Program 1

```
#include<iostream>
using namespace std;
class elebill
{
	char name[20];
	float unit,price,surcharge;
	public:
		void input();
		void calc();
		void display();
};
void elebill::input()
{
	cout<<"Enter name"<<endl;
	cin>>name;
	cout<<"Enter the unit"<<endl;
	cin>>unit;
}
void elebill::display()
{
	cout<<"Username="<<name<<endl;
	cout<<"Units consumed="<<unit<<endl;
	cout<<"Amount="<<price<<endl;
	cout<<"Surcharge="<<surcharge<<endl;
}
void elebill::calc()
{
	if(unit<=100)
	{
		price=(unit*1.5)+50;
	}
	else if(unit<=200)
	{
		price=50+(100*1.5)+(unit-100)*1.80;
	}
	else
	{
		price=50+(100*1.5)+(100*1.8)+(unit-200)*2.5;
	}
	if(price>300)
	{
		surcharge=(price*0.15);
		price=(price+surcharge);
	}
}
int main()
{
	elebill user1;
	user1.input();
	user1.calc();
	user1.display();
	return 0;
}
```

PROGRAM2

```
#include<iostream>
using namespace std;
class student
{
	static int count;
	char name[30];
	int reg;
	float t1,t2,t3;
	float avg;
	public:
		void readdata()
		{
			cout<<"Enter name"<<endl;
			cin>>name;
			cout<<"Enter the marks"<<endl;
			cin>>t1>>t2>>t3;
			reg=++count;
		}
		void display()
		{
			cout<<"Student details"<<endl;
			cout<<"Name:"<<name<<endl;
			cout<<"Reg no:"<<reg<<endl;
			cout<<"test 1:"<<t1<<endl;
			cout<<"test 2:"<<t2<<endl;
			cout<<"test 3:"<<t3<<endl;
			cout<<"Average:"<<avg<<endl;
		}
		void compute()
		{
			if(t1<=t2 && t1<=t3)
			{
				avg=(t2+t3)/2;
			}
			else if(t2<=t1 && t2<=t3)
			{
				avg=(t1+t3)/2;
			}
			else
			{
				avg=(t1+t2)/2;
			}
		}
};
```

```
int  student::count;
int main()
{
	student s[30];
	int n,i;
	cout<<"Enter number of students"<<endl;
	cin>>n;
	for(i=0;i<n;i++)
		s[i].readdata();
		s[i].compute();
	for(i=0;i<n;i++)
		s[i].display();
	return 0;
}
```

Program3

```
#include<iostream>
using namespace std;
#include<iomanip>
class complex
{
	float real,image;
	public:
		void input(float r,float i)
		{
			real=r;
			image=i;
		}
		complex add(int);
		complex add(complex);
		void show(complex);
};
complex complex::add(int a)
{
	complex s2;
	s2.real=real+a;
	s2.image=image;
	return s2;
}
complex complex::add(complex s2)
{
	complex s3;
	s3.real=real+s2.real;
	s3.image=image+s2.image;
	return s3;
}
void complex::show(complex c)
{
	cout<<"\n"<<c.real<<"+i"<<c.image<<endl;
}
int main()
{
	complex s1,s2,s3;
	s1.input(1,2);
	s2=s1.add(s1);
	s3=s2.add(s2);
	s1.show(s1);
```

```
	s2.show(s2);
	s3.show(s3);
	return 0;
}
```

Program4

```
#include<iostream>
#include<iomanip>
#include<string.h>
using namespace std;
class STRING
{
	char name[20];
	public: STRING()
		{
			name[20]='\0';
		}
		STRING(char a[20])
		{
			strcpy(name,a);
		}
		STRING(STRING &s)
		{
			strcpy(name,s.name);
		}
		friend STRING operator+(STRING S1,STRING s2);
		friend ostream &operator<<(ostream &p,STRING s);
};
STRING operator+(STRING s1,STRING s2)
{
	STRING temp(s1);
	strcat(temp.name," ");
	strcat(temp.name,s2.name);
	return temp;
}
ostream &operator<<(ostream &p,STRING s)
{
	p<<s.name<<endl;
	return p;
}
int main()
{
	STRING s3;
	STRING s1("Dr AIT");
	STRING s2("Bangalore");
	cout<<s1;
	cout<<s2;
	s3=s1+s2;
	cout<<s3;
}
```

