* Problem 1

**Linear Search**

**AIM :-** To write a ‘c’ Program to take input of the Numbers and search the element in the Array and display the location of the element**.**

**Theory :-** The collection of multiple Data in a single variable is known as **Array**. For example it is written as a[100];, a[5]={1,2,3,4,5};, a[]={1,2,3,4,5}; .

**Algorithms :-** Step 1:- Start.

Step 2:- Declare integer variable a for array, i, j for loops and ‘r’, pos=-1.

Step 3:- Check FOR condition i<?, if it is true go to step 4 otherwise go to step 6.

Step 4:- Read variable ‘a’ by For loop.

Step 5:- Increment i+1 and go to step 3.

Step 6:- Read variable ‘r’ to search the element.

Step 7:- Check FOR condition i<?, if it is true go to step 8 otherwise go to step 12.

Step 8:- check condition if(a[j]==r), if it is true go to step 9 otherwise go to step 11.

Step 9:- Assign “pos” by ‘j’, i.e pos=j;

Step 10:- Print the value of “pos” variable.

Step 11:- Increment i+1 and go to step 7.

Step 12:- check condition if(pos==-1), if it is true go to step 13 otherwise go to step 14.

Step 13:- Print Value not found.

Step 14:- Stop.

**Program :-**

#include<stdio.h>

int main()

{

int a[6];

int i,j,r,pos=-1;

printf("Enter the Values in the Array\n");

for(i=0;i<6;i++)

scanf("%d",&a[i]);

printf("Enter the value you want to Search??\n");

scanf("%d",&r);

for(j=0;j<6;j++)

{

if(a[j]==r)

{

pos=j;

printf("Position of the Value is %d\n",pos);

}

}

if(pos==-1)

printf("Value Not Found!!!!");

return 0;

}

**Output :-**

Enter the Values in the Array

1

6

7

3

5

9

Enter the value you want to Search??

7

Position of the Value is 2

**Observation :-** After performing the experiment we observed that Values can be stored in a single variable using Array and we were able to take input of many values using looping statements and also were able to search the element in a single dimension Array. It took 0.24 sec Compilation Time.

* Problem 2

**Binary Search**

**AIM :-** To write a ‘c’ Program to search the element in the Array, when an **Array** is **Sorted** and display the location of the element**.**

**Theory :-** The collection of multiple Data in a single variable is known as **Array**. For example it is written as a[100];, a[5]={1,2,3,4,5};, a[]={1,2,3,4,5}; .

**Algorithms :-** Step 1:- Start.

Step 2:- Declare integer variable ‘a’ for array, ‘I’ for loops and ‘f’, ‘low’, ‘high’, ‘mid’.

Step 3:- Read variable ‘f’ to find the element.

Step 4:- Initialize low=0 and high=9.

Step 5:- Check **While** condition low<=high, if it is true go to step 6 otherwise go to step 14.

Step 6:- Assign ‘mid’, mid=(low+high)/2.

Step 7:- Check **if** condition a[mid]==f, if it is true go to step 8 otherwise go to step 10.

Step 8:- Print the element’s Location.

Step 9:- Break the loop by ‘break’ statement and go to step 14.

Step 10:- check **else if** condition a[mid]>f, if it is true go to step 11 otherwise go to step 12.

Step 11:- Update ‘high’ with high=mid +1 and go to step 5.

Step 12:- **Else**, go to step 13.

Step 13:- Update ‘low’ with low=mid -1 and go to step 5.

Step 14:- Stop.

**Program :-**

#include<stdio.h>

int main()

{

int a[10]={2,4,7,9,10,13,16,20,25,30};

int i,f,low,high,mid;

printf("Enter the Element to Search\n");

scanf("%d",&f);

low=0;

high=9;

while(low<=high)

{

mid=(low+high)/2;

if(a[mid]==f)

{

printf("The Element is present in %d location\n",mid);

break;

}

else if(a[mid]>f)

{

high=mid-1;

}

else

{

low=mid+1;

}

}

return 0;

}

**Output :-**

Enter the Element to Search

7

The Element is present in 2 location

**Observation :-** After performing the experiment we observed that Values can be stored in a single variable using Array and we were able to search the element from the **Sorted** Array in a single dimension Array. It divides the Array into 2 parts until the element is found. It took 0.31 sec Compilation Time.