CSE225L: Data Structures and Algorithm Lab

Lab 03: Class Template

North South University

Task 1. Modify the header file and the source file given below so that they now work as template class (the array elements in the dynamically allocated memory can be any type as the user defines).

dynarr.h

```
#ifndef DYNARR_H_INCLUDED
#define DYNARR_H_INCLUDED

using namespace std;
class dynArr
{
private:
    int* data,
    int size;
public:
    dynArr(int size);
    ~dynArr();
    void setValue(int index,int
value);
    int getValue(int index);
    int getSize();
};
#include "dynarr.tpp"
#endif // DYNARR_H_INCLUDED
```

dynarr.tpp

```
#include "dynarr.h"
dynArr::dynArr(int size){
        data = new int[size];
        this->size = size;
}
dynArr::~dynArr(){
        delete [] data;
}
void dynArr::setValue(int index, int value){
        data[index] = value;
}
int dynArr::getValue(int index){
        return data[index];
}
int dynArr::getSize(){
        return size;
}
```

Now take size as input and create a dynamic array of **char** data type. Take characters as input from user and populate the array. Finally print the whole array.

Task 2: Convert the following class to class template.

Complex.h

```
#ifndef COMPLEX_H_INCLUDED
#define COMPLEX_H_INCLUDED

using namespace std;
class Complex
{
public:
    Complex();
    Complex(int, int);
    Complex add(Complex);
    void Print();
private:
    int Real, Imaginary;
};

#include "Complex.tpp"
#endif // COMPLEX_H_INCLUDED
```

Complex.tpp

```
#include "Complex.h"
Complex::Complex()
{
    Real = 0;
    Imaginary = 0;
}
Complex::Complex(int r, int i)
{
    Real = r;
    Imaginary = i;
}
Complex Complex::add(Complex a)
{
    Complex t;
    t.Real = Real + a.Real;
    t.Imaginary = Imaginary + a.Imaginary;
    return t;
}
void Complex::Print()
{
    if (Real==0)
        cout << Imaginary<<"i"<<endl;
    else
    {
        if(Imaginary<0)
            cout<<Real<<Imaginary<<"i"<<endl;
        else if(Imaginary==0)
            cout<<Real<<endl;
        else
            cout<<Real<<"+"<<Imaginary<<"i"<<endl;
    }
}
```

In the driver (main.cpp) file, create two Complex objects with float as datatype and add them using add function. Finally print the result using Print function.