Program5

```
#include<iostream>
#include<string.h>
using namespace std;
struct node
{
	char data[20];
	struct node *left,*right;
};
typedef struct node *Node;
class dlist
{
	private:Node first;
		int count;
	public: dlist()
		{
			first=NULL;
			count=1;
		}
		void create();
		void del(int pos);
		void print();
};
void dlist::create()
{
	char item[10];
	cout<<"enter the data\n";
	cin>>item;
	Node temp;
	temp=new(struct node);
	temp->left=temp->right=NULL;
	if(first==NULL)
	{
		strcpy(temp->data,item);
		first=temp;
	}
	else
	{
		strcpy(temp->data,item);
		temp->right=first;
		first->left=temp;
		first=temp;
	}
}
void dlist::del(int pos)
{
	Node cur,prev,temp;
	if(first==NULL)
	{
		cout<<"no element to delete"<<endl;
		return;
	}
	if(pos==1)
	{
		cout<<first->data;
		cur=first;
		first=first->right;
```

```
                first->left=NULL;
                delete cur;
                return;
        }
        prev=NULL;
        cur=first;
        while(cur!=NULL && count!=pos)
        {
                prev=cur;
                cur=cur->right;
                count++;
        }
        count=1;
        temp=cur->right;
        prev->right=temp;
        temp->left=prev;
        cout<<"the item deleted is :"<<cur->data<<endl;
        delete(cur);
}
void dlist::print()
{
        Node temp;
        if(first==NULL)
        {
                cout<<"list is empty"<<endl;
        }
        temp=first;
        while(temp!=NULL)
        {
                cout<<"->"<<temp->data;
                temp=temp->right;
        }
}
int  main()
{
        int pos,choice;
        char  item[10];
        dlist d;
        while(1)
        {
                cout<<endl<<"1:insert 2:delete 3:print 4:exit"<<endl;
                cin>>choice;
                switch(choice)
                {
                        case 1: d.create();
                                break;
                        case 2: cout<<"enter the position"<<endl;
                                cin>>pos;
                                d.del(pos);
                                break;
                        case 3 :cout<<"contents of the list are:"<<endl;
                                d.print();
                                break;
                        default:return(0);
                }
        }
}
```

Program6

```
#include<iostream>
#include<iomanip>
using namespace std;
template<class T>
class queue
{
	T a[10];
	int r,f,SIZE;
	public:queue(int n)
		{
			r=-1;
			f=0;
			SIZE=n;
		}
		void insert_rear();
		void delete_front();
		void display();
};

template<class T>
void queue<T>::insert_rear()
{
	T item;
	if(r==SIZE-1)
	{
		cout<<"queue is full"<<endl;
		return;
	}
	else
	{
		cout<<"enter the item to be inserted"<<endl;
		cin>>item;
		a[++r]=item;
		return;
	}
}

template<class T>
void queue<T>::delete_front()
{
	if(f>r)
	{
		cout<<"queue Underflow \n"<<endl;
		return;
	}
	cout<<"the deleted item ="<<a[f++]<<endl;
	return;
}




template<class T>
void queue<T>::display()
{
```

```
        if(f>r)
        {
                cout<<"queue is empty\n"<<endl;
                return;
        }
        cout<<"the elements of the queue are :: \n"<<endl;
        for(int i=f;i<SIZE;i++)
        cout<<a[i]<<endl;
        return;
}

int main()
{
        int n;
        cout<<"enter the size of an array"<<endl;
        cin>>n;
        queue<int>Q1(n);
        queue<double>Q2(n);
        int ch,c;
        cout<<"enter your choice"<<endl;
        cout<<"\n1:for intergers \n2:for Double \n3:Exit"<<endl;
        cin>>ch;
        switch(ch)
        {
                case 1:while(1)
         {
               cout<<"1:Insert 2:delete 3:display 4:exit \n"<<endl;
               cin>>c;
               switch(c)
               {
                  case 1:Q1.insert_rear();break;
                  case 2:Q1.delete_front();break;
                  case 3:Q1.display();break;
                  default:return 0;
               }
         }

                case 2:while(1)
                {
                        cout<<"1:Insert 2:delete 3:display 4:exit \n"<<endl;
                        cin>>c;
                        switch(c)
                        {
                                case 1:Q2.insert_rear();break;
                                case 2:Q2.delete_front();break;
                                case 3:Q2.display();break;
                                default:return 0;
                        }
                }
                default:return 0;//exit(0);
        }
}
```

