1.Array rotation for k times(left or right)

(LEFT ROTATE)

```
#include<stdio.h>
int main() {
    int a[10],i,j,k,n,temp;
    scanf("%d",&n);
printf("enter elements:");
    for(i=0;i<n;i++){
scanf("%d ",&a[i]);
}
        scanf("%d",&k);
    for(j=1;j<=k;j++){
    temp=a[0];
    for(i=0;i<n-1;i++){
        a[i]=a[i+1];
    }
    a[i]=temp;
}
    for(i=0;i<n;i++){
printf("%d ",a[i]);

    }

}
```

(RIGHT ROTATE)

```
#include<stdio.h>
int main() {
    int a[10],i,j,k,n,temp;
    scanf("%d",&n);
printf("enter elements:");
    for(i=0;i<n;i++)
scanf("%d ",&a[i]);
        scanf("%d",&k);
    for(j=1;j<=k;j++){
    temp=a[n-1];
    for(i=n-1;i>=1;i--){
        a[i]=a[i-1];
    }
    a[i]=temp;
}
    for(i=0;i<n;i++){
printf("%d ",a[i]);
    }
    printf("\n");


}
```

2. give c program to print the number of occurance of a number in an array

```
#include<stdio.h>
int main(){
    int s,i,j;
    scanf("%d",&s);
    int arr[s];
for(int i=0;i<s;i++){
        scanf("%d",&arr[i]);
    }
    for(i=0;i<s;i++){
        int count=1;
        if(arr[i]!=-1){
            for(j=i+1;j<s;j++){
                if(arr[i]==arr[j]){
                    count++;
                    arr[j]=-1;
            }
        }
printf("%d ocuurs %d times\n",arr[i],count );

                }
    }
}
```

```
#include <stdio.h>
```

```
int main() {
    int arr[] = {1, 2, 3, 4, 5, 2, 3, 4, 2};
    int n = sizeof(arr) / sizeof(arr[0]);
    int number_to_find = 2;

    int count = 0;
    int sum = 0;
    int product = 1;

    for (int i = 0; i < n; i++) {
        if (arr[i] == number_to_find) {
            count++;
        }
    }
printf(“count is %d” ,count);
}
```

3.Sum and product of array

```
#include<stdio.h>
int main(){
    int n,i;
    int myarr[]={2,1,3,4,5,6};
    n=sizeof(myarr)/sizeof(myarr[0]);
     int sum=0;
    int product=1;
    for(i=0;i<n;i++){
          sum+=myarr[i];
          product*=myarr[i];
    }
printf("sum is %d\n" ,sum);
printf("pro is %d" ,product);
}
```

4.Print square of the elements in an array

```
#include <stdio.h>
int main() {s
    int arr[] = {2, 2, 3, 4, 5};
    int n = sizeof(arr) / sizeof(arr[0]);

printf("Square of elements in the array:\n");
    for (int i = 0; i < n; i++) {
printf("%d ", arr[i] * arr[i]);
    }
    printf("\n");

    return 0;
}
```

5.Difference between maximum and minimum element of an array

```
#include<stdio.h>
int main(){
    int s,i;
    scanf("%d",&s);
    int arr[s];
    for(i=0;i<s;i++){
        scanf("%d",&arr[i]);
    }
    int min=arr[0];
    int max=arr[0];
    for(i=1;i<s;i++){
        if(arr[i]>max){
            max=arr[i];
        }
        if(arr[i]<min){
            min=arr[i];
        }
    }
printf("%d is min\n",min);
printf("%d is max\n",max);
printf("%d is diff",max-min);
}
```

6.alternate number print in array

```
#include<stdio.h>
```

```
void printalternate( int arr[], int size){
for(int i = 1;i<size;i+=2){

printf("%d ",arr[i]);
}
printf("\n");
}

int main(){
    int arr[]={1,3,4,5,3,5,9};
    int size=sizeof(arr)/sizeof(arr[0]);
    printalternate(arr,size);
}
```

(WITHOUT FUNC)

```
#include<stdio.h>
int main(){
    int i,n;
    scanf("%d",&n);
int  arr[n];
for( i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
 for(i=1;i<n;i+=2){
printf("%d ",arr[i]);
 }
 printf("\n");
}
```

7.count of array elements divisible by specific number

```
#include<stdio.h>
int main(){
    int i,n,div,count=0;
    scanf("%d",&n);
int  arr[n];
for( i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    scanf("%d",&div);

 for(i=0;i<n;i++){
     if(arr[i]%div==0){
         count++;
     }
 }
printf("Count of array elements divisible by %d: %d\n", div, count);
}
```

8.Sum the array after removing duplicate elements

```
#include <stdio.h>
int main() {
   int i,n,j,k,sum=0;
    scanf("%d",&n);
int  arr[n];
for( i=0;i<n;i++){
        scanf("%d",&arr[i]);
}
for(i=0;i<n-1;i++){
for(j=i+1;j<n;){
        if(arr[i]==arr[j]){
            for(k=j;k<n-1;k++){
                arr[k]=arr[k+1];
            }
            n--;
        }
else{
            j++;
        }
    }
}for(i=0;i<n;i++){
        sum+=arr[i];
    }
printf("%d ",sum);
}
```

7.Print negative elements in array

```
#include <stdio.h>
int main() {
   int i,n;
    scanf("%d",&n);
int  arr[n];
for( i=0;i<n;i++){
        scanf("%d",&arr[i]);
}
for(i=0;i<n;i++){
    if(arr[i]<0){
printf("%d ",arr[i]);
    }
}
printf("\n");
}
```

Non negative elements array print

```
#include <stdio.h>
int main() {
   int i,n;
    scanf("%d",&n);
int  arr[n];
for( i=0;i<n;i++){
        scanf("%d",&arr[i]);
}
int newarr[n];
int nn=0;
for(i=0;i<n;i++){
    if(arr[i]>=0){
        newarr[nn]=arr[i];
        nn++;
}

}
for(i=0;i<nn;i++){
printf("%d ",newarr[i]);
}
}
```

