**1. Write a C program to count distinct element in an array**

#include <stdio.h>

void main()

{

int arr1[100], n,ctr=0;

int i.j.k;

printf("\n\nPrint all unique elements of an array:\n");

printf("------------------------------------------\n");

printf("Input the number of elements to be stored in the array :");

scanf("%d",&n);

printf("Input %d elements in the array :\n",n);

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

{

printf("element - %d : ",i);

scanf("%d",&arr1[i]);

}

/\*Checking duplicate elements in the array \*/

printf("\nThe unique elements found in the array are : \n");

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

{

ctr=0;

/\*Check duplicate bifore the current position and increase counter by 1 if found.\*/

for(j=0; j<i-1; j++)

{

/\*Increment the counter when the seaarch value is duplicate.\*/

if(arr1[i]==arr1[j])

{

ctr++;

}

}

/\*Check duplicate after the current position and

increase counter by 1 if found.\*/

for(k=i+1; k<n; k++){

/\*Increment the counter when the seaarch value is duplicate.\*/

if(arr1[i]==arr1[k])

{

ctr++;

}

}

/\*Print the value of the current position of the array as unique value when counter remain contains its initial value.\*/

if(ctr==0)

{

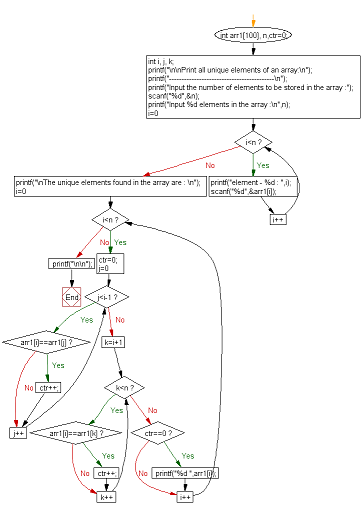
printf("%d ",arr1[i]);

}

}

printf("\n\n");

}



Algorithm:

Step1:Declare and variables and array

Step2:Input the number of elements

Step3: Input the array elements.

Step4: Traverse the array from beginning.

Step5:Check if the current element is found in the array,

Step6:If it is found, then do not print that element.

Step7:Else print the element and continue.