#include<stdio.h>

* Program to cyclically rotate an array by one

int prob\_1(){

    int n=5;

    int arr[5]={1,2,3,4,5};

    int arr\_1[n];

    for(int j=0;j<n;j++){

        arr\_1[j]=arr[j+1];

    }

    arr\_1[n-1]=arr[0];

    printf("list after rotation\n");

    for(int x=0;x<n;x++){

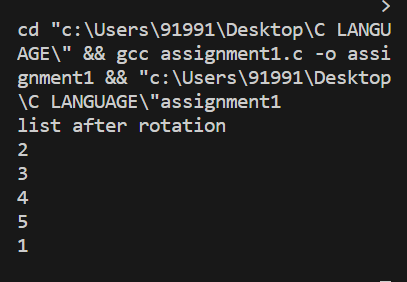
        printf("%d\n",arr\_1[x]);

    }

    return 0;

}

Output:



* Find the first repeating element in an array of integers

int prob\_2(int n){

    int arr[n];

    for(int i=0;i<n;i++){

        scanf("%d",&arr[i]);

    }

    printf("\n");

    printf("repeating elements\n");

    for(int j=0;j<n;j++){

        for(int k=j+1;k<n;k++){

            if(arr[j]==arr[k]){

                printf("%d",arr[k]);

                return 0;

            }

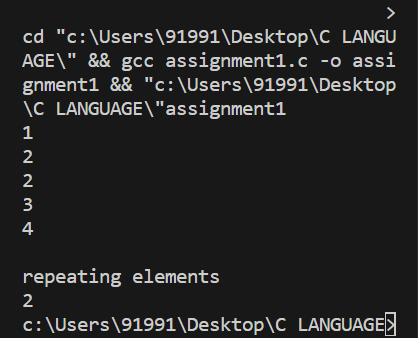
        }

    }

    printf("there is no repeating element");

}

Output



* Rearrange array in alternating positive & negative items

int prob\_3(int n){

    int arr[n];

    int negarr[n];

    int posarr[n];

    int negcount=0;

    int poscount=0;

    for(int i=0;i<n;i++){

        scanf("%d",&arr[i]);

    }

    printf("\n");

    printf("array after alternation\n");

    for(int j=0;j<n;j++){

        if(arr[j]<0){

            negarr[negcount++]=arr[j];

        }

        else{

            posarr[poscount++]=arr[j];

        }

    }

    negcount=0;

    poscount=0;

    for(int k=0;k<n;k++){

        if(k%2==0 && negcount<n){

            printf("%d\n",negarr[negcount++]);

        }

        else{

            printf("%d\n",posarr[poscount++]);

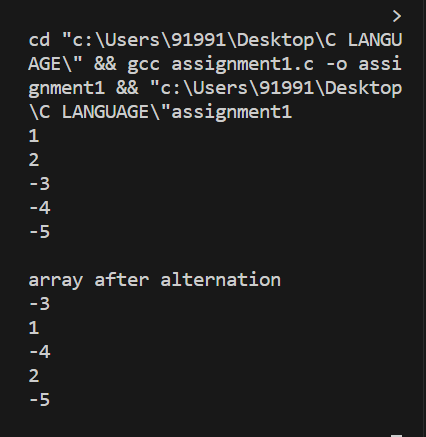
        }

    }

}

Output

\*\*this code only works when enough negative values are provided to fill up the even places, for example the below code\*\*\*



* Find if there is any subarray with a sum equal to zero

int prob\_4(int n){

    int arr[n];

    for(int i=0;i<n;i++){

        scanf("%d",&arr[i]);

    }

    int cumsum=0;

    int flag=0;

    for(int j=0;j<n;j++){

        cumsum+=arr[j];

        if(cumsum==0){

            flag=1;

            break;

        }

    }

if(flag){

    printf("there exists a substring whose sum is zero");

}

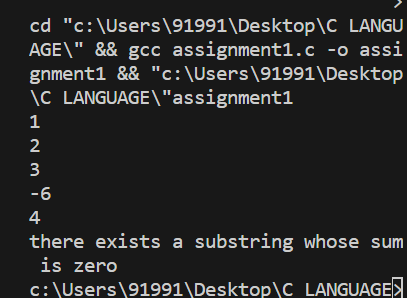
else{

    printf("there does not exists a substring whose sum is zero");

}

}

Output



* C Program To Find Maximum Difference Between Two Elements in an Array

int prob\_5(int n){

    int arr[n];

    for(int i=0;i<n;i++){

        scanf("%d",&arr[i]);

    }

    int max=arr[0];

    for(int j=0;j<n;j++){

        if(arr[j]>max){

            max=arr[j];

        }

    }

    int min=arr[0];

    for(int k=0;k<n;k++){

        if(arr[k]<min){

            min=arr[k];