8.Print the peak elements in array

```
#include<stdio.h>
int main(){
    int n;
printf("Enter no of elements: ");
   scanf("%d",&n);
    int arr[n];
for(int i=0;i<n;i++){
       scanf("%d",&arr[i]);
    }
    int k=0;
printf("The peak elements are: ");
    if(arr[k]>arr[k+1]){
printf("%d ",arr[k]);
       }
for(int i=1;i<n-1;i++){
      if(arr[i]>arr[i+1] && arr[i]>arr[i-1]){
printf("%d ",arr[i]);
      }
}
if(arr[n-1]>arr[n-2]){
printf("%d ",arr[n-1]);
       }
}
```

9. Print the count of positive numbers

```
#include<stdio.h>
int main(){
    int n,i,count=0;
printf("Enter no of elements: ");
   scanf("%d",&n);
getchar();
    int arr[n];
for(int i=0;i<n;i++){
       scanf("%d",&arr[i]);
    }
    for (int i = 0; i < n; i++) {
        if (arr[i] > 0) {
            count++;
        }
    }

printf("Count of positive numbers in the array: %d\n", count);

    return 0;
}
```

10.Delete the element in the given position in an array

```
#include<stdio.h>
int main(){
    int size,i;
printf("Enter no of elements: ");
   scanf("%d",&size);
    int arr[size];
    for(i=0;i<size;i++){
        scanf("%d",&arr[i]);
    }
    int positionToDelete=1;
  if (positionToDelete < 1 || positionToDelete > size) {
printf("Invalid position.\n");
        return 0;
    }
    for (int i = positionToDelete; i < size - 1; i++)(this is position to delete)
                                                      //(i=positiontodelete-1;i<n-1;i++) (this is index to
delete)
      {
  arr[i] = arr[i + 1];
    }
    size--;
printf("Array after deleting element at position %d:\n", positionToDelete);
    for (int i = 0; i < size; i++) {
printf("%d ", arr[i]);
    }
    printf("\n");
}
```

11. sum of duplicates in an array

```
#include<stdio.h>
int main(){
    int size,i,sumOfDuplicates=0;
printf("Enter no of elements: ");
   scanf("%d",&size);
    int arr[size];
    for(i=0;i<size;i++){
        scanf("%d",&arr[i]);
    }

    for (int i = 0; i < size - 1; i++) {
        for (int j = i + 1; j < size; j++) {
            if (arr[i] == arr[j]) {
                sumOfDuplicates += arr[i];
                break;
            }
        }
    }

printf("Sum of duplicate elements in the array: %d\n", sumOfDuplicates);

    return 0;
}
```

12.sum of even numbers in an array

```
#include<stdio.h>
int main(){
    int size,i,sum=0;
printf("Enter no of elements: ");
   scanf("%d",&size);
    int arr[size];
    for(i=0;i<size;i++)
{
        scanf("%d",&arr[i]);
    }
    for(i=0;i<size;i++)   // for(i=0;i<size;i+=2) -> (even place even number sum)
{
        if(arr[i]%2==0){
            sum+=arr[i];
        }
    }
printf("%d ",sum);

}
```

13.find the non prime numbers in an array

```
#include<stdio.h>

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

for(int i=0;i<n;i++){
for(int j=2;j<=arr[i]/2;j++){
            if(arr[i]%j==0){
printf("%d ",arr[i]);
 break;
            }
        }
    }
    }
```

14.Replace the peak number by adding it neighbor elements (example:- input: 1 2 4 3 output: 1 2 9 3

```
#include<stdio.h>
int main(){
    int n,i;
printf("Enter no of elements: ");
```

```
   scanf("%d",&n);
getchar();
    int arr[n];
for(int i=0;i<n;i++){
       scanf("%d",&arr[i]);
    }
    int k=0;
printf("elements added are ");
    if(arr[k]>arr[k+1]){
        arr[k]=arr[k]+arr[k+1];
        }
    for(i=1;i<n-1;i++){
        if(arr[i]>arr[i+1]&&arr[i]>arr[i-1]){
            arr[i]=arr[i]+arr[i+1]+arr[i-1];
        }
    }
    if(arr[n-1]>arr[n-2]){
        arr[n-1]=arr[n-1]+arr[n-2];
    }
for(int s=0;s<n;s++){
printf("%d ",arr[s]);
    }
}
```

17.Print the sum of first element, second element,last element,second last element in an array Example : [1,2,4,5,5] Output: 1+2+5+5 = 13

```
#include<stdio.h>

int main() {
    int i, n,j;
scanf("%d", &n);
    int arr[n];
for(i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
    int sum=0;
    if(n>=4){
        sum = arr[0]+arr[1]+arr[n-1]+arr[n-2];
}else if(n<=3){
        sum = arr[0]+arr[n-1];
    }
    printf("%d",sum);
}
```

Sorting array in ascending order /descending

```
#include<stdio.h>
int main(){
    int i,n,j,temp;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n-1;i++){
        for(j=i+1;j<n;j++){
            if(arr[i]>arr[j]){                   // if(arr[i]<arr[j])
            temp=arr[i];
            arr[i]=arr[j];
            arr[j]=temp;
            }
        }
    }
    for(i=0;i<n;i++){
printf("%d ",arr[i]);
    }
}
```

18.Print the median of an array after sorting

```
#include<stdio.h>

int main() {
    int i, n, j, temp, median;
scanf("%d", &n);
    int arr[n];
for(i = 0; i < n; i++) {
```

```
scanf("%d", &arr[i]);
    }
for(i=0;i < n-1; i++){
    for(j=i+1;j<n;j++){
        if(arr[i]>arr[j]){
            temp=arr[i];
            arr[i]=arr[j];
            arr[j]=temp;
        }
    }
}
if(n%2==0){
printf("%d %d\n ",arr[n/2-1],arr[n/2]);
}
else{
printf("%d ",arr[n/2]);
}
}
```

