**DAY-1 Solution**

# 1.Reverse a String

```
public static String reverseWord(String str)
   {
      // Reverse the string str
     String rev="";
     for(int i=str.length()-1;i>=0;i--)
     {
        rev+=str.charAt(i);
     }
     return rev;
   }
```

**2.Maximum & Minimum Element in an Array: -**

```
#include<bits/stdc++.h>
using namespace std;
int main(){
int t;cin>>t;
while(t--){
   int n;
   cin>>n;
   int arr[n];
   for(int i=0;i<n;i++)cin>>arr[i];
   int max=arr[0],min=arr[0];
   for(int i=1;i<n;i++){
      if(max<arr[i])max=arr[i];
      if(min>arr[i])min=arr[i];
   }
   cout<<max<<" "<<min;
}
```

```
return 0;
}
```

## 3. Kth smallest element

```
void swap(int *a,int *b){
   int temp=*a;
   *a=*b;
   *b=temp;
}
int partition(int arr[],int l,int r){
   int x=arr[r],i=l;
   int j;
   for(j=l;j<r;j++)
   {
      if(arr[j]<=x){
         swap(&arr[i++],&arr[j]);
      }
   }
   swap(&arr[i],&arr[j]);
   return i;
}
int randomPartition(int arr[],int l,int r){
   int n=r-l+1;
   int pivot=rand()%n;
   swap(&arr[l+pivot],&arr[r]);
   return partition(arr,l,r);
}
int kthSmallest(int arr[], int l, int r, int k) {
   //code here

   if(k>0&&k<=r-l+1){
   int pos=randomPartition(arr,l,r);
   if(pos-l==k-1)return arr[pos];
   if(pos-l>k-1)
   return kthSmallest(arr,l,pos-1,k);
   return kthSmallest(arr,pos+1,r,k-pos+l-1);
}return INT_MAX;
}
```

## 4.Sort an array of 0s, 1s and 2s

```
public static void sort012(int a[], int n)
   {
      // code here
      int mid=0,i=0,j=n-1;
      while(mid<=j){
         if(a[mid]==0){
            int temp=a[mid];
            a[mid]=a[i];
            a[i]=temp;



            i++;mid++;
         }else if(a[mid]==1){mid++;}else
         {
           int temp=a[mid];
            a[mid]=a[j];
            a[j]=temp;j--;
         }
      }
   }
```

## 5. Move all negative numbers to beginning and positive to end with constant extra space

```
#include<bits/stdc++.h>
using namespace std;
int main(){
int t;
cin>>t;
while(t--){
   int n;
```

```
    cin>>n;
    int arr[n];
    for(int i=0;i<n;i++)
        cin>>arr[i];
    int i=0,j=n-1;
    while(i<=j){
        if(arr[i]<0)
            i++;


        else{
            swap(arr[i],arr[j]);i++;j--;
        }

    }
     for(int i=0;i<n;i++)
        cout<<arr[i]<<" ";

}

}
```

## 6. Calculate Simple Interest

```
class Solution {
  public:
    double simpleInterest(int P, int R, int T) {
        // code here
        double res=double(P*R*T)/100;return res;
    }
};
```

7.Find Largest Of N numbers: -

```
int largest(int arr[], int n)
{
   int res=arr[0];
   for(int i=1;i<n;i++)
   if(res<arr[i])res=arr[i];
   return res;
}
```

8.**Check Prime Or not?**

```
#include<bits/stdc++.h>
using namespace std;
bool check(int num){
if(num<=1)return false;
if(num<=3)return true;
if(num%2==0||num%3==0)return false;
for(int i=5;i<=sqrt(num);i+=6)
{
   if(num%i==0||num%(i+2)==0)return false;


}
return true;
}
int main(){int t;
cin>>t;
while(t--)
   {
      int num;cin>>num;
      if(check(num))cout<<"Prime";
      else
         cout<<"Not Prime";
   }
}
```