PROBLEM 1 (INSERTION IN ARRAY)

```
#include<stdio.h>
int main(int argc, char const *argv[])
{
   int n;
   int arr[n+1];
   int i;
   int pos;
   printf("Enter the number of the elements:");
   scanf("%d",&n);
   for (int i = 0; i < n; ++i)
   {
      scanf("%d",&arr[i]);
   }
   for(i=0;i<n;++i)
   {
      printf("\n the array elements are: %d",arr[i]);
   }
   printf("\n enter the position to be entered:");
   scanf("%d",&pos);
   int new;
   printf("\n Enter the element to be enteres:");
   scanf("%d",&new);
   for (i = n-1; i >= pos; --i)
   {
      arr[i+1] = arr[i];
   }
   arr[pos] = new;
   for (i = 0; i <=n; i++)
   {
      printf("%d ", arr[i]);
   }
   printf("\n");
   return 0;
}
```

OUTPUT:

```
Enter the number of the elements:5
1
2
3
4
5

 the array elements are: 1
 the array elements are: 2
 the array elements are: 3
 the array elements are: 4
 the array elements are: 5
 enter the position to be entered:4

 Enter the element to be enteres:45
1 2 3 4 45 5
```

PROBLEM 2 (DELETION OF ARRAY)

```
#include<stdio.h>
int main(int argc, char const *argv[])
{
   int n;
   int arr[n];
   int i;
   int pos;
   printf("Enter the number of the elements:");
   scanf("%d",&n);
   for (int i = 0; i < n; ++i)
   {
      scanf("%d",&arr[i]);
   }
   for(i=0;i<n;++i)
   {
      printf("\n the array elements are: %d",arr[i]);
   }
   printf("\n enter the position to be deleted:");
   scanf("%d",&pos);
   if ( pos >= n+1 )
   {
      printf("Deletion not possible.\n");
   }
   else
   {
      for ( i = pos - 1 ; i < n - 1 ; i++ )
      {
         arr[i] = arr[i+1];
      }
      printf("Resultant array is\n");
      for( i = 0 ; i < n - 1 ; i++ )
      {
         printf("%d\n", arr[i]);
      }

   }
   return 0;
}
```

OUTPUT:

```
Enter the number of the elements:5
1
2
3
4
5

 the array elements are: 1
 the array elements are: 2
 the array elements are: 3
 the array elements are: 4
 the array elements are: 5
 enter the position to be deleted:5
Resultant array is
1
2
3
4
```

PROBLEM 3 [ARRAY ROTATION]

```
#include<stdio.h>
int main(int argc, char const *argv[])
{
    int n;
    int arr[n];
    int i;
    int pos;
    printf("Enter the number of the elements:");
    scanf("%d",&n);
    for (int i = 0; i < n; ++i)
    {
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;++i)
    {
        printf("\n the array elements are: %d",arr[i]);
    }
    printf("\n enter the rotation to be done:");
    int rotation;
    scanf("%d",&rotation);
    for(int i = 0; i < rotation; i++){
        int j, first;
        //Stores the first element of the array
        first = arr[0];

        for(j = 0; j < n-1; j++){
           //Shift element of array by one
           arr[j] = arr[j+1];
        }
        //First element of array will be added to the end
        arr[j] = first;
    }

    printf("\n");

    //Displays resulting array after rotation
    printf("Array after left rotation: \n");
    for(int i = 0; i < n; i++){
        printf("%d ", arr[i]);
    }
    return 0;
}
```

OUTPUT:

```
Enter the number of the elements:5
1
2
3
4
5

 the array elements are: 1
 the array elements are: 2
 the array elements are: 3
 the array elements are: 4
 the array elements are: 5
 enter the rotation to be done:2

Array after left rotation:
3 4 5 1 2
```

PROBLEM 4(REMOVAL OF DUPLICATION)

```
#include<stdio.h>
int main(int argc, char const *argv[])
{
    int n;
    int arr[n];
    int i;
    int pos;
    printf("Enter the number of the elements:");
    scanf("%d",&n);
    for (int i = 0; i < n; ++i)
    {
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;++i)
    {
        printf("\n the array elements are: %d",arr[i]);
    }
    // use nested for loop to find the duplicate elements in array
    for ( int i = 0; i < n; i ++)
    {
        for ( int j = i + 1; j < n; j++)
        {
            // use if statement to check duplicate element
            if ( arr[i] == arr[j])
            {
                // delete the current position of the duplicate element
                for ( int k = j; k < n - 1; k++)
                {
                    arr[k] = arr [k + 1];
                }
                // decrease the size of array after removing duplicate element
                n--;

            // if the position of the elements is changes, don't increase the index j
                j--;
            }
        }
    }


    /* display an array after deletion or removing of the duplicate elements */
    printf (" \n Array elements after deletion of the duplicate elements: ");

    // for loop to print the array
    for ( i = 0; i < n; i++)
    {

        printf (" %d \t", arr[i]);
    }
    return 0;
}
```

OUTPUT:

```
Enter the number of the elements:5
1
1
2
3
4

 the array elements are: 1
 the array elements are: 1
 the array elements are: 2
 the array elements are: 3
 Array elements after deletion of the duplicate elements:  1    2     3
```

STRNGS

COLLECTION OF CHARACTER OR CHARACTER ARRAY IT ENCLOSED WITHIN DOUBLE QUOTES

STRING FUNCTIONS:

1. STRLEN ()
2. STRCAT ()
3. STRCPY ()
4. STRCMP ()
5. STRREV ()
6. STRLWR ()
7. STRUPR ()