19.Print the average of an array

```
#include<stdio.h>

int main() {
    int i, n,sum=0;
scanf("%d", &n);
    int arr[n];
for(i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
for(i=0;i<n;i++){
    sum+=arr[i];
}
double average = (double)sum/n;
printf("%.2f ",average);
}
```

20.Array Input range 2-10 if exceeds print "Invalid". print two elements which is the sum of two elements closest to 0(zero). Input: [-1,-10,8,2] output: [-1,2]. Explanation -1+2=-1 is the closest to 0 than other combinations.

```
#include <stdio.h>
#include <limits.h>
#include<stdlib.h>
int main() {
    int n,i,j;
    scanf("%d",&n);
    if(n<2||n>10){
printf("Invalid input");
    }

    int arr[n];
for(int i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
int cpair[2];
int csum=INT_MAX;
for(i=0;i<n;i++){
    for(j=0;j<n;j++){
        if(arr[i]+arr[j]< csum &&i!=j){  (0th index wont enter into loop )
            csum=arr[i]+arr[j];
            cpair[0]=arr[i];
cpair[1] =arr[j];
        }
    }
}
printf("closest pair sum %d\n",csum);
printf("closest num %d %d ",cpair[0],cpair[1]);
}
```

PRINTING INDEX OF TWO ELEMENT

```
#include <stdio.h>
#include <stdlib.h>
#include <limits.h>
#include <math.h>

int main() {
    int nums[] = {-1,-2,-3};
    int n = sizeof(nums) / sizeof(nums[0]);
    int minSum = INT_MAX;
    int index1, index2;

printf("Array: ");
    for (int i = 0; i < n; i++) {
printf("%d ", nums[i]);
    }

    printf("\n");

    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            int sum = nums[i] + nums[j];
            if (abs(sum) < abs(minSum)) {
                minSum = sum;
                index1 = i;
                index2 = j;
            }
        }
    }

printf("Indices of elements with sum closest to zero: %d and %d\n", index1, index2);

    return 0;
}
```

21.print the unique elements in array Example: input = [0,5,3,5,2,3] output = 0 and 2

```
#include <stdio.h>

int main() {
    int n, i,j;
printf("Enter the number of elements in the array: ");
scanf("%d", &n);

    int arr[n];
printf("Enter %d elements:\n", n);
for(i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }

int count=0;
for(i=0;i<n;i++){
    count=0;
    for(j=0;j<n;j++){
        if(arr[i]==arr[j] &&i!=j){
            count++;
    break;
        }
    }

    if(count==0){

printf("%d ",arr[i]);
}
}
}
```

22.Find the second largest element in the array

```
    int largest = arr[0];
    int secondLargest = arr[0];

for(i = 1; i < n; i++) {
        if(arr[i] > largest) {
            secondLargest = largest;
            largest = arr[i];
        } else if(arr[i] > secondLargest && arr[i] != largest) {
            secondLargest = arr[i];
        }
    }

printf("Second largest element: %d\n", secondLargest);

    return 0;
}
```

23.Find the count of repeated element in array

```
#include <stdio.h>
int main() {
   int s, i, j;

   // Read the size of the array
scanf("%d", &s);

   int arr[s];

   // Read elements into the array
   for (i = 0; i < s; i++) {
scanf("%d", &arr[i]);
   }
   int found_repeated = 0;  // Variable to check if any repeated elements were found
   // Find and print repeated elements
   for (i = 0; i < s; i++) {
      int count = 1;
      if (arr[i] != -1) {
         for (j = i + 1; j < s; j++) {
            if (arr[i] == arr[j]) {
               count++;
               arr[j] = -1;  // Mark this element as counted
            }
         }
         if (count > 1) {
printf("%d occurs %d times\n", arr[i], count);
            found_repeated = 1;  // Mark that at least one repeated element was found
         }
      }
   }
   if (found_repeated==0) {
printf("no repeated elements\n");
   }
   return 0;
}
```

24.Sort the array in ascending order and print even numbers first and odd numbers next

```
#include<stdio.h>
int main(){
    int i,n,j,temp;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
for(i=0;i<n-1;i++){
    for(j=i+1;j<n;j++){
        if(arr[i]>arr[j]){
            temp=arr[i];
            arr[i]=arr[j];
            arr[j]=temp;
        }
    }
}
for(i=0;i<n;i++){
    if(arr[i]%2==0){
printf("%d ",arr[i]);
    }
}
for(i=0;i<n;i++){
    if(arr[i]%2!=0){
printf("%d ",arr[i]);
    }
}
}
```

25.Removing even numbers from an array

```
#include <stdio.h>
int main() {
    int i, n, index = 0;
printf("Enter the number of elements in the array: ");
scanf("%d", &n);
    int arr[n];
    // Read elements into the array
printf("Enter %d elements:\n", n);
    for (i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
    int narr[n];
    for (i = 0; i < n; i++) {
        if (arr[i] % 2 != 0) {
            narr[index] = arr[i];
            index++;
        }
    }
printf("Array after removing even numbers:\n");
    for (i = 0; i < index; i++) {
printf("%d ", narr[i]);
    }
    return 0;
}
```

(creating without new array)

```
#include <stdio.h>
int main() {
    int n, i,j;
printf("Enter the number of elements in the array: ");
scanf("%d", &n);
    int arr[n];
printf("Enter %d elements:", n);
for(i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
for(i=0;i<n;i++){
    if(arr[i]%2==0){
        for(j=i;j<n;j++){
            arr[j]=arr[j+1];
        }
        n--;
    }
}
for(i=0;i<n;i++){
```

```
printf("%d ",arr[i]);
}
```

26.Removing first occurrence of a given number from an array

```
#include <stdio.h>
int main() {
   int s, i, num_to_remove;
printf("Enter the size of the array: ");
scanf("%d", &s);
   int arr[s];
printf("Enter %d elements:\n", s);
   for (i = 0; i < s; i++) {
scanf("%d", &arr[i]);
   }
printf("Enter the number to remove: ");
scanf("%d", &num_to_remove);
   for (i = 0; i < s; i++) {
      if (arr[i] == num_to_remove) {
         for (int j = i; j < s - 1; j++) {
            arr[j] = arr[j + 1];
         }
s--;
         break;
      }
   }
printf("Updated array:\n");
   for (i = 0; i < s; i++) {
printf("%d ", arr[i]);
   }
   printf("\n");
   return 0;
}
```

Removing repeated elements first occurenbce

```
#include <stdio.h>

int main() {
   int n, i, j, k;
printf("Enter the number of elements in the array: ");
scanf("%d", &n);
    int arr[n];
printf("Enter %d elements: ", n);
for(i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
    // Removing first occurrence of any repeated elements
for(i = 0; i < n - 1; i++) {
for(j = i + 1; j < n; j++) {
            if(arr[i] == arr[j]) {
                // Shift elements to the left
for(k = i; k < n - 1; k++) {
                    arr[k] = arr[k + 1];
                }
                n--; // Reduce array size
                i--; // Adjust index after shifting
            }
        }
    }
    // Print the updated array
printf("Updated array: ");
for(i = 0; i < n; i++) {
printf("%d ", arr[i]);
    }
    printf("\n");
    return 0;
}
```

Product array without its self number

```
#include <stdio.h>
int main() {
    int n, i,j,k;
printf("Enter the number of elements in the array: ");
scanf("%d", &n);
    int arr[n];
printf("Enter %d elements:", n);
for(i = 0; i < n; i++) {
scanf("%d", &arr[i]);
```

```
    }
    int tp=1;
    for(i=0;i<n;i++){
        tp=tp*arr[i];
    }
    for(i=0;i<n;i++){
printf("%d ",tp/arr[i]);
    }
}
```

27.Array Chunking - Eg1 : Chunk size = 2, array size 8, elements (1,2,3,4,5,6,7,8) - Ans : [1,2] [3,4] [5,6] [7,8] Eg2: Chunk size = 3, array size 4, elements (1,2,3,4) - Ans : [1,2,3]

```
#include <stdio.h>
int main(){
    int i,j,n;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
    scanf("%d",&arr[i]);
    }
    int cs;
    scanf("%d",&cs);
    for(i=0;i<n;i+=cs){
printf("[");
for(j=i;j<i+cs && j < n;j++){
        printf("%d",arr[j]);
if(j<i+cs-1 && j<n-1){
printf(",");
        }
        }
printf("]");
    }
    printf("\n");
```

How Many Numbers Are Smaller Than the CurrentNumber
Input:nums=[8,1,2,2,3] Output:[4,0,1,1,3]

```
#include<stdio.h>
int main(){
    int n,i,j;
    scanf("%d",&n);
    int arr[n];
    int counts[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;i++){
         int count=0;
        for(j=0;j<n;j++){
            if(arr[j]<arr[i]){
                count++;
            }
        }
        counts[i]=count;
    }
    for (int i = 0; i < n; i++) {
printf("%d ", counts[i]);
    }
}
```

MaximumDistance Between Same Elements Given an array with duplicate elements, find the maximum distance between two occurrences of the same element.

```
#include <stdio.h>
#include<stdlib.h>

int main(){
    int n,i,j;
    scanf("%d",&n);
    int arr[n];
    for (i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    int d;
    scanf("%d",&d);
    for(i=0;i<n;i++){
        if(arr[i]==d){
            for(j=n-1;j>=0;j++
                if(arr[i]==arr[j]){
                    printf("%d",abs(i-j)-1);
                    break;
                }
            }
            break;
        }
    }
}
```

Ranks for marks

```
#include<stdio.h>
int main() {
    int i, n,j;
     int ranks[n];
scanf("%d", &n);
    int arr[n];
for(i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }

  for(i=0;i<n;i++){
     int rank=1;
     for(j=0;j<n;j++){
         if(arr[j]>arr[i]){
             rank++;
         }
     }
     ranks[i]=rank;
  }
  for(i=0;i<n;i++){
printf("%d ",ranks[i]);
  }
}
```

Leader element (that is…. num should be greater than of that number's right side)

```
#include <stdio.h>
int main() {
  int i,j,n;
  scanf("%d",&n);
  int arr[n];
  for(i=0;i<n;i++){
      scanf("%d",&arr[i]);
  }
int right=arr[n-1];
printf("%d ",right);

for(i=n-2;i>=0;i--){
    if(arr[i]>right){
printf("%d ",arr[i]);
          right=arr[i];
    }
}
}
```

Find the element in an array that are greater than all elements to their right and print them in ascending order sample input : 5 arr: 5 3 20 15 8 ouput: 8 15 20

```
#include <stdio.h>
int main() {
    int n;
printf("Enter the number of elements: ");
scanf("%d", &n);
    int arr[n];
printf("Enter the elements: ");
    for (int i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
    int max_right = arr[n - 1];
printf("%d ",max_right);
    for (int i = n - 2; i >= 0; i--) {
        if (arr[i] > max_right) {
printf("%d ", arr[i]);
            max_right = arr[i];
        }
    }
    printf("\n");
    return 0;
}
```

filtering the elements .get value by using scanf. compare each element from the array with value .if arr element greater than value ,delete that particular array element from the array and decrement the size of an array.

```
#include <stdio.h>
int main() {
    int i, j, n;
scanf("%d", &n);
    int arr[n];
    for (i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
int value;
scanf("%d",&value);
    for (i = 0; i < n; i ++) {
        if(arr[i]>value){
            for(j=i;j<n;j++){
                arr[j]=arr[j+1];
            }
            n--;
    I--;   // decrement the index to recheck the array
        }
}
for(i=0;i<n;i++){
printf("%d ",arr[i]);
}
}
```

Reverse an array

```
#include <stdio.h>
int main() {
    int i, j, n,temp;
scanf("%d", &n);
    int arr[n];
    for (i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
    for (int i = 0, j = n - 1; i < j; i++, j--) {
    temp=arr[i];
    arr[i]=arr[j];
    arr[j]=temp;
}
for(i=0;i<n;i++){
printf("%d ",arr[i]);
}
}
```

Assign Cookies

```
#include <stdio.h>
int main() {
    int g[] = {1, 2, 3};
    int s[] = {1, 1};
    int gSize = sizeof(g) / sizeof(g[0]);
    int sSize = sizeof(s) / sizeof(s[0]);
    // Sort both arrays
    for (int i = 0; i < gSize - 1; i++) {
        for (int j = i + 1; j < gSize; j++) {
            if (g[i] > g[j]) {
                int temp = g[i];
                g[i] = g[j];
                g[j] = temp;
            }
        }
    }
    for (int i = 0; i < sSize - 1; i++) {
        for (int j = i + 1; j < sSize; j++) {
            if (s[i] > s[j]) {
                int temp = s[i];
                s[i] = s[j];
                s[j] = temp;
            }
        }
    }
    int count = 0;
    int i = 0, j = 0;
    // Iterate over both arrays
    while (i < gSize && j < sSize) {
        // If the cookie size is greater than or equal to the greed factor, assign the cookie
        if (s[j] >= g[i]) {
            count++;
            i++;
        }
        j++;
    }
printf("The number of content children is: %d\n", count);
    return 0;
}
```

No of handshakes using nC2 problem

```
#include <stdio.h>
int main() {
int n,i;
scanf("%d",&n);
    int handshakes = n * (n - 1) / 2;
printf("Number of handshakes: %d\n", handshakes);
    return 0;
}
```

Without nC2 problem

```
#include <stdio.h>
int main() {
    int n = 5; // Number of people
    int count = 0;
    // Calculate nC2
    for (int i = 0; i < n - 1; i++) {
        for (int j = i + 1; j < n; j++) {
            count++;
        }
    }
printf("The number of handshakes is: %d\n", count);
    return 0;
}
```

LAST REPEATED ELEMENT

```
#include <stdio.h>
int main() {
   int arr[]={10,20,30,50,50,30};
   int l = sizeof(arr)/sizeof(arr[0]);
   int repeat;
   int count=0;
for(int i=0;i<l;i++){
for(int j=l-1;j>=0;j--){
         if(arr[i]==arr[j] &&i!=j){
            repeat =  arr[i];
            count++;
         }
      }
   }
if(count == 0){
printf("No repeated elements found : ");
}else{
      printf("%d",repeat);
   }
   return 0;
}
```

Get two user array , the sum of the first array should be less than the sum of the second array, add the elements of the second array, and print the min count of the number of elements that can be added to get the sum of the first array.

```
#include<stdio.h>
int main(){
   int i,j,n1,n2,temp;
   scanf("%d",&n1);
   scanf("%d",&n2);
   int arr1[n1];
   for(i=0;i<n1;i++){
   scanf("%d",&arr1[i]);
   }
   int arr2[n2];
```

```
    for(i=0;i<n2;i++){
        scanf("%d",&arr2[i]);
    }
    int sum1=0;
    int sum2=0;
    for(i=0;i<n1;i++){
        sum1+=arr1[i];
    }
    for(i=0;i<n2;i++){
        sum2+=arr2[i];
    }
    if(sum1>sum2){
printf("2 should be greater than 1");
    }
    for(i=0;i<n2;i++){
        for(j=i+1;j<n2;j++){
            if(arr2[i]<arr2[j]){
              temp=arr2[i];
              arr2[i]=arr2[j];
              arr2[j]=temp;
            }
        }
    }
    int mincount=0;
    int rsum=0;
    for(i=0;i<n2;i++){
        rsum+=arr2[i];
        mincount++;
    if(rsum>=sum1){
        break;
    }
    }
printf("%d ",mincount);

   }
```

Even place even num remover

```
#include<stdio.h>
int main(){
    int i,j,n;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
    scanf("%d",&arr[i]);
    }
    for(i=2;i<n;i+=2){
        if(arr[i]%2==0){
            for(j=i;j<n;j++){
                arr[j]=arr[j+1];
            }
            n--;
  I--; // decrement index to recheck the array
        }
    }
for(i=0;i<n;i++){
printf("%d ",arr[i]);
}

}
```

Non prime num removing from array

```
#include <stdio.h>
int main() {
    int arr[] = {2,3,4,5};
    int n = sizeof(arr) / sizeof(arr[0]);
    for (int i = 0; i < n - 1; i++) {
        for (int j = 2; j < arr[i]; j++) {
            if (arr[i] % j == 0) {
                for(j=i;j<n;j++){
                arr[j] = arr[j+1];
                }
                n--;
    i--; // decrement index to recheck the array
                break; }
        }
    }
printf("Prime numbers: ");
    for (int i = 0; i < n; i++) {
```

```
printf("%d ", arr[i]);
    }
    printf("\n");

    return 0;
}
```

Sum of major diagonal in matrix

```
#include <stdio.h>
int main() {
   int i,j,n,sum=0;
   scanf("%d",&n);
   int major[n][n];
   for(i=0;i<n;i++){
       for(j=0;j<n;j++){
           scanf("%d",&major[i][j]);
       }
   }
   for(i=0;i<n;i++){
       sum+=major[i][i];
   }
printf("sum is %d ",sum);
}
```

Sum of minor diagonal in matrix

```
#include <stdio.h>
int main() {
 int i,j,n,sum=0;
 scanf("%d",&n);
 int minor[n][n];
 for(i=0;i<n;i++){
     for(j=0;j<n;j++){
         scanf("%d",&minor[i][j]);
     }
 }
 for(i=0;i<n;i++){
     for(j=0;j<n;j++){
         if(i+j==n-1){
             sum+=minor[i][j];
         }
     }
 }
 printf("%d",sum);
}
```

Transpose of matrix

```
#include <stdio.h>
int main() {
 int i,j,n,sum=0;
 scanf("%d",&n);
 int trans[n][n];
 for(i=0;i<n;i++){
     for(j=0;j<n;j++){
         scanf("%d",&trans[i][j]);
     }
 }
 for(i=0;i<n;i++){
     for(j=0;j<n;j++){
printf("%d ",trans[j][i]);
     }
     printf("\n");
 }
}
```

