













Source :

#include<stdio.h>

#include<string.h>

struct Numbers{

    char Num[100];

};

int count = 0;

void AddNum(int \*n,Numbers num[100]){

    // printf("\nNumber of Numberss add: ");

    // scanf("%d",n);

    printf("Enter new Numbers Num: ");

        fflush(stdin);

        gets(num[count].Num);

        count ++;

    // for(int i=0;i<\*n;i++){

    // }

    printf ("Numbers has been added to list successfully!\n");

    printf("New List: ");

    for (int i = 0; i < count; i++)

    {

        printf("%s ",num[i].Num);

    }

    printf("\n");

}

void RemoveNum(int \*n,Numbers num[100]){

    char So[100];

    printf ("\nEnter Number Num to remove: ");

    fflush(stdin);

    gets(So);

    for(int i=0;i<count;i++){

        if(strcmp(num[i].Num,So)==0){

            num[i] = num[i+1];

        for (int k = i+1; k < count; k++)

        {

            num[k] = num[k+1];

        }

         printf("Number has been Remove successfully!\n");

    printf("New List: ");

    for (int i = 0; i < count-1 ; i++)

    {

        printf("%s ",num[i].Num);

    }

    printf("\n");

            break;

        }else

        {

            printf("Number does not exist in list\n");

        }

    }

}

void PrintArr(int \*n,Numbers num[100]){

printf("List is: ");

    for (int i = 0; i < count ; i++)

    {

        printf("%s ",num[i].Num);

    }

}

void RemoveAllNum(int \*n,Numbers num[100]){

    char So[100];

    printf ("\nEnter Number Num to remove: ");

    fflush(stdin);

    gets(So);

    for(int i=0;i<count;i++){

        if(strcmp(num[i].Num,So)==0){

            num[i] = num[i+1];

        for (int k = i+1; k < count; k++)

        {

            num[k] = num[k+1];

        }

        }

    }

    printf("Number has been Remove successfully!\n");

    printf("New List: ");

    for (int i = 0; i < count-1 ; i++)

    {

        printf("%s ",num[i].Num);

    }

    printf("\n");

}

void SearchNum(int \*n,Numbers num[100]){

    int i=0;

    char Num[100];

    printf("\nEnter Num to search: ");

    fflush(stdin);

    gets(Num);

    for(i=0;i<count;i++){

        if(strcmp(num[i].Num,Num)==0){

            printf("Number Found\n");

            printf("The position of %s in list is: %d \n",Num,i+1);

            break;

        }

    }

    if(i==count){

        printf("Not Found !!!\n");

    }

}

void SortAs(int n,Numbers num[100]){

    printf("\nAscending order: ");

    for(int i=0;i<count;i++){

        if (count == 1)

        {

             printf("\n%d. %s",i+1,num[i].Num);

        }else

        {

         for(int j=i+1;j<count;j++){

           if(strcmp(num[i].Num,num[j].Num)>0){

                Numbers k=num[i];

                num[i]=num[j];

                num[j]=k;

            }

        }

        printf("\n%d.number %s ",i+1,num[i].Num);

        }

    }

}

void SortDes(int n,Numbers num[100]){

    printf("\nAlphabet sorted list: ");

    for(int i=0;i<count;i++){

        if (count == 1)

        {

             printf("\n%d. %s",i+1,num[i].Num);

        }else

        {

         for(int j=i+1;j<count;j++){

           if(strcmp(num[i].Num,num[j].Num)<0){

                Numbers k=num[i];

                num[i]=num[j];

                num[j]=k;

            }

        }

        printf("\n%d. %s",i+1,num[i].Num);

        }

    }

}

int main(){

    int choice;

    Numbers num[100];

    int n;

    while(1){

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

        printf("1.Add a value  \n");

        printf("2.Search a value  \n");

        printf("3.Remove the first existence of a value \n");

        printf("4.Remove all existences of a value\n");

        printf("5.Print out the array \n");

        printf("6.Sort the array in ascending order (positions of elements are preserved) \n");

        printf("7.Sort the array in descending order (positions of elements are preserved) \n");

        printf("Others. Quit\n\n");

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

        printf("\n\nEnter your choice: ");

        scanf("%d",&choice);

        switch(choice){

            case 1:

            if (count <= 100)

            {

                AddNum(&n,num);

            }else

            {

                printf("The list has full. It cannot add a new Numbers.");

            }

                break;

            case 2:

                SearchNum(&n,num);

                break;

            case 3:

                RemoveNum(&n,num);

                break;

            case 4:

                RemoveAllNum(&n,num);

                break;

            case 5:

                PrintArr(&n,num);

                break;

            case 6:

            SortAs(n, num);

                break;

            case 7:

                SortDes(n,num);

                break;

                default:

                return 0;

        }

    }

}