

Activity No. 1.2	
Hands-on Activity 1.2 Basic C++ Programming	
Course Code: CPE010	Program: Computer Engineering
Course Title: Data Structures and Algorithms	Date Performed: 9/9/24
Section: CPE21S4	Date Submitted: 9/10/24
Name(s): Santos, Andrei R.	Instructor: Professor Maria Rizette Sayo
6. Output	
SECTIONS	ANSWER
Header File Declaration Section	#include <iostream>
Global Declaration Section	None
Class Declaration and Method Definition Section	<pre> class Triangle{ private: double totalAngle, angleA, angleB, angleC; public: Triangle(double A, double B, double C); void setAngles(double A, double B, double C); const bool validateTriangle(); }; </pre>
Main Function	<pre> int main(){ //driver code Triangle set1(40, 30, 110); if(set1.validateTriangle()){ std::cout &lt;&lt; "The shape is a valid triangle.\n"; } else { std::cout &lt;&lt; "The shape is NOT a valid triangle.\n"; } return 0; } </pre>
Method Definition	<pre> Triangle::Triangle(double A, double B, double C) { angleA = A;  angleB = B; angleC = C; totalAngle = A+B+C; } void Triangle::setAngles(double A, double B, double C) { angleA = A; angleB = B; angleC = C; totalAngle = A+B+C; } </pre>

```
}  
const bool Triangle::validateTriangle() {  
    return (totalAngle <= 180);  
}
```

## 7. Supplementary Activity

1.

Create a C++ program to swap the two numbers in different variables.

```
#include <iostream>  
using namespace std;  
  
int main()  
{  
    int numb1;  
    int numb2;  
    int temp;  
    cout << "Enter number 1: "; cin >> numb1;  
    cout << "Enter number 2: "; cin >> numb2;  
    cout << "The numbers are: " << numb1 << " " << numb2 << endl;  
    temp = numb1;  
    numb1 = numb2;  
    numb2 = temp;  
  
    cout << "The numbers are now swapped: " << numb1 << " " << numb2 << endl;  
    return 0;  
}
```

2.

Create a C++ program that has a function to convert temperature in Kelvin to Fahrenheit.

```
#include <iostream>  
using namespace std;  
  
void KelvtoFahr(double Kelv, double Fahr)  
{  
    Fahr = (Kelv - 273.15) * 9/5 + 32;  
}  
  
int main()  
{  
    double Kelv;  
    double Fahr;  
    cout << "Enter Kelvin temperature: ";  
    cin >> Kelv;  
    KelvtoFahr(Kelv, Fahr);  
    cout << "The temperature of Kelvin to Fahrenheit is: " << Fahr << " F" << endl;
```

```
    return 0;
}
```

3.

Create a C++ program that has a function that will calculate the distance between two points.

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

```
void calcthedist(double x_1, double y_1, double x_2, double y_2)
{
    double dist = sqrt(pow(x_2 - x_1, 2) + pow(y_2 - y_1, 2));
    cout << "Now, this is the distance between the given two points: " << dist << endl;
}
```

```
int main()
{
    double a1x_1;
    double a2x_2;
    double b1y_1;
    double b2y_2;
    cout << "Enter your coordinates for x1: "; cin >> a1x_1;
    cout << "Enter your coordinates for y1: "; cin >> b1y_1;
    cout << "Enter your coordinates for x2: "; cin >> a2x_2;
    cout << "Enter your coordinates for y2: "; cin >> b2y_2;
    calcthedist(a1x_1, b1y_1, a2x_2, b2y_2);
    return 0;
}
```

4. Modify the code given in ILO B and add the following functions:

a. A function to compute for the area of a triangle

b. A function to compute for the perimeter of a triangle

c. A function that determines whether the triangle is acute-angled, obtuse-angled or 'others.'

```
#include <iostream>
#include <cmath>
```

```
class Triangle
{
private:
    double angleA;
    double angleB;
    double angleC;
    double sideA;
    double sideB;
```

```

double sideC;

public:
    Triangle(double A, double B, double C, double a, double b, double c);
    void Tria();
    void comparea();
    void compperim();
    void dettype();
};

Triangle::Triangle(double A, double B, double C, double a, double b, double c)
{
    angleA = A;
    angleB = B;
    angleC = C;
    sideA = a;
    sideB = b;
    sideC = c;
}

void Triangle::Tria()
{
    if (angleA + angleB + angleC == 180) {
        std::cout << "The shape is a valid triangle.\n";
    } else {
        std::cout << "The shape is NOT a valid triangle.\n";
    }
}

void Triangle::comparea()
{
    double s = (sideA + sideB + sideC) / 2;
    double area = std::sqrt(s * (s - sideA) * (s - sideB) * (s - sideC));
    std::cout << "Area: " << area << "\n";
}

void Triangle::compperim()
{
    double perimeter = sideA + sideB + sideC;
    std::cout << "Perimeter: " << perimeter << "\n";
}

void Triangle::dettype()
{
    if (angleA < 90 && angleB < 90 && angleC < 90) {
        std::cout << "Type: Acute-angled\n";
    } else if (angleA > 90 || angleB > 90 || angleC > 90) {
        std::cout << "Type: Obtuse-angled\n";
    } else {
        std::cout << "Type: Others\n";
    }
}

```

```

}

int main()
{
    Triangle set1(36, 124, 20, 18, 24, 27);
    set1.Tria();
    set1.comparea();
    set1.compperim();
    set1.detttype();
    return 0;
}

```

## 8. Conclusion

I have regained some knowledge regarding the C++ programming language. As of now, I am starting to understand other concepts, though some are still a little confusing. That is why I am reviewing some certain topics. However, I do remember the necessary commands to include before executing the program in a compiler, such as `#include <iostream>`, using namespace `std`;, and other `#include` libraries needed for specific programs. Regarding the procedure, it is different from Python, which is more concise and only needs minimal input to run or interpret the data. Last semester, we focused on and tackled the Python programming language. As for the activity, I need to review and understand the final part, especially where angles are involved. Overall, learning, understanding, and mastering C++ along with other programming languages would greatly benefit my studies and future learning.

## 9. Assessment Rubric