Activity No. 1	
REVIEW OF CPP PROGRAMMING	
Course Code: CPE010	**Program:** Computer Engineering
Course Title: Data Structures and Algorithms	**Date Performed:** 09 09 2024
Section: CpE21S4	**Date Submitted:** 09 09 2024
Name(s): Prince Wally G. Esteban	**Instructor:** M. Rizette Sayo

6. Output

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

int count = 0;


class rectangle{
private:
double recLength, recWidth;
public:
rectangle(double L, double W);
void setLength(double L);
void setWidth(double W);
double getPerimeter();
};

int main(){
rectangle shape1(2, 5);
std::cout << "The perimeter of the rectangle is " <<
shape1.getPerimeter() << ".\n";
std::cout << count << " number of objects created.";
return 0;
}

rectangle::rectangle(double L, double W) {
recLength = L;
recWidth = W;
count++;
}
void rectangle::setLength(double L) {
recLength = L;
}
void rectangle::setWidth(double W) {
recWidth = W;
}
double rectangle::getPerimeter() {
return (2*recLength) + (2*recWidth);
}
```

```
The perimeter of the rectangle is 14.
1 number of objects created.
```

7. Supplementary Activity

1.

```cpp
#include <iostream>

using namespace std;

int main() {
  int num1, num2, temp;

  //input
  cout << "Enter the first number: ";
  cin >> num1;

  cout << "Enter the second number: ";
  cin >> num2;

  cout << endl;

  //swapping
  temp = num1;
  num1 = num2;
  num2 = temp;

  //output
  cout << "After swapping:\n";
  cout << "First number: " << num1 << endl;
  cout << "Second number: " << num2 << endl;

  return 0;
}
```

```
Enter the first number: 7
Enter the second number: 2

After swapping:
First number: 2
Second number: 7
```

2.

```cpp
#include <iostream>

using namespace std;

// Function to convert Kelvin to Fahrenheit (The formula of K to F is F = 9/5(K - 273) + 32)
double kelvinToFahrenheit(double kelvin) {
  return (kelvin - 273.15) * 9 / 5 + 32;
}

int main() {
  double kelvin, fahrenheit;

  // Get Kelvin temperature input from the user
  cout << "Enter temperature in Kelvin: ";
  cin >> kelvin;

  // Convert Kelvin to Fahrenheit
  fahrenheit = kelvinToFahrenheit(kelvin);

  // Display the result
  cout << kelvin << " Kelvin is equal to " << fahrenheit << " Fahrenheit." << endl;

  return 0;
}
```

```
Run  Ask AI  7s on 14:34:21, 09/09
Enter temperature in Kelvin: 243
243 Kelvin is equal to -22.27 Fahrenheit.

Run  Ask AI  3s on 14:34:35, 09/09
Enter temperature in Kelvin: 459
459 Kelvin is equal to 366.53 Fahrenheit.

Run  Ask AI  3s on 14:34:42, 09/09
Enter temperature in Kelvin: 333
333 Kelvin is equal to 139.73 Fahrenheit.

Run  Ask AI  2s on 14:35:01, 09/09
Enter temperature in Kelvin: 275
275 Kelvin is equal to 35.33 Fahrenheit.
 Enter temperature in Kelvin: 275
275 Kelvin is equal to 35.33 Fahrenheit.
```

3.

main.cpp

```cpp
#include <iostream>
#include <cmath> // For mathematical functions

using namespace std;

// Function to calculate distance between two points
double calculateDistance(double x1, double y1, double x2, double y2) {
  // Calculate the difference in x-coordinates and y-coordinates
  double xDiff = x2 - x1;
  double yDiff = y2 - y1;

  // Calculate the distance using the distance formula
  double distance = sqrt(xDiff * xDiff + yDiff * yDiff);

  return distance;
}

int main() {
  double x1, y1, x2, y2, distance;

  // Get input for the coordinates of the two points
  cout << "Enter the x-coordinate of the first point: ";
  cin >> x1;
  cout << "Enter the y-coordinate of the first point: ";
  cin >> y1;

  cout << "Enter the x-coordinate of the second point: ";
  cin >> x2;
  cout << "Enter the y-coordinate of the second point: ";
  cin >> y2;

  // Calculate the distance
  distance = calculateDistance(x1, y1, x2, y2);

  // Display the distance
  cout << "The distance between the two points is: " << distance << endl;

  return 0;
}
```

```
Show Only Latest  Clear History
Run  Ask AI  31s on 14:40:23, 09/09
Enter the x-coordinate of the first point: 24
Enter the y-coordinate of the first point: 50
Enter the x-coordinate of the second point: 51
Enter the y-coordinate of the second point: 43
The distance between the two points is: 27.8927
 Enter the x-coordinate of the first point: 24
Enter the y-coordinate of the first point: 50
Enter the x-coordinate of the second point: 51
Enter the y-coordinate of the second point: 43
The distance between the two points is: 27.8927

Run  Ask AI  3s on 14:42:17, 09/09
Enter the x-coordinate of the first point: 1
Enter the y-coordinate of the first point: 1
Enter the x-coordinate of the second point: 1
Enter the y-coordinate of the second point: 1
The distance between the two points is: 0
```

4.

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

class Triangle{
private:
  double totalAngle, angleA, angleB, angleC;
  double sideA, sideB, sideC; // Add side lengths

public:
  Triangle(double A, double B, double C, double a, double b, double c); // Constructor with side lengths
  void setAngles(double A, double B, double C);
  void setSides(double a, double b, double c); // Function to set side lengths
  const bool validateTriangle();
  double calculateArea(); // Function to calculate area
  double calculatePerimeter(); // Function to calculate perimeter
  string determineTriangleType(); // Function to determine triangle type
};

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

void Triangle::setAngles(double A, double B, double C) {
  angleA = A;
  angleB = B;
  angleC = C;
  totalAngle = A+B+C;
}

void Triangle::setSides(double a, double b, double c) {
  sideA = a;
  sideB = b;
  sideC = c;
}

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

double Triangle::calculateArea() {
  // Heron's formula
  double s = (sideA + sideB + sideC) / 2;
  return sqrt(s * (s - sideA) * (s - sideB) * (s - sideC));
}

double Triangle::calculatePerimeter() {
  return sideA + sideB + sideC;
}

string Triangle::determineTriangleType() {
  if (angleA < 90 && angleB < 90 && angleC < 90) {
    return "Acute-angled";
  } else if (angleA > 90 || angleB > 90 || angleC > 90) {
    return "Obtuse-angled";
  } else {
    return "Others (Right-angled or degenerate)";
  }
}

int main(){
  //driver code
  Triangle set1(40, 30, 110, 5, 6, 7); // Add side lengths to constructor
  if(set1.validateTriangle()){
    std::cout << "The shape is a valid triangle.\n";
    std::cout << "Area: " << set1.calculateArea() << endl;
    std::cout << "Perimeter: " << set1.calculatePerimeter() << endl;
    std::cout << "Type: " << set1.determineTriangleType() << endl;
  } else {
    std::cout << "The shape is NOT a valid triangle.\n";
  }
  return 0;
}
```

```
The shape is a valid triangle.
Area: 14.6969
Perimeter: 18
Type: Obtuse-angled
```

8. Conclusion

Today, I gained knowledge about the various components of C++ coding structures. The header file declaration section, the global declaration section, etc. are a few examples. These sections are required for the proper operation of a code. This new knowledge will help me through my course in computer engineering.

9. Assessment Rubric