

## C PROGRAM :

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

```
include <stdio.h>

void distict_elements(int a[], int n);

int main()
{
    int size_array, i, arr[20];
    scanf("%d", &size_array);
    for(i=0; i<size_array; i++)
    {
        scanf("%d", &arr[i]);
    }
    distict_elements(arr, size_array);
    return 0;
}

void distict_elements(int a[], int n)
{
    int i, j;
    for (i=0; i<n; i++)
    {
        for (j=0; j<i; j++)
        {
            if (a[i] == a[j])
                break;
        }
        if (i == j)
            printf("%d ", a[i]);
    }
}
```

## SNAPSHOT OF C PROGRAM:

```
1 #include <stdio.h>
2
3 void distict_elements(int a[], int n);
4 int main()
5 {
6     int size_array, i, arr[20];
7     // Get the array size
8     scanf("%d", &size_array);
9     // Get the array elements
10    for(i=0; i<size_array; i++)
11    {
12        scanf("%d", &arr[i]);
13    }
14    // Function call to print the distinct elements in an array
15    distict_elements(arr, size_array);
16    return 0;
17 }
```

```
17 }
18 void distict_elements(int a[], int n)
19 {
20     int i, j;
21     // Pick all elements one by one
22     for (i=0; i<n; i++)
23     {
24         // Check if the picked element is already printed
25         for (j=0; j<i; j++)
26         {
27             if (a[i] == a[j])
28                 break;
29         }
30         // If not printed earlier, then print it
31         if (i == j)
32             printf("%d ", a[i]);
33     }
```

### Output

Input- Enter the size of the array:5 Enter the array elements:1 2 3 4 4 Output- Input Array:1 2 3 4 4 Distinct Elements:1 2 3 4

## **ALGORITHM:**

**STEP1:** Declare and input the array elements.

**STEP2:** Traverse the array from the beginning.

**STEP3:** Check if the current element is found in the array again.

**STEP4:** if it is found, then do not print the element.

**STEP5:** Else, print that element and continue.

## FLOWCHART:

