Task 1

Create a "**Rectangle**" class with a private attribute "**length" and "width"**.

- Provide function to calculate **perimeter, diagonal length** and the **area** of the **Rectangle** object.
- Provide necessary public **getter** and **setter** function to assign values to the attribute. Note that setter function will check the validity of the variable (such as the "**length" and "width"** cannot be less than 0).
- Provide zero and two argument constructor.

Task 2

Add necessary member functions to the class **Rectangle** to execute the following codes. The description is given in the comment. Print in the console to check whether the functionality matches the description or not.

```
int main()
{
      Rectangle s1,s2(5,3); // length and width of s1 is 0,0 unit
      Rectangle s3 = ++s1; // after this both s1, s3 has 1,1 unit length and width
      Rectangle s4 = s1++; // after this s4 has 1,1 unit length and width
                           // & s1 has 2,2 unit length and width
      s3 = --s2; // after this both s2, s3 has 4,2 unit length and width
      s4 = s2--; // after this s4 has 4,2 unit length and width & s2 has 3,1 unit length
                 //and width
      s4 = s1+s2 // after this s4 has 5,3 unit length and width& s1 and s2 remains
unchanged

      // == returns true if area is on both side same else false
      if(s4 == s1+s2)
            cout<<"Area is same";
      // == returns true if LHS has bigger area else false
      else if(s4 > s1+s2)
            cout<<"LHS has bigger area";
      // == returns true if LHS has smaller area else false
      else if(s4 < s1+s2)
            cout<<"RHS has bigger area";

      return 0;
}
```

Task 3

Create a class called TAKA. It has an int private attribute one, two, five, ten, twenty. This attribute indicates how many notes of respective attributes are present in the object.

- Provide necessary getter and setter function. Remember that, none should be able to assign less than 0 to any of the attributes.
- Write getTaka() function to get the amount corresponding to the number of notes.
- Overload the == operator which will check whether the taka amount is the same or not, (the number of corresponding notes are not required to be the same.)
- Similarly overload > and < operator.
- Overload + operator which will return a Taka Object. The resulting object will contain the summation of corresponding notes.

Bonus:

- Overload - operator which will also return a Taka object. Plan a intuitive functionality of the - operator.