

Logic Building Assignment : 51

Complete below code snippet by writing definitions of below function

1. SearchLast() - Search last occurrence of number
2. EvenCount() - Count even elements
3. OddCount(). - Count odd elements
4. SumAll(). - sum of all elements

```
#include<iostream>
```

```
using namespace std;
```

```
class Array
```

```
{
```

```
protected:
```

```
    int *Arr;
```

```
    int size;
```

```
public:
```

```
    Array(int value = 10)
```

```
    {
```

```
        cout<<"Inside Connstructor\n";
```

```
        this->size = value;
```

```
        this->Arr = new int[size];
```

```
    }
```

```
    Array(Array &ref)
```

```
    {
```

```
        cout<<"Inside copy connstructor\n";
```

```
        this->size = ref.size;
```

```
        this->Arr = new int[this->size];
```

```
        for(int i = 0;i<size;i++)
```

```
        {
```

```
            this->Arr[i] = ref.Arr[i];
```

```
        }
```

```
    }
```

```
    ~Array()
```

```
    {
```

```
        cout<<"Inside Destructor\n";
```

```
        delete []Arr;
```

```
    }
```

```
        inline void Accept();
        inline void Display();
};

void Array::Accept()
{
    cout<<"Please enter the values\n";

    for(int i = 0;i<this->size;i++)
    {
        cin>>Arr[i];
    }
}

void Array::Display()
{
    cout<<"Elements are\n";

    for(int i = 0;i<this->size;i++)
    {
        cout<<Arr[i]<<" ";
    }
    cout<<"\n";
}

class ArrSearch : public Array
{
public:
    ArrSearch(int no = 10) : Array(no)
    {}

    int Frequency(int);
    int SearchFirst(int);
    int SearchLast(int);
    int EvenCount();
    int OddCount();
};

int ArrSearch::SearchFirst(int value)
{
    int i = 0;

    for(i = 0; i < size; i++)
    {
        if(Arr[i] == value)
        {
            break;
        }
    }
}
```

```
    }  
}  
  
if(i == size)  
{  
    return -1;  
}  
else  
{  
    return i + 1;  
}  
}  
int ArrSearch::Frequency(int value)  
{  
    int icnt = 0;  
  
    for(int i = 0; i<size; i++)  
    {  
        if(Arr[i] == value)  
        {  
            icnt++;  
        }  
    }  
  
    return icnt;  
}  
  
int ArrSearch::SearchLast(int value)  
{  
    // Logic  
}  
  
int ArrSearch::EvenCount()  
{  
    // Logic  
}  
  
int ArrSearch::OddCount()  
{  
    // Logic  
}  
  
int ArrSearch::SumAll()  
{  
    // Logic  
}
```

```
int main()
{
    cout<<"Inside main\n";

    ArrSearch sobj1(5);
    sobj1.Accept();
    sobj1.Display();

    int iret = sobj1.Frequency(11);

    cout<<"Frequency is "<<iret<<"\n";

    iret = sobj1.SearchFirst(11);

    cout<<"First occurrence is "<<iret<<"\n";

    // Call all the above functions

    return 0;
}
```