Program7

```
#include<iostream>
#include<iomanip>
using namespace std;
class date
{
        int d,m,y;
public: int leap(int);
        int days(int);
        void getdate(void);
        int operator-(date);
        date operator+(int);
        friend ostream &operator<<(ostream &,date);
        friend int checkdate(date,date);
};
ostream &operator<<(ostream &out,date d)
{
        out<<d.d<<"/"<<d.m<<"/"<<d.y;
        return(out);
}
int date::leap(int y)
{
        if(y%4==0)
        return(1);
        else
        return(0);
}
int date::days(int i)
{
        if(i==2)
        return(28);
        if(i==1 || i==3 || i==5 || i==7 || i==8 || i==10 || i==12)
        return(31);
        return(30);
}
int date::operator-(date d2)
{
        int leapyear,count=0;
        while(m!=d2.m || y!=d2.y)
        {
                d2.d++;
                count++;
                leapyear=0;
                if(d2.m==2)
                leapyear=d2.leap(d2.y);
                if(d2.d>(days(d2.m)+leapyear))
                {
                        d2.d=1;
                        d2.m++;
                }
                if(d2.m>12)
                {
                        d2.m=1;
                        d2.y++;
                }
        }
        count+=(d-d2.d);
```

```
	return(count);
}
date date::operator+(int ndays)
{
	int i,leapyear=0;
	for(i=1;i<=ndays;i++)
	{
		d++;
		leapyear=0;
		if(m==2)
		leapyear=leap(y);
		if(d>(days(m)+leapyear))
		{
			m++;
			d=1;
		}
		if(m>12)
		{
			m=1;
			y++;
		}
	}
	return (*this);
}
void date::getdate(void)
{
	cout<<"\n Enter the date\n";
	cout<<"Day(dd):\n";
	cin>>d;
	cout<<"month(mm):\n";
	cin>>m;
	cout<<"year (yy):\n";
	cin>>y;
}
int checkdate(date d1,date d2)
{
	if(d1.y<d2.y)
	return(0);
	else if(d1.y==d2.y)
	{
		if(d1.m<d2.m)
		return(0);
		else
		if(d1.m==d2.m)
		{
			if(d1.d<d2.d)
				return(0);
			else
				return(1);
		}
		else return (1);
	}
	else return(1);
}
int main()
{
	date d1,d2,d3,d4;
```

```
	int choice,no_days=0,i;
	for(;;)
	{
		cout<<"1:substract 2 dates\n2:add no. of days to a date\n";
		cin>>choice;
		switch(choice)
		{
			case 1: cout<<"Enter date 1 greater then date 2"<<endl;
				d1.getdate();
				d2.getdate();
				i=checkdate(d1,d2);
				if(i==1)
				{
					no_days=d1-d2;
					cout<<endl<<d1<<"-"<<d2<<endl;
				}
				else
				{
					no_days=d2-d1;
					cout<<endl<<d2<<"-"<<d1<<endl;
				}
				break;
			case 2: d3.getdate();
				cout<<endl<<"Enter no of days to be added\n";
				cin>>no_days;
				cout<<endl<<d3<<"+"<<no_days<<endl;
				d4=d3+no_days;
				cout<<endl<<d4<<endl;
				break;
			default:cout<<"Thnx";
				break;
		}
		break;
	}
	return 0;
}

Program8
#include<iostream>
using namespace std;
class num
{
	public: char a[10];
		int o;
};
class hexa:public num
{
	public:int val;
		hexa()
		{
			val=0;
		}
		void read()
		{
			cout<<"Enter hexa no"<<endl;
			cin>>a;
		}
```

```
		void hextodec()
		{
			int t,i,j=1,n=0;
			for(i=0;a[i]!='\0';i++)
			{
				n++;
			}
			for(i=n-1;i>=0;i--)
			{
				if(a[i]>'9')
				{
					t=a[i]-87;
					cout<<"t="<<t<<endl;
					val=val+(t*j);
					cout<<"Val="<<val<<endl;
				}
				else
				{
					t=a[i]-48;
					val=val+(t*j);
				}
				j=j*16;
			}
			cout<<"The hexdecimal to decimal is "<<val<<endl;
		}
};
class octal:public num
{
	public: int val1;
		octal()
		{
			val1=0;
		}
		void read()
		{
			cout<<"Enter octal no"<<endl;
			cin>>o;
		}
		void octaltodec()
		{
			int j1=1,d;
			while(o!=0)
			{
				d=o%10;
				val1=val1+(d+j1);
				o=o/10;
				j1=j1*8;
			}
			cout<<"The octal to decimal is"<<val1<<endl;
		}
		friend int operator +(hexa n1,octal n2);
};
int operator +(hexa n1,octal n2)
{
	int k=n1.val+n2.val1;
	return k;
}
```

```
int operator +(octal n1,int n2)
{
	int x=n1.val1+n2;
	return x;
}
int main()
{
	hexa h;
	h.read();
	h.hextodec();
	octal o1,o2;
	o1.read();
	o1.octaltodec();
	int c=h+o1;
	cout<<"The value of J= "<<c<<endl;
	o2.read();
	o2.octaltodec();
	cout<<"Enter a integer number "<<endl;
	int no;
	cin>>no;
	int y=o2+no;
	cout<<"The value of y="<<y;
	return 0;
}
```

