



# KRISHNA ENGINEERING COLLEGE

(Approved by All India Council for Technical Education, Affiliated to Dr A.P.J Abdul Kalam Technical University)

95, Loni Road, Mohan Nagar, Ghaziabad, Uttar Pradesh.

## DATA STRUCTURE (KCS -301)

## LAB PRACTICAL FILE

NAME: ***KISHAN KUMAR.***

ROLL NO.: ***1901610100090.***

## PROGRAM: 1

### IMPLEMENTATION OF TRAVERSE

#### INPUT:

```
#include <stdio.h>

int main()
{
    int i, size, a[20];
    printf("Enter no. of elements of array : \n");
    scanf("%d", &size);

    if (size > 20)
    {
        printf("Overflow");
    }
    else
    {
        printf("Elements of array : \n");
        for (i = 0; i < size; i++)
        {
            scanf("%d", &a[i]);
        }
        printf("Elements are: ");
        for (i = 0; i < size; i++)
        {
            printf("%d ", a[i]);
        }
    }
}
```

## OUTPUT :

PROBLEMS   OUTPUT   TERMINAL   DEBUG CONSOLE

Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> cd "c:\Users\ASUS\Documents\2nd year\labs practical\ds"

Enter no. of elements of array :

6

Elements of array :

1

4

9

5

3

6

Elements are: 1 4 9 5 3 6

PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> █

## PROGRAM : 2

### IMPLEMENTATION OF BINARY SEARCH

#### INPUT :

```
#include <stdio.h>

int main()
{
    int arr[50], i, n, x, flag = 0, first, last, mid;

    printf("Enter size of array:");
    scanf("%d", &n);
    printf("\nEnter array element(ascending order)\n");

    for (i = 0; i < n; ++i)
        scanf("%d", &arr[i]);

    printf("\nEnter the element to search:");
    scanf("%d", &x);

    first = 0;
    last = n - 1;

    while (first <= last)
    {
        mid = (first + last) / 2;

        if (x == arr[mid])
        {
            flag = 1;
            break;
        }
        else if (x > arr[mid])
            first = mid + 1;
        else
            last = mid - 1;
    }

    if (flag == 1)
        printf("\nElement found at position %d", mid + 1);
    else
        printf("\nElement not found");

    return 0;
}
```

## OUTPUT :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> cd "c:\U
($?) { .\BinarySearch }
Enter size of array:6

Enter array element(ascending order)
4
8
9
11
25
44

Enter the element to search:11

Element found at position 4
PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> █
```

## PROGRAM : 3

### IMPLEMENTATION OF LINEAR SEARCH

#### INPUT :

```
#include <stdio.h>

int main()
{
    int a[20], i, x, n;
    printf("How many elements? \n");
    scanf("%d", &n);

    printf("Enter array elements:\n ");
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);

    printf("\nEnter element to search: \n");
    scanf("%d", &x);

    for (i = 0; i < n; ++i)
        if (a[i] == x)
            break;

    if (i < n)
        printf("Element found at index %d", i);
    else
        printf("Element not found");
}
```

## OUTPUT :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> cd "c:\Users\ASUS\Do
($?) { .\LinearSearch }
How many elements?
5
Enter array elements:
1
4
2
5
7

Enter element to search:
5
Element found at index 3
PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> |
```

## PROGRAM : 4

### IMPLEMENTATION OF INSERTION SORT

#### INPUT :

```
#include <stdio.h>

int main()
{
    int i, j, n, temp, a[30];
    printf("Enter the number of elements:");
    scanf("%d", &n);
    printf("\nEnter the elements\n");

    for (i = 0; i < n; i++)
    {
        scanf("%d", &a[i]);
    }

    for (i = 1; i <= n - 1; i++)
    {
        temp = a[i];
        j = i - 1;

        while ((temp < a[j]) && (j >= 0))
        {
            a[j + 1] = a[j]; //moves element forward
            j = j - 1;
        }

        a[j + 1] = temp; //insert element in proper place
    }

    printf("\nSorted list is as follows\n");
    for (i = 0; i < n; i++)
    {
        printf("%d ", a[i]);
    }

    return 0;
}
```



## OUTPUT :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> cd "c:\Users\A
if ($?) { .\InsertionSort }
Enter the number of elements:5

Enter the elements
12
4
8
6
3

Sorted list is as follows
3 4 6 8 12
PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> |
```

## PROGRAM : 5

### IMPLEMENTATION OF SELECTION SORT

#### INPUT :

```
#include <stdio.h>
void selection(int a[], int size);
int main()
{
    int a[10], size, i;

    printf("\nenter size of array:");
    scanf("%d", &size);
    printf("enter array elements:");
    for (i = 0; i < size; i++)
        scanf("%d", &a[i]);
    selection(a, size);
    return 0;
}
void selection(int a[], int size)
{
    int min, temp, i;
    for (int i = 0; i < size - 1; i++)
    {
        min = i;
        for (int j = i + 1; j < size; j++)
        {
            if (a[min] > a[j])
                min = j;
        }
        if (min != i)
        {
            temp = a[min];
            a[min] = a[i];
            a[i] = temp;
        }
    }
    printf("\nsorted array is:");
    for (i = 0; i < size; i++)
        printf("%d    ", a[i]);
}
```

## OUTPUT :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> cd "c:\U
if ($?) { .\SelectionSort }

enter size of array:6
enter array elements:12
4
9
6
58
74

sorted array is:4    6    9    12    58    74
PS C:\Users\ASUS\Documents\2nd year\labs practical\ds> █
```