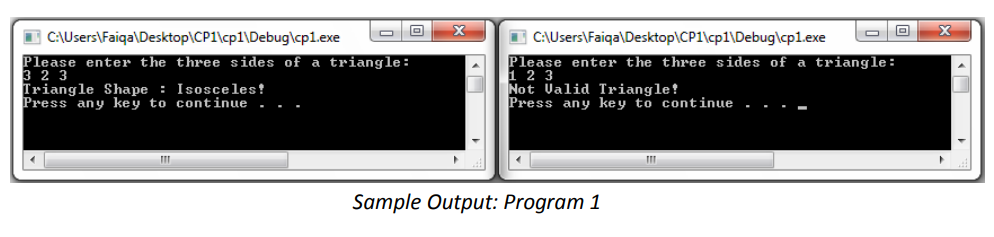
**Laboratory Exercises (3)**

**EXERCISES Solutions**

**Enumerations and Function Overloading**

**KEYWORDS: enum, function overloading, default parameter**

**Program 1:** Define an enumeration type, triangleType, that has the values scalene, isosceles, equilateral, and noTriangle. Write a function, triangleShape, that takes as parameters three numbers, each of which represents the length of the sides of the triangle. The function should return the shape of the triangle. (Note: In a triangle, the sum of the lengths of any two sides is greater than the length of the third side).