Identity matrix

```
#include <stdio.h>
int main() {
 int i,j,n,sum=0;
 scanf("%d",&n);
 int identity[n][n];
 for(i=0;i<n;i++){
     for(j=0;j<n;j++){
          identity[i][j]=(i==j) ? 1 :0;
     }
 }
 for(i=0;i<n;i++){
     for(j=0;j<n;j++){
printf("%d ",identity[i][j]);
     }
     printf("\n");
 }
}
```

Sum two matrix

```
#include <stdio.h>
int main() {
    int n;
    scanf("%d",&n);
    int a[n][n];
    int b[n][n];
    int c[n][n];
for(int i=0;i<n;i++){
for(int j=0;j<n;j++){
            scanf("%d",&a[i][j]);
        }
    }
for(int i=0;i<n;i++){
for(int j=0;j<n;j++){
            scanf("%d",&b[i][j]);
        }
    }
for(int i=0;i<n;i++){
for(int j=0;j<n;j++){
            c[i][j] = a[i][j] + b[i][j];
        }
    }
for(int i=0;i<n;i++){
for(int j=0;j<n;j++){
printf("%d ",c[i][j]);
        }
        printf("\n");
    }
}
```

Multiple of 2 matrix

```
#include <stdio.h>
#define SIZE 3 // Size of the matrix
int main()
{
    int A[SIZE][SIZE]; // Matrix 1
    int B[SIZE][SIZE]; // Matrix 2
    int C[SIZE][SIZE]; // Resultant matrix
    int row, col, i, sum;


    /* Input elements in first matrix from user */
printf("Enter elements in matrix A of size %dx%d: \n", SIZE, SIZE);
for(row=0; row<SIZE; row++)
    {
for(col=0; col<SIZE; col++)
        {
scanf("%d", &A[row][col]);
        }
    }
printf("\nEnter elements in matrix B of size %dx%d: \n", SIZE, SIZE);
for(row=0; row<SIZE; row++)
    {
for(col=0; col<SIZE; col++)
        {
scanf("%d", &B[row][col]);
        }
    }
    /*
     * Multiply both matrices A*B
     */
for(row=0; row<SIZE; row++)
    {
for(col=0; col<SIZE; col++)
        {
            sum = 0;
            /*
             * Multiply row of first matrix to column of second matrix
             * and store sum of product of elements in sum.
             */
for(i=0; i<SIZE; i++)
            {
                sum += A[row][i] * B[i][col];
            }

            C[row][col] = sum;
        }
    }

    /* Print product of the matrices */
printf("\nProduct of matrix A * B = \n");
for(row=0; row<SIZE; row++)
    {
for(col=0; col<SIZE; col++)
        {
printf("%d ", C[row][col]);
        }
        printf("\n");
    }
    return 0;
}
```

Given matrix is a symmetric matrix or not.

```
#include <stdio.h>
int main() {
    int i, j, n;
scanf("%d", &n);
    int a[n][n];
for(i = 0; i < n; i++) {
for(j = 0; j < n; j++) {
scanf("%d", &a[i][j]);
        }
    }
    int symmetric = 1;
for(i = 0; i < n; i++) {
for(j = 0; j < n; j++) {
            if(a[i][j] != a[j][i]) {
                symmetric = 0;
            }
        }
    }
printf("The matrix is:\n");
for(i = 0; i < n; i++) {
for(j = 0; j < n; j++) {
printf("%d ", a[i][j]);
        }
        printf("\n");
    }
    if(symmetric) {
printf("Is symmetric.\n");
    } else {
        printf("Not.\n");
    }
    return 0;
}
```

Arrange zeros at last in array

```
#include <stdio.h>
int main() {
    int i,n,j,count=0;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    int res[n];
    for(i=0;i<n;i++){
        if(arr[i]!=0){
            res[count]=arr[i];
            count++;
        }
    }
    for(i=count;i<n;i++){
        res[i]=0;
    }
    for(i=0;i<n;i++){
printf("%d ",res[i]);
    }
    return 0;
}
```

Merge two array to third array (O(n)):

```
// Online C compiler to run C program online
#include <stdio.h>

int main() {
    // Write C code here
    int i,j,k,n1,n2,n3,temp;
    scanf("%d",&n1);
    scanf("%d",&n2);
    int arr1[n1],arr2[n2];
    for(i=0;i<n1;i++){
        scanf("%d",&arr1[i]);
    }
    for(j=0;j<n2;j++){
        scanf("%d",&arr2[j]);
    }
    n3 = n1+n2;
    int res[n3];
    i=0;
    j=0;
    while(i<n1&&j<n2){
        if(arr1[i]<=arr2[j]){
            res[k] = arr1[i];
            i++;
        }else{
            res[k] = arr2[j];
            j++;
        }
        k++;
    }
    while(i<n1){
        res[k] = arr1[i];
        i++;
        k++;
    }
    while(j<n2){
        res[k] = arr2[j];
        j++;
        k++;
    }
    for(k=0;k<n3;k++){
        printf("%d ",res[k]);
    }
    printf("\n");
    return 0;
}
```

Find a pair with a given sum

```
#include <stdio.h>
int main() {
    int n,i,j;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    int temp =0;
    for(i=0;i<n;i++){
        for(j=i+1;j<n;j++){
            if(arr[i]>arr[j]){
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
    for(i=0;i<n;i++){
printf("%d \n",arr[i]); // sorted array
    }
    int Gs;
    scanf("%d",&Gs);
    int l=0; // index from starting
    int r = n-1; // index from last
    while(l<r){
        if(arr[l]+arr[r]>Gs){
            r--;
        }
        else if(arr[l]+arr[r]<Gs){
            l++;
        }
else{  // else it must the correct apir with the given sum value
            printf("(%d,%d)",arr[l],arr[r]);
            l++;
            r--;
        }
    }
    return 0;
```

Sum of individual rows and coulumns:

```
#include <stdio.h>
int main() {
    int n,i,j,SumRow,SumCol;
    scanf("%d",&n);
    int arr[n][n];
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            scanf("%d",&arr[i][j]);
        }
    }
    for(i=0;i<n;i++){
        SumRow=0;
        SumCol =0;
        for(j=0;j<n;j++){
            SumRow=SumRow+arr[i][j];
            SumCol = SumCol+arr[j][i];
            }
printf("SumRow = %d ,SumCol = %d\n",SumRow,SumCol);
    }
    return 0;
}
```

Arrange zeros after each element:

```
#include<stdio.h>
int main(){
  int n ;
  scanf("%d\n" , &n) ;
  int arr[n] ;
  for(int i = 0 ; i < n ; i++){
    scanf("%d\n" , &arr[i]);
  }
  int res[n] ;
  int zero[n] ;
  int nonzero[n] ;
  int zero_count = 0;
  int nonzero_count = 0 ;
  int j = 0 ;
  int k = 0 ;
  for(int i = 0 ; i < n ; i++){
    if(arr[i] == 0){
      zero[j++] = arr[i] ;
      zero_count++;
    }
    if(arr[i] != 0){
      nonzero[k++] = arr[i] ;
      nonzero_count++;
    }
  }
  int m = 0 ;
  for(int i = 0 ; i < n ; i++){
    res[m++] = nonzero[i];
    if(i <zero_count){
      res[m++] = zero[i] ;
    }
  }
  for(int i = 0 ; i < n ; i++){
    printf("%d\n" , res[i]);
  }
}
```

4**. Maximum and minimum element.**

```
#include <stdio.h>
int main() {
    int i, j, n, temp;
printf("Enter the number of elements: ");
scanf("%d", &n);
    int arr[n];
printf("Enter %d integers:\n", n);
    for (i = 0; i < n; i++) {
scanf("%d", &arr[i]);
```

```
    }
    for (i = 0; i < n - 1; i++) {
        for (j = 0; j < n - i - 1; j++) {
            if (arr[j] >arr[j + 1]) {
                temp = arr[j];
                arr[j] = arr[j + 1];
arr[j + 1] = temp;
            }
        }
    }
printf("Minimum Number is %d\n", arr[0]);
printf("Maximum Number is %d\n", arr[n - 1]);
    return 0;
}
```

5. Find pair with given Sum:

```
#include <stdio.h>
int main() {
    int i, j, n, num;
printf("Enter the number of elements: ");
scanf("%d", &n);
    int arr[n];
printf("Enter %d integers:\n", n);
  for (i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
printf("Enter the sum to find pairs: ");
scanf("%d", &num);
    int found = 0;
    for (i = 0; i < n; i++) {
        for (j = i + 1; j < n; j++) {
            if (arr[i] + arr[j] == num) {
printf("Index %d and Index %d, Values %d and %d\n", i, j, arr[i], arr[j]);
                found = 1;
            }
        }
    }

    if (!found) {
printf("No pairs found with the given sum.\n");
    }
    return 0;
}
```

6. Delete the Element from Array

```
#include <stdio.h>
int main() {
    int i, n, posToDlt;
printf("Enter the number of elements: ");
scanf("%d", &n);
    if (n <= 0) {
printf("Invalid size of array.\n");
        return 0;
    }
    int arr[n];
printf("Enter %d integers:\n", n);
    for (i = 0; i < n; i++) {
scanf("%d", &arr[i]);
    }
printf("Enter the position to delete (1 to %d): ", n);
scanf("%d", &posToDlt);
    if (posToDlt < 1 || posToDlt > n) {
printf("Invalid Position.\n");
        return 0;
    }
    for (i = posToDlt - 1; i < n - 1; i++) {
        arr[i] = arr[i + 1];
    }
    n--; // Decrease the size of the array

    // Output the updated array
printf("Array after deleting element at position %d:\n", posToDlt);
    for (i = 0; i < n; i++) {
printf("%d\n", arr[i]);
    }
    return 0;
}
```

Array rotation for k times(left or right)

```
#include<stdio.h>
int main() {
    int a[10],i,j,k,n,temp;
    scanf("%d",&n);
printf("enter elements:");
    for(i=0;i<n;i++){
scanf("%d ",&a[i]);
}
        scanf("%d",&k);
    for(j=1;j<=k;j++){
    temp=a[0];
    for(i=0;i<n-1;i++){
        a[i]=a[i+1];
    }
    a[i]=temp;
}
    for(i=0;i<n;i++){
printf("%d ",a[i]);

    }
}
```

give c program to print the number of occurance of a number in an array

```
#include<stdio.h>
int main(){
    int s,i,j;
    scanf("%d",&s);
    int arr[s];
for(int i=0;i<s;i++){
        scanf("%d",&arr[i]);
    }
    for(i=0;i<s;i++){
        int count=1;
        if(arr[i]!=-1){
            for(j=i+1;j<s;j++){
                if(arr[i]==arr[j]){
                    count++;
                    arr[j]=-1;
            }
}
printf("%d ocuurs %d times\n",arr[i],count );

                }
    }
}
```

\\\
\\

Decimal value of maximum number of consecutive 1’s in an array:

```
#include <stdio.h>
int main() {
    int size;
printf("Enter the size of the array: ");
scanf("%d", &size);
    int arr[size];
printf("Enter the elements (0s and 1s only): ");
    for (int i = 0; i < size; i++) {
scanf("%d", &arr[i]);
    }
    int maxCount = 0;
    int currentCount = 0;
    for (int i = 0; i < size; i++) {
        if (arr[i] == 1) {
            currentCount++;
        } else {
            if (currentCount > maxCount) {
                maxCount = currentCount;
            }
            currentCount = 0;
        }
    }
    if (currentCount > maxCount) {
        maxCount = currentCount;
    }
printf("Maximum consecutive 1's in the array: %d\n", maxCount);
    return 0;
}
```