Program9

```
#include<iostream>
#include<iomanip>
using namespace std;
class shape
{
	public: int x,y;
		virtual void read()=0;
		virtual void compute()=0;
		virtual void display()=0;
};
class rectangle:public shape
{
	float area;
	public:void read()
		{
			cout<<"Enter x and y"<<endl;
			cin>>x>>y;
		}
		void compute()
		{
			area=x*y;
		}
		void display()
		{
			cout<<"Area rectangle"<<area<<endl;
		}
};
class circle:public shape
{
	float area;
	public:void read()
```

```
                {
                        cout<<"Enter radius"<<endl;
                        cin>>x;
                }
                void compute()
                {
                        area=3.142*x*x;
                }
                void display()
                {
                        cout<<"Area of circle:"<<area<<endl;
                }
};
class square:public shape
{
        float area;
        public:void read()
                {
                        cout<<"Enter the side"<<endl;
                        cin>>x;
                }
                void compute()
                {
                        area=x*x;
                }
                void display()
                {
                        cout<<"Area square="<<area<<endl;
                }
};
int main()
{
    shape *s[3];
    circle c;
    square sq;
    rectangle r;
    s[0]=&c;
    s[1]=&r;
    s[2]=&sq;
    while(1)
    {
        cout<<"enter the choice"<<endl;
        cout<<"1:area of circle 2:area of rectangle 3:area of square"<<endl;
        int ch;
        cin>>ch;
        switch(ch)
        {
            case 1: s[0]->read();
                    s[0]->compute();
                    c.display();
                    break;
            case 2: s[1]->read();
                    s[1]->compute();
                    r.display();
                    break;
            case 3: s[2]->read();
                    s[2]->compute();
```

```
                    sq.display();
                    break;
                default:return 0;
            }
        }
}

Program10
#include<iostream>
#include<string.h>
using namespace std;
void fact(int);
class ide
{
        public:char str_what[80];
        int what;
        ide()
        {
                *str_what=0;
                what=0;
        }
        ide(char *s,int e)
        {
                strcpy(str_what,s);
                what=e;
        }
};
int main()
{
        int n;
        try
        {
                cout<<"Enter no to compute n fact"<<endl;
                cin>>n;
                if(n<0)
                throw ide("negative number entered",n);
                else
                fact(n);
        }
        catch(ide e)
        {
                cout<<e.str_what<<":";
                cout<<e.what<<endl;
        }
        return 0;
}
void fact(int n)
{
        int fact=1,i;
        for(i=1;i<=n;i++)
        {
                fact=fact*i;
        }
        cout<<"The factorial is :"<<fact<<endl;
}
```

