

```
/*LINEAR SEARCH*/
#include <stdio.h>
int main()
{
int array[100], search, i, n;
printf("Enter number of elements in array\n");
scanf("%d", &n);
printf("Enter %d integer(s)\n", n);
for (i = 0; i < n; i++)
scanf("%d", &array[i]);
printf("Enter a number to search\n");
scanf("%d", &search);
for (i = 0; i < n; i++)
{
if (array[i] == search) /* If required element is found */
{
printf("%d is present at location %d.\n", search, i+1);
break;
}
}
if (i == n)
printf("%d isn't present in the array.\n", search);
return 0;
}
```

OUTPUT:

```
Enter number of elements in array
3
Enter 3 integer(s)
5
4
3
Enter a number to search
3
3 is present at location 3.
```

EVALUATION:

Particulars	Marks Allotted	Marks Obtained
Program & Execution	15	
Viva	10	
Total	25	

```
/*BINARY SEARCH*/
#include <stdio.h>
int main()
{
int c, first, last, middle, n, search, array[100];
printf("Enter number of elements\n");
scanf("%d", &n);
printf("Enter %d integers\n", n);
for (c = 0; c < n; c++)
scanf("%d", &array[c]);
printf("Enter value to find\n");
scanf("%d", &search);
first = 0;
last = n - 1;
middle = (first+last)/2;
while (first <= last) {
if (array[middle] < search)
first = middle + 1;
else if (array[middle] == search) {
printf("%d found at location %d.\n", search, middle+1);
break;
}
else
last = middle - 1;

middle = (first + last)/2;
}
if (first > last)
printf("Not found! %d isn't present in the list.\n", search);
return 0;
}
```

OUTPUT:

```
Enter number of elements
3
Enter 3 integers
2
3
4
Enter value to find
3
3 found at location 2.
```

EVALUATION:

Particulars	Marks Allotted	Marks Obtained
Program & Execution	15	
Viva	10	
Total	25	

```

/*INSERTION SORT*/
#include<stdio.h>
#include<conio.h>
int main ()
{
int arr[20], n, i, j, temp;
clrscr();
printf("Enter a size of an array:");
scanf("%d",&n);
printf("Enter the elements of an array:\n");
for(i=0; i<n; i++)
{
scanf("%d",&arr[i]);
}
for(i=1;i<n;i++)
{
temp= arr[i];
j=i-1;
while(j>=0 && arr[j]>)
{
arr[j+1]=arr[j];
j--;
}
arr[j+1] = temp;
}
printf("Elements of an array after sorted\n");
for(i=0; i<n; i++)
{
printf("%d  ",arr[i]);
}
getch();
return 0;
}

```

OUTPUT:

```
Size of array:6
Elements of array:
5
4
10
1
6
2
Enter the elements of an array after sorted:
1 2 4 5 6 10
```

EVALUATION:

Particulars	Marks Allotted	Marks Obtained
Program & Execution	15	
Viva	10	
Total	25	

```

/*SELECTION SORT*/
#include<stdio.h>
#include<conio.h>
int main ()
{
int a[20], n, i, j, min, temp;
clrscr();
printf("Enter a size of an array:");
scanf("%d",&n);
printf("Enter the elements of an array:\n");
for(i=0; i<n; i++)
{
scanf("%d",&a[i]);
}
for(i=0; i<n-1; i++)
{
int min=i;
for(j=i+1; j<n; j++)
{
if(a[j]<a[min])
{
min=j;
}
}
if (min!=i)
{
temp=a[i];
a[i]=a[min];
a[min]=temp;
}
}
printf("Elements of an array after sorted\n");
for(i=0; i<n; i++)
{
printf("%d ",a[i]);
}
getch();
return 0;
}

```

OUTPUT:

```
Size of array:6
Elements of array:
7
4
10
8
3
1
Enter the elements of an array after sorted:
1 3 4 7 8 10
```

EVALUATION:

Particulars	Marks Allotted	Marks Obtained
Program & Execution	15	
Viva	10	
Total	25	


```

/*BUBBLE SORT*/
#include<stdio.h>
#include<conio.h>
int main ()
{
int a[20], n, i, j, temp;
clrscr();
printf("Enter a size of an array:");
scanf("%d",&n);
printf("Enter the elements of an array:\n");
for(i=0; i<n; i++)
{
scanf("%d",&a[i]);
}
for(i=0; i<n-1; i++)
{
for(j=0; j<n-1; j++)
{
if( a[j] > a[j+1] )
{
temp=a[j];
a[j]= a[j+1];
a[j+1]=temp;
}
}
}
printf("Elements of an array after sorted\n");
for(i=0; i<n; i++)
{
printf("%d  ",a[i]);
}
getch();
return 0;
}

```

OUTPUT:

```
Size of array:6
Elements of array:
4
7
10
8
3
1
Enter the elements of an array after sorted:
1 3 4 7 8 10
```

EVALUATION:

Particulars	Marks Allotted	Marks Obtained
Program & Execution	15	
Viva	10	
Total	25	

```

/*SHELL SORT*/
#include<stdio.h>
#include<conio.h>
int main ()
{
int arr[20], n, i, j, gap, temp;
clrscr();
printf("Enter a size of an array:");
scanf("%d",&n);
printf("Enter the elements of an array:\n");
for(i=0; i<n; i++)
{
scanf("%d",&arr[i]);
}
for(gap=n/2; gap>0; gap=gap/2)
{
for(j=gap; j<n; j++)
{
for(i=j-gap; i>=0; i=i-gap)
{
if(arr[i+gap]>arr[i])
{
break;
}
else
{
temp= arr[i];
arr[i]= arr[i+gap];
arr[i+gap]=temp;
}
}
}
}
printf("Elements of an array after sorted\n");
for(i=0; i<n; i++)
{
printf("%d  ",arr[i]);
}
getch();
return 0;
}

```

OUTPUT:

```
Size of array:9
Elements of array:
23
29
15
19
31
7
9
5
2
Enter the elements of an array after sorted:
2 5 7 9 15 19 23 29 31
```

EVALUATION:

Particulars	Marks Allotted	Marks Obtained
Program & Execution	15	
Viva	10	
Total	25	

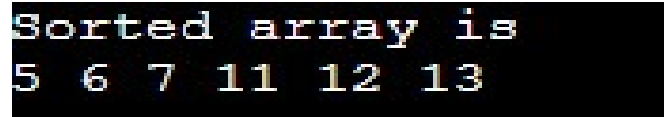
```

/*HEAP SORT*/
#include <stdio.h>
void swap(int* a, int* b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}
void heapify(int arr[], int N, int i)
{
    int largest = i;
    int left = 2 * i + 1;
    int right = 2 * i + 2;
    // If left child is larger than root
    if (left < N && arr[left] > arr[largest])
        largest = left;
    if (right < N && arr[right] > arr[largest])
        largest = right;
    if (largest != i)
    {
        swap(&arr[i], &arr[largest]);
        heapify(arr, N, largest);
    }
}
void heapSort(int arr[], int N)
{
    for (int i = N / 2 - 1; i >= 0; i--)
        heapify(arr, N, i);
    for (int i = N - 1; i >= 0; i--)
    {
        swap(&arr[0], &arr[i]);
        heapify(arr, i, 0);
    }
}
void printArray(int arr[], int N)
{
    for (int i = 0; i < N; i++)
        printf("%d ", arr[i]);
    printf("\n");
}
int main()
{
    int arr[] = { 12, 11, 13, 5, 6, 7 };
    int N = sizeof(arr) / sizeof(arr[0]);

```

```
heapSort(arr, N);  
printf("Sorted array is\n");  
printArray(arr, N);  
}
```

OUTPUT:



```
Sorted array is  
5 6 7 11 12 13
```

EVALUATION:

Particulars	Marks Allotted	Marks Obtained
Program & Execution	15	
Viva	10	
Total	25	