```c
#include<stdio.h>
int main()
{
int arr[10];
  int i,n;
  for(i=0;i<=9;i++)
  {
    printf("Enter no:");
    scanf("%d",&arr[i]);
  }
 printf("Enter number to search:");
 scanf("%d",&n);
  for(i=0;i<=9;i++)
  {
      if(arr[i]==n)
      {
         printf("Number is at %d pos",i+1);
         break;
      }
  }
if(i==10)
     printf("Number not found");
return 0;
}
```

i = 0 1 2 3 4 5 6 7 8 9 10

n = 24

	arr
0	15
1	12
2	9
3	34
4	18
5	26
6	14
7	23
8	11
9	35

```c
#include<stdio.h>
int main()
{
  int arr[10];
  int i,n,count=0;
  for(i=0;i<=9;i++)
  {
    printf("Enter no:");
    scanf("%d",&arr[i]);
  }
 printf("Enter number to search:");
 scanf("%d",&n);
 for(i=0;i<=9;i++)
 {
     if(arr[i]==n)
     {
        printf("\nNumber is at %d pos",i+1);
        count++;
     }
 }
if(count==0)
     printf("Number not found");
return 0;
}
```

n = 14

	arr
0	15
1	12
2	14
3	34
4	18
5	26
6	14
7	23
8	11
9	35

WAP to accept 10 integers from the user and store them in an integer array. Now find out the LARGEST element of the array. Make sure that your code should not change the original order of elements in the array

-2147483648 To 2147483647

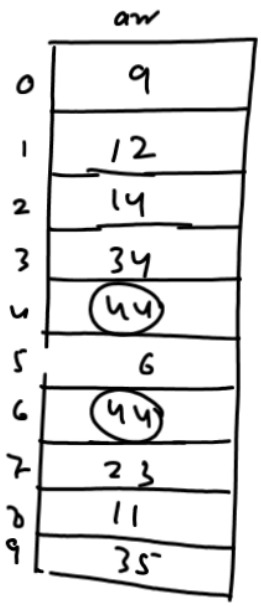
```
#include<stdio.h>
int main()
{
  int arr[10];
  int i,max;
  for(i=0;i<=9;i++)
  {
    printf("Enter no:");
    scanf("%d",&arr[i]);
  }

  for(max=arr[0],i=1;i<=9;i++)
  {
     if(max<arr[i])
        max=arr[i];
  }
printf("Largest no is %d",max);
return 0;
}
```

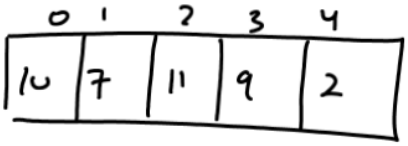
i = 0 1 2 3 4

5

max = 9 - 6



SORTING THE ARRAY



```
#include<stdio.h>
int main()
{
   int arr[5];
   int i,j,temp;
   for(i=0;i<=4;i++)
   {
     printf("Enter no:");
     scanf("%d",&arr[i]);
   }
   for(i=0;i<4;i++)
   {
     for(j=i+1;j<5;j++)
     {
       if(arr[i]>arr[j])
         {
           temp=arr[i];
           arr[i]=arr[j];
           arr[j]=temp;
         }
     }
   }
```

I/p

Sorting

```
   for(i=0;i<=4;i++)
   {
     printf("\n%d",arr[i]);
   }
   return 0;
}
```

O/p