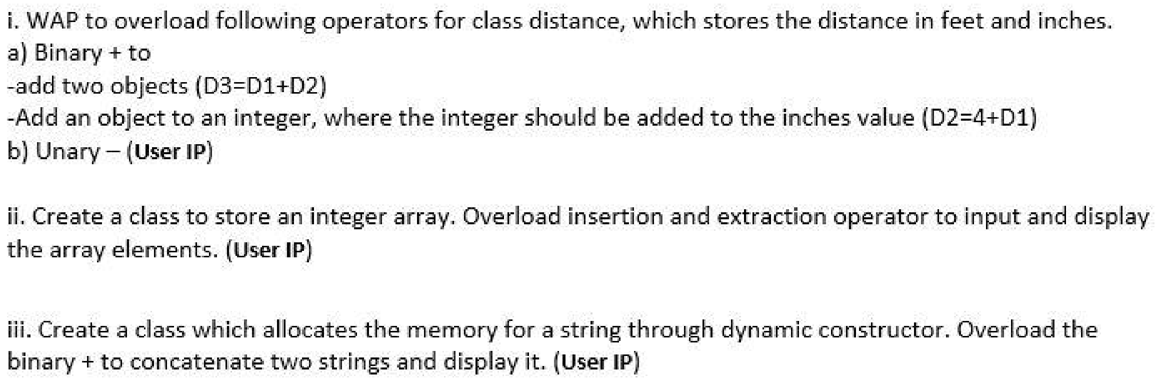
Name:Auro Saswat Raj
Roll:22057020

8. Operator Overloading



i. WAP to overload following operators for class distance, which stores the distance in feet and inches.
a) Binary + to
-add two objects (D3=D1+D2)
-Add an object to an integer, where the integer should be added to the inches value (D2=4+D1)
b) Unary – (**User IP**)

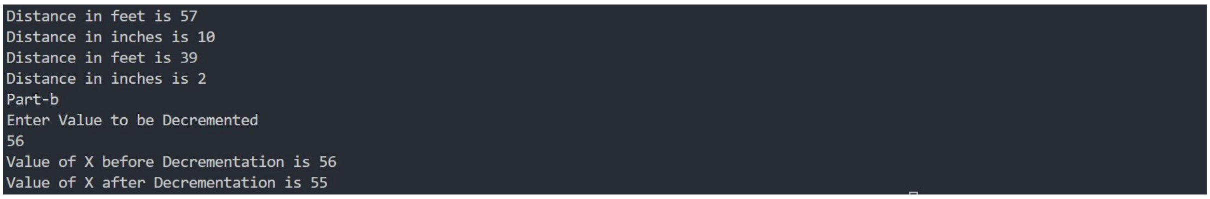
ii. Create a class to store an integer array. Overload insertion and extraction operator to input and display the array elements. (**User IP**)

iii. Create a class which allocates the memory for a string through dynamic constructor. Overload the binary + to concatenate two strings and display it. (**User IP**)

Program-1

```
#include<iostream>

using namespace std;

class quest1

{

private:

  int feet,inches;

public:

  quest1(){

  feet=0;

  inches=0;

  }

  quest1(int feet,int inches){

    this->feet=feet;

    this->inches=inches;

  }


  quest1 operator+(quest1 const &q1){

    quest1 q2;

   q2.feet=feet+q1.feet;
   q2.inches=inches+q1.inches;
   while(q2.inches >= 12.0) {
   q2.inches = q2.inches - 12.0;
   ++q2.feet;
  }
  return q2;
  }
  friend quest1 operator+(int z,quest1 const&q1){
    quest1 q2;
    q2.feet=z+q1.feet;
    q2.inches=0+q1.inches;
    return q2;
  }
  void show(){
    cout<<"Distance in feet is "<<this->feet<<endl;
    cout<<"Distance in inches is "<<this->inches<<endl;
  }
};

class quest1b
{
private:
  int x;
public:
  quest1b(){
    cout<<"Enter Value to be Decremented"<<endl;
    cin>>x;
    cout<<"Value of X before Decrementation is "<<x<<endl;
  }
  quest1b operator--(){
    --x;
```

```
        return *this;
    }
    void show(){
        cout<<"Value of X after Decrementation is "<<x<<endl;
    }
};

int main()
{
    quest1 q1(23,8.6),q2(34,2.4);
    quest1 q3;
    q3=q1+q2;
    q3.show();
    quest1 q4;
    q4=5+q2;
    q4.show();
    cout<<"Part-b"<<endl;
    quest1b q1b;
    --q1b;
    q1b.show();
    return 0;
}
```

