Question 1:

date.h

```
class date
{
	int day;
	int month;
	int year;
public:
	date();
	~date();
	date(int d, int m, int y);
	bool isvalid(int d,int m);
	void showdate();
	void getdate();

	date operator -(date d)
	{
		date temp;
		temp.day = day - d.day;
		temp.month = month - d.month;
		temp.year = year - d.year;

		temp.month = temp.month * 30;
		temp.year = temp.year * 365;
		return temp;
	}

	bool operator ==(date d)
	{
		if (d.day == day)
		{
			if (d.month == month)
			{
				if (d.year == year)
					return true;
				else
					return false;
			}
			else
				return false;
		}
		else
			return false;
	}

	bool operator <(date d)
	{
		if (year < d.year)
			return true;
		else if (year > d.year)
			return false;
```

```
	else if (year == d.year)
	{
		if (month < d.month)
			return true;
		else if (month > d.month)
			return false;
		else if (month == d.month)
		{
			if (day<d.day)
				return true;
			else
				return false;
		}
	}
}

bool operator <=(date d)
{
	if (year <= d.year)
	{
		if (month <= d.month)
		{
			if (day <= d.day)
				return true;
			else
				return false;
		}
		else
			return false;
	}
	else
		return false;
}

bool operator >(date d)
{
	if (year > d.year)
		return true;
	else if (year < d.year)
		return false;
	else if (year == d.year)
	{
		if (month > d.month)
			return true;
		else if (month < d.month)
			return false;
		else if (month == d.month)
		{
			if (day>d.day)
				return true;
			else
				return false;
		}
	}
```

```
	}


	void operator ++(int)
	{
		day++;
		if (day > 30)
		{
			month++;
			day = day - 30;
		}
		if (month > 12)
		{
			year++;
			month = month - 12;
		}
	}
	void operator --()
	{
		day--;
		if (day <= 0)
		{
			month--;
			day = day + 30;
		}
		if (month <=0)
		{
			year--;
			month = month + 12;
		}
	}

	int tellday()
	{
		return day;
	}
	int tellmonth()
	{
		return month;
	}
	int tellyear()
	{
		return year;
	}
};
```

date.cpp

```
#include "date.h"
#include<iostream>
using namespace std;

date::date()
```

```
{
}


date::~date()
{
}

date::date(int d, int m, int y)
{
	day = d;
	month = m;
	year = y;
}

bool date::isvalid(int d,int m)
{
	if (d >= 1 && d <= 31 || m >= 1 && m <= 12)
		return true;
	else
		return false;
}

void date::showdate()
{
	cout << endl;
	cout << " Date : " << day << "/" << month << "/" << year << endl;
}

void date::getdate()
{

	int d, m, y;
	cout << endl;
	cout << " Enter Day : ";
	cin >> d;
	cout << " Enter Month : ";
	cin >> m;
	cout << " Enter Year : ";
	cin >> y;

	if (isvalid(d, m))
	{
		day = d;
		month = m;
		year = y;
	}
	else
		cout << " Invalid values" << endl;
}
```

source.cpp

```
#include"date.h"
```

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

void main()
{
	date d1,d2,d3;
	int d,ch;
	d1.getdate();
	d2.getdate();
	d1.showdate();
	d2.showdate();
	do
	{
		cout << "\n\n Menu" << endl;
		cout << " 1. Comparison" << endl;
		cout << " 2. Number of days between two dates" << endl;
		cout << " 3. Increment date by a day" << endl;
		cout << " 4. Decrement date by a day" << endl;
		cout << " 5. Show all dates" << endl;
		cout << " 6. EXIT!" << endl << endl;
		cout << "\n Enter choice : ";
		cin >> ch;
		switch (ch)
		{
		case 1:
			if (d1 == d2)
				cout << " Both dates are equal" << endl;
			else if (d1 < d2)
				cout << " 1st date is smaller then 2nd date" <<
endl;
			else if (d1 > d2)
				cout << " 1st date is greater then 2nd date" <<
endl;
			if (d1 <= d2)
				cout << " 1st date is less then or equal to 2nd
date" << endl;
			break;
		case 2:
			if (d1 > d2)
				d3 = d1 - d2;
			else if (d1 < d2)
				d3 = d2 - d1;
			d = d3.tellday() + d3.tellmonth() + d3.tellyear();
			cout << "\n Total days : " << d << endl;
			break;
		case 3:
			d1++;
			d1.showdate();
			break;
		case 4:
			--d2;
			d2.showdate();
			break;
		case 5:
```

