Task 1

Write a class called **Marks** with a required constructor. Then perform the addition using operator overloading

#Write your code here

#Do not change the following lines of code

```
int main() {
    int q1, q2, lab, mid, final;
    cout << "Quiz 1 (out of 10): ";
    cin >> q1;
    cout << "Quiz 2 (out of 10): ";
    cin >> q2;
    cout << "Lab (out of 30): ";
    cin >> lab;
    cout << "Mid (out of 20): ";
    cin >> mid;
    cout << "Final (out of 30): ";
    cin >> final;
    Marks Q1(q1), Q2(q2), Lab(lab), Mid(mid), Final(final);
    Marks total = Q1 + Q2 + Lab + Mid + Final;
    cout << "Total marks: " << total.getMark() << endl;

    return 0;
}
```

Sample Input 1:

```
Quiz 1 (out of 10): 10
Quiz 2 (out of 10): 10
Lab (out of 30): 30
Mid (out of 20): 20
Final (out of 30): 30
```

Sample Output 1:

```
Total marks: 100
```

Sample Input 2:

```
Quiz 1 (out of 10): 10
Quiz 2 (out of 10): 8
Lab (out of 30): 30
Mid (out of 20): 20
Final (out of 30): 29
```

Sample Output 2:

```
Total marks: 97
```

Task 2

Write a class called Circle with the required constructor and methods to get the following output.

Subtasks:

1. Create a **class** called Circle.

2. Create the required **constructor**. Use **Encapsulation** to protect the variables. [**Hint:** Assign the variables in **private**]

3. Create **getRadius()** and **setRadius()** method to access variables.

4. Create a **method** called area to calculate the area of circles.

5. Handle the **operator overloading** by using a **special method** to calculate the radius and area of circle 3.

[You are not allowed to change the code below]

```
int main() {
   Circle c1(4);
   cout << "First circle radius: " <<
c1.getRadius() << endl;
   cout << "First circle area: " << c1.getArea()
<< endl;

   Circle c2(5);
   cout << "Second circle radius: " <<
c2.getRadius() << endl;
   cout << "Second circle area: " <<
c2.getArea() << endl;

   Circle c3 = c1 + c2;
   cout << "Third circle radius: " <<
c3.getRadius() << endl;
```

Output:

```
First circle radius: 4
First circle area: 50.26548245743669
Second circle radius: 5
Second circle area: 78.53981633974483
Third circle radius: 9
Third circle area: 254.46900494077323
```

```
   cout << "Third circle area: " << c3.getArea()
<< endl;


   return 0;
}
```

Task 3

Look at the code and the sample inputs and outputs below to design the program accordingly.

1. Write a class called **Color** that only adds the 3 primary colors (red, blue and yellow).
2. Write a required constructor for the class.
3. You have to use operator overloading to get the desired outputs as shown.

Hint:

There will be only one constructor and only one method tackling the addition operation. No other methods are required.
Note: Order of the color given as input should not matter. For example, in sample input 1, if the first input was yellow and then red, the output would still be orange.

#Write your code here

#Do not change the following lines of code

```
int main() {
   Color C1 = readColor();
   Color C2 = readColor();
   Color C3 = C1 + C2;
```

Sample Input 1:

```
First Color: red
Second Color: yellow
```

Sample Output 1:

```
Color formed: Orange
```

Sample Input 2:

```
First Color: red
Second Color: blue
```

Sample Output 2:

```
Color formed: Violet
```

cout << "Color formed: " << C3.getColor() << endl;
return 0;
}
	Sample Input 3:
	First Color: yellow
	Second Color: BLUE
	
	Sample Output 3:
	Color formed: Green

Task 4

Write a class called Triangle with the required constructor and methods to get the following output.

Subtasks:

1. Create a **class** called Triangle.

2. Create the required **constructor**. Use **Encapsulation** to protect the variables. [**Hint:** Assign the variables in **private**]

3. Create **getBase(), getHeight(), setBase()** and **setHeight()** methods to access variables.

4. Create a **method** called area to calculate the area of triangles.

5. Handle the **operator overloading** by using a **special method** to calculate the radius and area of triangle 3.

[You are not allowed to change the code below]

# Write your code here for subtasks 1-5

```
int main() {

Triangle t1(10, 5);

cout << "First Triangle Base: " << t1.getBase() << endl;

cout << "First Triangle Height: " << t1.getHeight() << endl;

cout << "First Triangle area: " << t1.calcArea() << endl;

Triangle t2(5, 3);

cout << "Second Triangle Base: " << t2.getBase() << endl;

cout << "Second Triangle Height: " << t2.getHeight() << endl;

cout << "Second Triangle area: " << t2.calcArea() << endl;

Triangle t3 = t1 - t2;

cout << "Third Triangle Base: " << t3.getBase() << endl;

cout << "Third Triangle Height: " << t3.getHeight() << endl;

cout << "Third Triangle area: " << t3.calcArea() << endl;

return 0;

}
```

Output:

```
First Triangle Base: 10
First Triangle Height: 5
First Triangle area: 25.0
Second Triangle Base: 5
Second Triangle Height: 3
Second Triangle area: 7.5
Third Triangle Base: 5
Third Triangle Height: 2
Third Triangle area: 5.0
```

Task 5

Design a class called **Coordinates** with an appropriate constructor. Then perform the subtraction, multiplication and equality check operations in the given code using operator overloading.

#Write your code here

#Do not change the following lines of code

```
int main() {
int x1, y1, x2, y2;
cin >> x1;
cin >> y1;

Coordinates p1(x1, y1);

cin >> x2;
cin >> y2;

Coordinates p2(x2, y2);

Coordinates p4 = p1 - p2;
cout << "Result of subtraction: (" << p4.getX()
<< ", " << p4.getY() << ")" << endl;

Coordinates p5 = p1 * p2;
cout << "Result of multiplication: (" << p5.getX()
<< ", " << p5.getY() << ")" << endl;

string point_check = p4.checkEqual(p5);
cout << point_check << endl;

return 0;
}
```

Sample Input 1:

```
1
2
3
4
```

Sample Output 1:

```
Result of subtraction is: (-2,-2)
Result of multiplication is: (3,8)
The calculated coordinates are NOT the same.
```

Sample Input 2:

```
0
0
0
0
```

Sample Output 2:

```
Result of subtraction is: (0,0)
Result of multiplication is: (0,0)
The calculated coordinates are the same.
```