Program 11

```
#include<iostream>
#include<list>
#include<vector>
using namespace std;
void display(vector <int>&v2)
{
	cout<<"The contents of v2 is "<<endl;
	for(int i=0;i<v2.size();i++)
	{
		cout<<v2[i]<<endl;
	}
}
void displaylist(list <int>&l)
{
	list <int>::iterator p;
	cout<<"The content of list are"<<endl;
	for(p=l.begin();p!=l.end();p++)
	{
		cout<<*p<<endl;
	}
}
int main()
{
	vector <int>v1;
	int x,i;
	cout<<"Enter 5 unsorted elements on the vector 1"<<endl;
	for(i=0;i<5;i++)
	{
		cin>>x;
		v1.push_back(x);
	}
	list <int>l;
	cout<<"Enter n sorted form vector 1 to lst"<<endl;
	copy(v1.begin(),v1.end(),back_inserter(l));
	l.sort();
	displaylist(l);
	vector <int>v2;
	copy(l.begin(),l.end(),back_inserter(v2));
	display(v2);
	return 0;
}
```

Outputs:

program1:

```
Enter name
apple
Enter the unit
276
Username=apple
Units consumed=276
Amount=655.5
Surcharge=85.5
```

program2:
Enter number of students
1
Enter name
12
Enter the marks
20
19
18
Student details
Name:12
Reg no:1
test 1:20
test 2:19
test 3:18
Average:19.5

Program3

1+i2

2+i4

4+i8

Program4
Dr AIT
Bangalore
Dr AIT Bangalore

Program5
1:insert 2:delete 3:print 4:exit
1
enter the data
2

1:insert 2:delete 3:print 4:exit
1
enter the data
3

1:insert 2:delete 3:print 4:exit
3
contents of the list are:
->3->2
1:insert 2:delete 3:print 4:exit
2
enter the position
1
3
1:insert 2:delete 3:print 4:exit
3
contents of the list are:
->2
1:insert 2:delete 3:print 4:exit
4

Program6
(1)
enter the size of an array
2
enter your choice

1:for intergers
2:for Double
3:Exit
1
1:Insert 2:delete 3:display 4:exit

1
enter the item to be inserted
3
1:Insert 2:delete 3:display 4:exit

1
enter the item to be inserted
4
1:Insert 2:delete 3:display 4:exit

3
the elements of the queue are ::

3
4
1:Insert 2:delete 3:display 4:exit

2
the deleted item =3
1:Insert 2:delete 3:display 4:exit

3
the elements of the queue are ::

4
1:Insert 2:delete 3:display 4:exit

4

(2)

enter the size of an array
2
enter your choice

1:for intergers
2:for Double
3:Exit
2
1:Insert 2:delete 3:display 4:exit

1
enter the item to be inserted
14.5
1:Insert 2:delete 3:display 4:exit

1
enter the item to be inserted
15.7
1:Insert 2:delete 3:display 4:exit

3
the elements of the queue are ::

14.5
15.7
1:Insert 2:delete 3:display 4:exit

2
the deleted item =14.5
1:Insert 2:delete 3:display 4:exit

3
the elements of the queue are ::

15.7
1:Insert 2:delete 3:display 4:exit

4

Program7
1:substract 2 dates
2:add no. of days to a date
2

Enter the date
Day(dd):
19
month(mm):
12
year (yy):
1997

Enter no of days to be added
34

19/12/1997+34

22/1/1998

2) 1:substract 2 dates
2:add no. of days to a date
1
Enter date 1 greater then date 2

Enter the date
Day(dd):
11
month(mm):
1
year (yy):
1997

```
 Enter the date
Day(dd):
11
month(mm):
5
year (yy):
1996

11/1/1997-11/5/1996

Program8
Enter hexa no
64
The hexdecimal to decimal is 100
Enter octal no
800
The octal to decimal is81
The value of J= 181
Enter octal no
99
The octal to decimal is27
Enter a integer number
56
The value of y=83

Program9
enter the choice
1:area of circle 2:area of rectangle 3:area of square
1
Enter radius
6
Area of circle:113.112
enter the choice
1:area of circle 2:area of rectangle 3:area of square
2
Enter x and y
17 9
Area rectangle153
enter the choice
1:area of circle 2:area of rectangle 3:area of square
3
Enter the side
4
Area square=16
enter the choice
1:area of circle 2:area of rectangle 3:area of square
4

Program10
Enter no to compute n fact
6
The factorial is :720

Enter no to compute n fact
-15
negative number entered:-15
```

program11
Enter 5 unsorted elements on the vector 1
89
54
78
22
5
Enter n sorted form vector 1 to lst
The content of list are
5
22
54
78
89
The contents of v2 is
5
22
54
78
89