Majority Element in an array

```
// Online C compiler to run C program online
#include <stdio.h>
int main() {
    // Write C code here
    int i,k,j,n;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    int maxcount =0;
    int index =-1;
    for(i=0;i<n;i++){
        int count = 1;
        for(j=i+1;j<n;j++){
            if(arr[i] == arr[j]){
                count++;
            }
        }
            if(count>maxcount){
                maxcount =count;
                index =i;
            }
        }
        if(maxcount>n/2){
printf("%d is majority element:",arr[index]);
        }
else{
printf("No Majority Element");
        }
    return 0;
}
```

Program to calculate determinant of 2×2 matrix

```
#include <stdio.h>
int main() {
    // Write C code here
    int i,j,n,deter;
    scanf("%d",&n);
    int arr[n][n];
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            scanf("%d",&arr[i][j]);
        }
    }
    deter=(arr[0][0]*arr[1][1])-(arr[0][1]*arr[1][0]);
    printf("%d",deter);
    return 0;
}
```

Program to calculate determinant of 3×3 matrix

[a b c
 d e f
 g h i]
\\\
Deter = A(ei-hf) – b(di-gf) – c(dh-ge);
\\

Program to interchange diagonal elements of a matrix

```
#include <stdio.h>
int main() {
    // Write C code here
    int i,j,n,deter;
    scanf("%d",&n);
    int a[n][n];
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            scanf("%d",&a[i][j]);
        }
    }
    int max_i =3;
    int max_j = 3;
    n = (max_i < max_j) ? max_i : max_j;
    for(i=0;i<n;i++){
        j = i;
        int temp = a[i][j];
        a[i][j] = a[i][(n-j)-1];
        a[i][(n-j)-1] = temp;
    }
printf("After interchanging:\n");
    for(i=0;i<max_i;i++){
        for(j=0;j<max_j;j++){
printf("%d ",a[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

Program to check sparse matrix

```
#include <stdio.h>
int main() {
    // Write C code here
    int i,j,n,total =0;
    scanf("%d",&n);
    int arr[n][n];
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            scanf("%d",&arr[i][j]);
        }
    }
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            if(arr[i][j]==0){
                total++;
            }
        }
    }
    if(total>(i*j)/2){
```

```
printf("\n Is a sparse matrix");
}else{
        printf("\nNot");
    }
    return 0;
}
```

Print positive elements first and negative elements next and no space after the last element:

```
#include <stdio.h>
int main() {
    int i,j,n;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;i++){
        if(arr[i]>0){
            printf("%d",arr[i]);
if(i != n-1){
printf(" ");
            }
        }
    }
      for(i=0;i<n;i++){
        if(arr[i]<0){
            printf("%d",arr[i]);
if(i != n-1){
printf(" ");
            }
        }
    }
    printf("\n");
    return 0;
}
```

Sort the first half of the array elements in ascending order and second half of the array in descending order:

```
#include <stdio.h>
int main() {
    int i,j,n,temp;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
        for(i=0;i<n/2;i++){
            for(j=i+1;j<n/2;j++){
                if(arr[i]>arr[j]){
                    temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
        }
        for(i=n/2;i<n;i++){
            for(j=i+1;j<n;j++){
                if(arr[i]<arr[j]){
                    temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
        }
    for(i=0;i<n;i++){
        printf("%d\n",arr[i]);
    }
    return 0;
}
```

Print the square of duplicate elements otherwise print as it is:

```
#include <stdio.h>
int main() {
    int i,j,n,count;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;i++){
        count = 0;
        for(j=0;j<n;j++){
            if(arr[i]==arr[j]&&i!=j){
                count++;
            }
        }
        if(count!=0){
            printf("%d ",arr[i]*arr[i]);
        }
        else{
            printf("%d ",arr[i]);
        }
    }
    return 0;
}
```

Code to sort first half in asc and second half in dec and in first half if n is even print even first and odd next and if n is odd print odd first and even next:

```
#include <stdio.h>
int main() {
    int n, arr[100], temp;
    scanf("%d", &n);

    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    // Sort the first half of the array in ascending order
    for (int i = 0; i < n / 2; i++) {
        for (int j = i + 1; j < n / 2; j++) {
            if (arr[i] > arr[j]) {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
    // Sort the second half of the array in descending order
    for (int i = n / 2; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            if (arr[i] < arr[j]) {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
    // Print the first half of the array
    if ((n / 2) % 2 == 0) {
        // Print even numbers first, then odd numbers
        for (int i = 0; i < n / 2; i++) {
            if (arr[i] % 2 == 0) {
                printf("%d ", arr[i]);
            }
        }
        for (int i = 0; i < n / 2; i++) {
            if (arr[i] % 2 != 0) {
                printf("%d ", arr[i]);
            }
        }
    } else {
        // Print odd numbers first, then even numbers
        for (int i = 0; i < n / 2; i++) {
            if (arr[i] % 2 != 0) {
                printf("%d ", arr[i]);
            }
        }
        for (int i = 0; i < n / 2; i++) {
            if (arr[i] % 2 == 0) {
                printf("%d ", arr[i]);
            }
        }
    }
    // Print the second half of the array
    for (int i = n / 2; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

Printing unique elements or deleting duplicates (using frequency)

```
#include <stdio.h>
int main() {
    int n,i,j,sum=0;
    scanf("%d", &n);
    int arr[n],freq[n];
    for(i=0;i<n;i++) {
        scanf("%d", &arr[i]);
        freq[i]=1;
    }
    for(i=0;i<n;i++) {
        int count=1;
        for(j=i+1;j<n;j++) {
            if(arr[i]==arr[j]) {
                count++;
                freq[j]=0;
            }
        }
            if(freq[i]!=0) {
               freq[i]=count;
            }
       }
    for(i=0;i<n;i++) {
        if(freq[i]!=0||freq[i]==1) {
            printf("%d ",arr[i]);

        }
    }
}
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
Freq[i]!=0 \\\ remves duplicates
Freq[i]==1\\\\unique elements
Freq[i]==0\\\\duplicates print
(freq[i]!=0||freq[i]==1)  \\\remove dup and uniq (print both)
```