

Compile and run commands

```
g++ main.cpp Money.cpp -o hw3  
./hw3
```

## Report

### Money Class

- Money class has 2 attribute, lira and kurus. Attributes are in private mode there is no need to reach them without getter.
- Default constructor creates an object with lira = 0 and kurus =0.
- Constructor takes to integer and assigns them lira and kurus.
  - \*Constructor includes two **exceptions**.
    - 1.lira or kurus is less than 0.
    - 2.kurus is greater than 100.
- In case of these situations, an error message is thrown.
- Money + Money operator is overloaded to find sum of the array.
- Money + int operator is overloaded to add 5 TL.
- Money – int operator is overloaded to subtract 5TL.
- Money = int operator is overloaded to assign a money object to 0.
- Comparison operators are reloaded to compare an object with an integer.
- Getter functions of lira and kurus are needed so they are created.

### genericArray Class

- Whole class is under an **template**. T is the keyword for variables.
- genericArray class has 3 attributes. Since they are used in the example main function, elements and total attributes are in public mode. size is in private mode.
- the constructor takes an integer for size and sets size of the array.
- the class is also has sum function. Since different variable types needs different messages and typeid functions can be change by compiler to compiler. With assume of the variables can be int, double or Money, different sum functions are defined.
- Sum function of Money determines the bonus and applicate it. In case of the money is less than 5 TL is decreased to 0.
- Sum functions of double and integer only determines bonus and applicate it. Since these are points, negative points is not exceptions.

=== a main.cpp file which tests various of conditions included in homework file.

## Main.cpp :

```
#include <iostream>
#include "Money.h"
#include "genericArray.h"
using namespace std;
int main(int argc, const char * argv[]) {
    try{
        Money d(-3,5);//negative exception
    }catch(const char* result){
        cout<< result<<endl;
    }
    try{
        Money d(3,105);//kurus is greater than 99 exception
    }catch(const char* result){
        cout<< result<<endl;
    }
    Money a(50,0);
    Money b(50,0);

    Money c(51,0);
    Money d(50,0);

    Money e(49,0);
    Money f(50,0);

    genericArray<Money>mEqual(2);
    genericArray<Money>mGreater(2);
    genericArray<Money>mLess(2);
    genericArray<int>mInt(2);
    genericArray<double>mDouble(2);

    mEqual.elements[0]=a;
    mEqual.elements[1]=b;

    mGreater.elements[0]=c;
    mGreater.elements[1]=d;

    mLess.elements[0]=e;
    mLess.elements[1]=f;

    mInt.elements[0]=30;
    mInt.elements[1]=20;

    mDouble.elements[0]= 90.1;
    mDouble.elements[1]= 20.5;

    mEqual.total = mEqual.sum();
    mGreater.total = mGreater.sum();
    mLess.total = mLess.sum();
    mInt.total = mInt.sum();
    mDouble.total = mDouble.sum();

    return 0;
}
```

## Output :

```
The amount of Money can not be below zero
The kurus amount of Money can not be more than 99
Your total amount of money objects array 100 lira :0kurus
You still have the same money , your amount is 100 lira :0 kurus
Your total amount of money objects array 101 lira :0kurus
You have won 5 tl bonus, your new amount is 106 lira :0 kurus
Your total amount of money objects array 99 lira :0kurus
You have been punished 5 tl , your new amount is 94 lira :0 kurus
The total of integer array is 50
You have lost 5 points new total is 45
The total of double array is 110.6
You have won 5 points bonus new total is 115.6
Program ended with exit code: 0
```