```
                d1.showdate();
                d2.showdate();
                break;
            }
        } while (ch != 6);
        system("pause");
}
```

Output

```
Enter Day : 1
Enter Month : 2
Enter Year : 3

Enter Day : 5
Enter Month : 6
Enter Year : 7

Date : 1/2/3

Date : 5/6/7


Menu
1. Comparison
2. Number of days between two dates
3. Increment date by a day
4. Decrement date by a day
5. Show all dates
6. EXIT!


Enter choice : 1
1st date is smaller then 2nd date
1st date is less then or equal to 2nd date
```

Question 2

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

template<class c>
double circlearea(c radius)
{
        double area = 3.14*radius*radius;
        return area;
}

void main()
{
        double area, area1;
        int radius;
        double radius1;
        cout << "Enter Radius in integer : ";
        cin >> radius;
```

```
	cout << "Enter Radius in float: ";
	cin >> radius1;
	area = circlearea(radius);
	area1 = circlearea(radius1);
	cout << "Area 1 = "<< area<<endl;
	cout << "Area 2 = "<< area1 << endl;
	system("pause");
}
```

Output

```
Enter Radius in integer : 12
Enter Radius in float: 12.5
Area 1 = 452.16
Area 2 = 490.625
Press any key to continue . . .
```

Question 3

Array.h

```
template<class a>
class Array
{
private:
	a arr[5];
public:
	void getdata()
	{
		for (int i = 0; i <= 4; i++)
		{
			cout << "Enter element # "<<i+1<<" : ";
			cin >> arr[i];
			cout << endl;
		}
	}
	void showlist()
	{
		for (int i = 0; i <= 4; i++)
		{
			cout << "element # " << i+1 << " : " <<arr[i] << endl;
		}
	}
	void showfirst()
	{
		cout << "First element : "<<arr[0]<<endl;
	}

	void sum()
```

```
	{
		a sum = 0;
		for (int i = 0; i <= 4; i++)
		{
			sum = sum + arr[i];
		}
		cout << "Total sum = " << sum<<endl;
	}
	void reverse()
	{
		a temp;
		int i=0;
		int j = 4;
		while (i < j)
		{
			temp = arr[i];
			arr[i] = arr[j];
			arr[j] = temp;
			i++;
			j--;
		}

	}
	void specifiedposition(int p)
	{
		int found = 0;
		for (int i = 0; i <= 4 && found==0; i++)
		{
			if (i == p)
			{
				cout << arr[p] << " is at position " << i << endl;
				found = 1;
			}
		}
		if (found == 0)
			cout << "Invalid position!" << endl;
	}
};
```

source.cpp

```
#include"Array.h"
#include<iostream>
using namespace std;
void main()
{
	Array <float>arr;
	arr.getdata();
	int ch,p;
	do
	{
		cout << "\n Menu" << endl;
		cout << " 1. Show list" << endl;
		cout << " 2. Show First" << endl;
```

```
            cout << " 3. Displays an element in a specified position " <<
endl;
            cout << " 4. Reverses the order of an array" << endl;
            cout << " 5. Sum the elements in an array " << endl;
            cout << " 6 EXIT!" << endl << endl;
            cout << "Enter choice : ";
            cin >> ch;
            if (ch == 1)
                  arr.showlist();
            else if (ch == 2)
                  arr.showfirst();
            else if (ch == 3)
            {
                  cout << "Enter the position : ";
                  cin >> p;
                  arr.specifiedposition(p);
            }
            else if (ch == 4)
                  arr.reverse();
            else if (ch == 5)
                  arr.sum();
      } while (ch != 6);
      system("pause");
}
```

Output

```
Enter element # 1 : 1

Enter element # 2 : 2

Enter element # 3 : 3

Enter element # 4 : 4

Enter element # 5 : 5


 Menu
 1. Show list
 2. Show First
 3. Displays an element in a specified position
 4. Reverses the order of an array
 5. Sum the elements in an array
 6 EXIT!

Enter choice : 1
element # 1 : 1
element # 2 : 2
element # 3 : 3
element # 4 : 4
element # 5 : 5

 Menu
 1. Show list
 2. Show First
 3. Displays an element in a specified position
```

Question 4

realestate.h

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

class realestate
{
	char location;
	float price;
	int rc;
public:
	class error
	{
	public:
		void error1()
		{
			cout << "Invalid" << endl;
		}
	};
	realestate();
	~realestate();

	realestate(char h, float p, int r)
	{
		if (h != 'S' && h != 'N' && h != 'W' && h != 'E')
			throw error();
		else
			location = h;

		if (p< 100000)
			throw error();
		else
			price = p;

		if (r <= 0)
			throw error();
		else
			rc = r;
	}
	void setlocation()
	{
		cout << "Enter location : ";
		cin >> location;
		if (location != 'S' && location != 'W' && location != 'E' &&
location != 'N')
			throw error();
	}
	void setprice()
	{
		cout << "Enter price : ";
		cin >> price;
		if (price < 100000)
```

```
                throw error();
        }
        void setrc()
        {
                cout << "Enter room count : ";
                cin >> rc;
                if (rc <= 0)
                        throw error();
        }
        void show()
        {
                cout << "Location : " << location << endl;
                cout << "Price : " << price << endl;
                cout << "Room count : " << rc << endl;
        }
};
```

source.cpp

```
#include"realestate.h"
void main()
{
        try
        {
                realestate r('N', 200000, 12);
                r.setlocation();
                r.setprice();
                r.setrc();
                r.show();
        }
        catch (realestate::error e)
        {
                e.error1();
        }
        system("pause");
}
```

Output

```
Enter location : N
Enter price : 2000000
Enter room count : 12
Location : N
Price : 2e+006
Room count : 12
Press any key to continue . . .
```

Question 5

lcd.h

```
#include<iostream>
#include<string>
using namespace std;
class lcd
{
	string num;
	int size;
	float price;
public:
	class anexcep
	{
	public:
		void what()
		{
			cout << "Error" << endl;
		}
	};
	void setmodnum()
	{
		cout << "\nEnter Model number : ";
		cin >> num;
		if (num[0] != 'v')
			throw anexcep();
	}
	void setsize()
	{
		cout << "Enter screen size : ";
		cin >> size;
		if (size < 30 || size>55)
			throw anexcep();
	}
	void setprice()
	{
		cout << "Enter item price : ";
		cin >> price;
		if (price < 0 || price>90000)
			throw anexcep();
	}
	string getnum()
	{
		return num;
	}
	int getsize()
	{
		return size;
	}
	float getprice()
	{
		return price;
	}
	lcd();
	~lcd();
};
```

Lcd.cpp

```
#include "lcd.h"


lcd::lcd()
{
	num = "";
	size = 0;
	price = 0;
}


lcd::~lcd()
{
}
```

Source.cpp

```
#include"lcd.h"
void main()
{
	lcd l1, l2, l3;
	try
	{
		l1.setmodnum();
		l1.setprice();
		l1.setsize();
		cout << "\n MOdel number : " << l1.getnum() << endl;
		cout << " Price : " << l1.getprice() << endl;
		cout << " Size : " << l1.getprice() << endl;
		l2.setmodnum();
		l2.setprice();
		l2.setsize();
		cout << "\n MOdel number : " << l2.getnum() << endl;
		cout << " Price : " << l2.getprice() << endl;
		cout << " Size : " << l2.getprice() << endl;
		l3.setmodnum();
		l3.setprice();
		l3.setsize();
		cout << "\n MOdel number : " << l3.getnum() << endl;
		cout << " Price : " << l3.getprice() << endl;
		cout << " Size : " << l3.getprice() << endl;
	}
	catch (lcd::anexcep e)
	{
		e.what();
	}
	system("pause");
}
```

Output

```
Enter Model number : vff5
Enter item price : 10000
Enter screen size : 38

 MOdel number : vff5
 Price : 10000
 Size : 10000

Enter Model number : 5
Error
Press any key to continue . . .
```