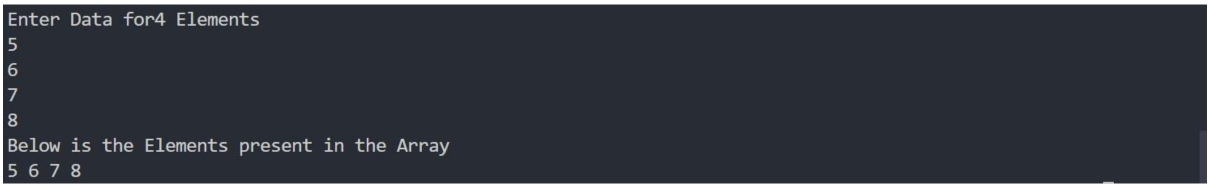


```
Distance in feet is 57
Distance in inches is 10
Distance in feet is 39
Distance in inches is 2
Part-b
Enter Value to be Decremented
56
Value of X before Decrementation is 56
Value of X after Decrementation is 55
```

Program-2

```
#include<iostream>

using namespace std;

class quest2

{

private:

   int n;

   int *arr;

public:

   quest2(int n);

   friend istream & operator >>(istream &CIN,quest2 &q2){

      cout<<"Enter Data for"<<q2.n<<" Elements"<<endl;

      for (int i = 0; i < q2.n; i++)

      {

         CIN>>q2.arr[i];

      }

   }

   friend ostream & operator<<(ostream &COUT,quest2 &q2){

      cout<<"Below is the Elements present in the Array"<<endl;

      for (int i = 0; i < q2.n; i++)

      {

         COUT<<q2.arr[i]<<" ";

      }

   }

};

quest2::quest2(int n)

{

   this->n=n;

   arr=new int[n];

}
```

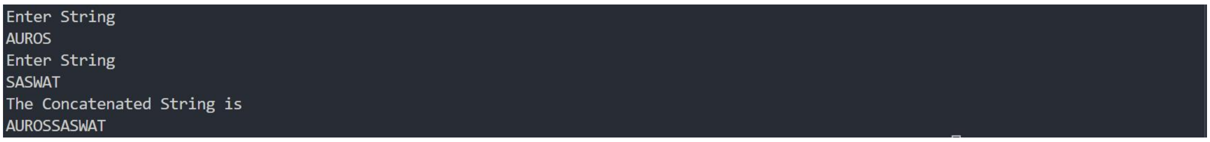
```
int main()

{

    quest2 q2(4);

    cin>>q2;

    cout<<q2;

    return 0;

}
```



```
Enter Data for4 Elements
5
6
7
8
Below is the Elements present in the Array
5 6 7 8
```

Program-3

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

class quest3
{
private:
    int l;
    char *c;
public:

    quest3(){
        l=100;
        c=new char[l];
    }
    quest3(int l){
        this->l=l;
        c=new char[this->l];
        cout<<"Enter String"<<endl;
        cin>>c;
    }
    quest3 operator+(quest3 &q3){
        quest3 q2;
        q2.l=l+q3.l;
        q2.c=new char[q2.l];
        int i;
        for (i = 0; i < l; i++)
        {
            q2.c[i]=this->c[i];
        }
        for (int j = 0; j < q3.l; j++)
        {
            q2.c[i++]=q3.c[j];
        }
        return q2;
    }

    void show(){
        cout<<"The Concatenated String is "<<endl;
        for (int i = 0; i < l; i++)
        {
            cout<<c[i];
        }
    }
};

int main()
{
    quest3 q1,q3(5),q2(6);
    q1=q3+q2;
    q1.show();

    return 0;
}
```



```
Enter String
AUROS
Enter String
SASWAT
The Concatenated String is
AUROSSASWAT
```