Answer to the question no-01

```
#include<bits/stdc++.h>
using namespace std;
vector<int>quick_sort(vector<int>A)
{
   if(A.size()<=1){return A;}
   int P=rand()%(A.size());
   vector<int>B,C;
   for(int i=0;i<A.size();i++)
   {
      if(i==P){continue;}
      if(A[i]<=A[P]){B.push_back(A[i]);}
      else{C.push_back(A[i]);}
   }
   vector<int>Sort_B=quick_sort(B);
   vector<int>Sort_C=quick_sort(C);
   vector<int>Sort_A;
   for(int i=0;i<Sort_C.size();i++)
   {
      Sort_A.push_back(Sort_C[i]);
   }
   Sort_A.push_back(A[P]);
   for(int i=0;i<Sort_B.size();i++)
   {
      Sort_A.push_back(Sort_B[i]);
   }
   return Sort_A;
}

int main()
{
  int n;
  cin>>n;
  vector<int>a(n);
  for(int i=0;i<n;i++)
  {
     cin>>a[i];
```

```
  }
  vector<int>Sort_A=quick_sort(a);
  for(int i=0;i<Sort_A.size();i++)
  {
     cout<<Sort_A[i]<<" ";
  }
  cout<<"\n";
}
```

Answer to the question no-02

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
  int n;
  cin>>n;
  vector<int>A(n-1);
  for(int i=0;i<n-1;i++)
  {
     cin>>A[i];
  }
  sort(A.begin(),A.end());
  for(int i=0;i<n;i++)
  {
     if(A[0]+i!=A[i])
     {
        cout<<A[0]+i;
        break;
     }
  }
}
```

Answer to the question no-03

```
#include<bits/stdc++.h>
using namespace std;
int a,sum=0;
vector<int>Merge(vector<int>A)
{
   if(A.size()<=1)
   {
      return A;
   }
   int m=A.size()/2;
   vector<int>B,C;
   for(int i=0; i<m; i++)
   {
      B.push_back(A[i]);
   }
   for(int i=m; i<A.size(); i++)
   {
      C.push_back(A[i]);
   }
   vector<int>Sort_B=Merge(B);
   vector<int>Sort_C=Merge(C);
   vector<int>Sort_A;
   int id1=0,id2=0,s=A.size();
   for(int i=0; i<s; i++)
   {
      if(id1==Sort_B.size())
      {
         Sort_A.push_back(Sort_C[id2]);
         id2++;
      }
      else if(id2==Sort_C.size())
      {
         Sort_A.push_back(Sort_B[id1]);
         id1++;
      }
      if(Sort_B[id1]<Sort_C[id2])
```

```
        {
            Sort_A.push_back(Sort_B[id1]);
            id1++;
        }
        else
        {
            Sort_A.push_back(Sort_C[id2]);
            id2++;
        }
    }
    for(int i=0; Sort_A[i]<a; i++)
    {
        for(int j=i+1; Sort_A[j]+Sort_A[i]==a; j++)
        {
            sum++;
        }
    }
    return Sort_A;
}
int main()
{
    int n;
    cin>>n;
    vector<int>b(n);
    for(int i=0; i<n; i++)
    {
        cin>>b[i];
    }
    cin>>a;
    vector<int>A=Merge(b);
    cout<<sum<<"\n";
}
```

Answer to the question no-04

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
   int m,n;
   cin>>m;
   vector<int>A(m);
   for(int i=0;i<m;i++)
   {
      cin>>A[i];
   }
   sort(A.begin(),A.end());
   cin>>n;
   vector<int>B(n);
   for(int i=0;i<n;i++)
   {
      cin>>B[i];
   }
   sort(B.begin(),B.end());
   int c=0,j=0;
   for(int i=0;i<n;i++)
   {
      if(B[i]==A[j]){c++;j++;}
   }
   if(c==m){cout<<"YES\n";}
   else{cout<<"NO\n";}
}
```

Answer to the question no-05

```
#include<bits/stdc++.h>
using namespace std;
class node
{
public:
   int data;
   node *nxt;
};
class LinkedList
{
public:
   node *head;
   LinkedList()
   {
      head=NULL;
   }
   node *CreateNewNode(int value)
   {
      node *newnode=new node;
      newnode->data=value;
      newnode ->nxt=NULL;
   }
   void InsertAtHead(int value)
   {
      node *a=CreateNewNode(value);
      if(head==NULL)
      {
         head=a;
         return;
      }
      else
      {
         a->nxt=head; head=a;
      }
   }
   void Traverse()
```

```
{
   node *a=head;
   while(a!=NULL)
   {
      cout<<a->data<<" ";
      a=a->nxt;
   }
   cout<<"\n";
}
int getSize()
{
    node *a=head;
   int id=0;
   while(a!=NULL)
   {
      id++;
      a=a->nxt;
   }
   return id;
}
int getValue(int value)
{
   node *a=head;
   int id=0;
   while(a!=NULL)
   {
      if(a->data==value){return id;}
      a=a->nxt;id++;
   }
   return -1;
}
node *a=head;
void printReverse()
{
   int z;
   if(a!=NULL)
   {
```

```
            z=a->data;
            a=a->nxt;
            printReverse();
        }
        cout<<z<<" ";
        cout<<"\n";
    }
    void swapFirst()
    {
        node *a=head;
        int z=a->data;
        node *b=a->nxt;
        a->data=b->data;
        b->data=z;
    }
};
int main()
{
    LinkedList l;
    cout<<l.getSize()<<"\n";
    l.InsertAtHead(5);
    cout<<l.getSize()<<"\n";
    l.InsertAtHead(6);
    l.InsertAtHead(30);
    cout<<l.getSize()<<"\n";
    l.InsertAtHead(20);
    l.InsertAtHead(30);

    cout<<l.getValue(2)<<"\n";

    cout<<l.getValue(6)<<"\n";

    l.printReverse();
    l.Traverse();
    l.swapFirst();
    l.Traverse();
    l.printReverse();
```

```
}
```

Answer to the question no-06

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
   int n,k,x=0;
   cin>>n;
   vector<int>A(n);
   for(int i=0;i<n;i++)
   {
      cin>>A[i];
   }
   cin>>k;
   int l=0,h=n-1;
   while(l<=h)
   {
      int m=(l+h)/2;
      if(A[m]==k)
      {
         x++;
      }
      if(A[m]>k)
      {
         h=m-1;
      }
      else
      {
         l=m+1;
      }
   }
   if(x>1)
   {
      cout<<"YES\n";
```

```
    }
    else
    {
        cout<<"NO\n";
    }
}
```

Answer to the question no-07

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin>>n;
    vector<int>A(n+1);
    for(int i=1;i<=n;i++)
    {
        cin>>A[i];
    }
    int a,b;
    cin>>a>>b;
    for(int i=1;i<=n;i++)
    {
        if(i>=a&&i<=b){continue;}
        cout<<A[i]<<" ";
    }
}
```

Answer to the question no-08

```
#include<bits/stdc++.h>
using namespace std;
vector<int>even_generator(vector<int>A)
{
    vector<int>B;
    for(int i=0;i<A.size();i++)
```

```
    {
        if(A[i]%2==0)
        {
            B.push_back(A[i]);
        }
    }
    return B;
}
int main()
{
    vector<int>A;
    vector<int>B=even_generator(A);
}
```