**#include<iostream>**

**using namespace std;**

**enum triangleType{scalene, isosceles , equilateral, noTriangle};**

**triangleType triangleShape(int x, int y, int z)**

**{**

**if((x + y <= z) || (y + z <= x) || (x + z <= y))**

**{**

**return noTriangle;**

**}**

**if((x==y) && (x==z))**

**{**

**return equilateral;**

**}**

**if((x==y) || (y==z) || (x==z))**

**{**

**return isosceles;**

**}**

**return scalene;**

**}**

**void printTriangleShape(triangleType shape)**

**{**

**switch (shape)**

**{**

**case scalene:**

**cout<<"Triangle Shape: Scalene!"<<endl;**

**break;**

**case isosceles:**

**cout << "Triangle Shape: Isosceles!"<< endl;**

**break;**

**case equilateral:**

**cout << "Triangle Shape: Equilateral!" << endl;**

**break;**

**case noTriangle:**

**cout<<"Not a Valid Triangle!"<<endl;**

**break;**

**}**

**}**

**int main()**

**{**

**int x, y, z;**

**cout<<"Please enter thr three sides of a tringle:"<<endl;**

**cin>>x>>y>>z;**

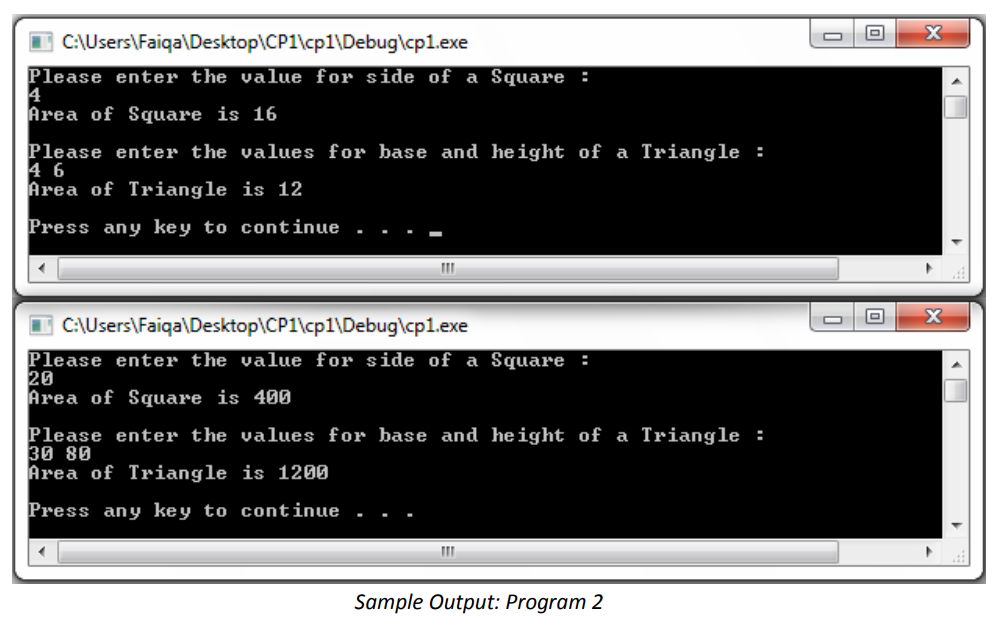
**triangleType t;**

**t = triangleShape( x, y , z);**

**printTriangleShape(t);**

**}**

**Program 2:** Write overloaded function for calculating the area of square and rectangle named area (int side) and area (int base, int height) respectively. Write a program that accept side of a square as input and returns its area using the overloaded function named area. The program also accepts base and height of a rectangle and returns its area using the overloaded function named area.



#include<iostream>

using namespace std;

double area(int side)

{

return side \* side;

}

double area(int base, int height)

{

return 0.5 \* base \* height;

}

int main()

{

int side, base, height;

cout<<"Please enter the value for side of a Square :"<<endl;

cin>>side;

cout<<"Area of Square is "<<area (side) <<endl<<endl ;

cout<<"Please enter the values for base and height of a Triangle: "<<endl;

cin>>base>>height;

cout<<"Area of Triangle is "<<area (base, height) <<endl<<endl;

return 0;

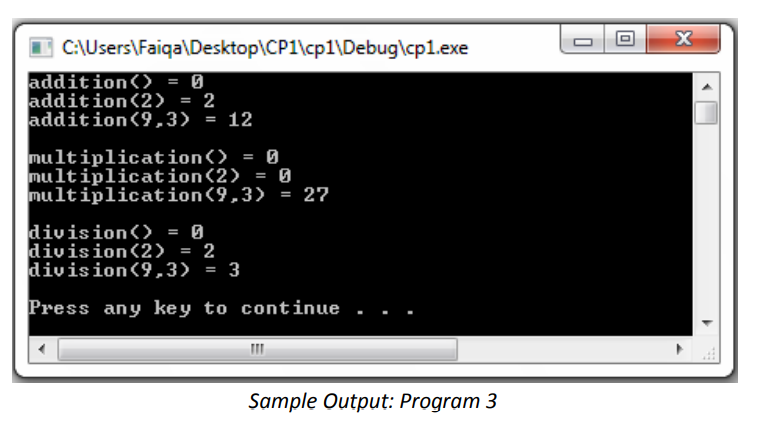
}

**Program 3:** Write functions with default argument for addition, multiplication, and division of two numbers. For addition and multiplication, the default values for two numbers will be 0. For division the default value for first and second arguments will be zero and one respectively. Use the above function for finding addition, multiplication, and division when

1) No arguments are supplied,

2) First argument is 2 and second argument is not supplied,

3) Arguments are 9 and 3.



#include<iostream>

using namespace std;

int addition (int a = 0, int b = 0)

{

return a + b;

}

int multiplication (int a = 0, int b = 0)

{

return a \* b;

}

double division (int a = 0, int b = 1)

{

return a / b;

}

int main()

{

cout<<"addition() = "<<addition () <<endl;

cout<<"addition (2) = "<<addition (2) <<endl;

cout<<"addition (9,3) = "<<addition (9,3)<<endl<<endl;

cout<<"multiplication() = "<<multiplication () <<endl;

cout<<"multiplication (2) = "<<multiplication (2) <<endl;

cout<<"multiplication (9,3) = "<<multiplication (9,3) <<endl<<endl;

cout<<"division() = "<<division () <<endl;

cout<<"division (2) = "<<division (2) <<endl;

cout<<"division (9,3) = "<<division (9,3)<<endl<<endl;

}