        }

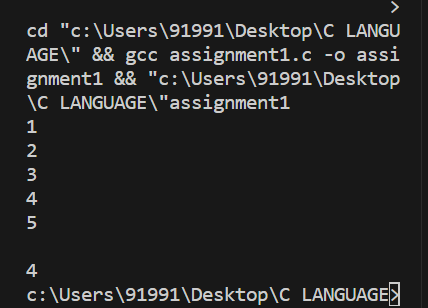
    }

    printf("\n");

    return max-min;

}

Output



* C Program to Count the Occurrences of Each Keyword using Array Structure

int prob\_6(){

  ///unattempted

}

* C Program to Find Union and Intersection of Two Arrays

int prob\_7(int n,int m){

    int arr\_1[n],arr\_2[m],un[m+n];

    for(int i=0;i<n;i++){

        scanf("%d",&arr\_1[i]);

    }

    printf("\n");

    for(int j=0;j<m;j++){

        scanf("%d",&arr\_2[j]);

    }

    printf("\n");

    printf("intersections\n");

    for(int a=0;a<n;a++){

        for(int b=0;b<m;b++){

            if(arr\_1[a]==arr\_2[b]){

                printf("%d ",arr\_1[a]);

            }

        }

    }

    printf("\n");

    printf("union\n");

    for(int i=0;i<n;i++){

        un[i]=arr\_1[i];

    }

    for(int j=0;j<m;j++){

        un[n+j]=arr\_2[j];

    }

    for(int c=0;c<(m+n);c++){

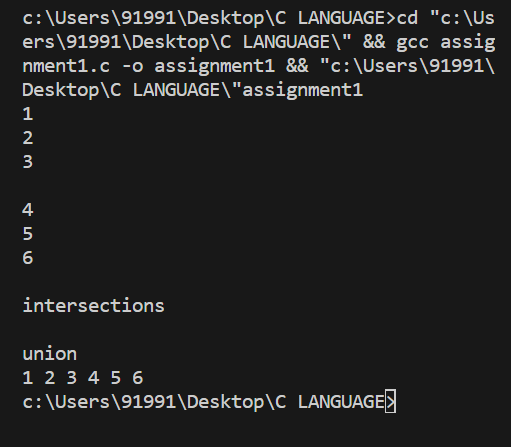
        printf("%d ",un[c]);

    }

}

Output

\*\*This code only works for arrays with same size\*\*



* Write a program in C to insert the values in the array (sorted list).

int prob\_8(int n){

    int temp;

    int i;

    int arr[7]={10,20,30,40,50};

    printf("initial list\n");

    for(int k=0;k<5;k++){

        printf("%d\n",arr[k]);

    }

    printf("\n");

    printf("final list\n");

    for( i=0;i<5;i++){

        if(arr[i]>n){

            temp=arr[i];

            arr[i]=n;

            break;

        }

    }

    if(i==5){

        arr[i]=n;

    }

    for(int j=6;j>(i+1);j--){

        arr[j]=arr[j-1];

    }

    arr[i+1]=temp;

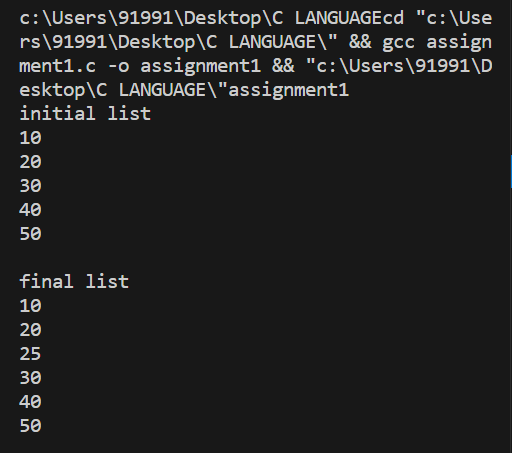
    for(int i=0;i<6;i++){

        printf("%d\n",arr[i]);

    }

}

Output



* Write a program in C to find the sum of rows and columns of a matrix

int prob\_9(int n, int m){

    int arr[n][m];

    for(int i=0;i<n;i++){

        for(int j=0;j<m;j++){

            scanf("%d\n",&arr[i][j]);

        }

        printf("\n");

    }

    printf("\n");

    printf("initial matrix\n");

    for(int k=0;k<n;k++){

        for(int l=0;l<m;l++){

            printf("%d ",arr[k][l]);

        }

        printf("\n");

    }

    printf("sum of rows\n");

    printf("\n");

    int rowsum=0;

        for(int a=0;a<n;a++){

            rowsum=0;

            for(int b=0;b<m;b++){

                rowsum+=arr[a][b];

            }

            printf("sum of %d row is: %d\n",a+1,rowsum);

        }

    printf("\n");

    printf("sum of columns");

    printf("\n");

    int columnsum=0;

    for(int c=0;c<m;c++){

        columnsum=0;

        for(int d=0;d<n;d++){

            columnsum+=arr[d][c];

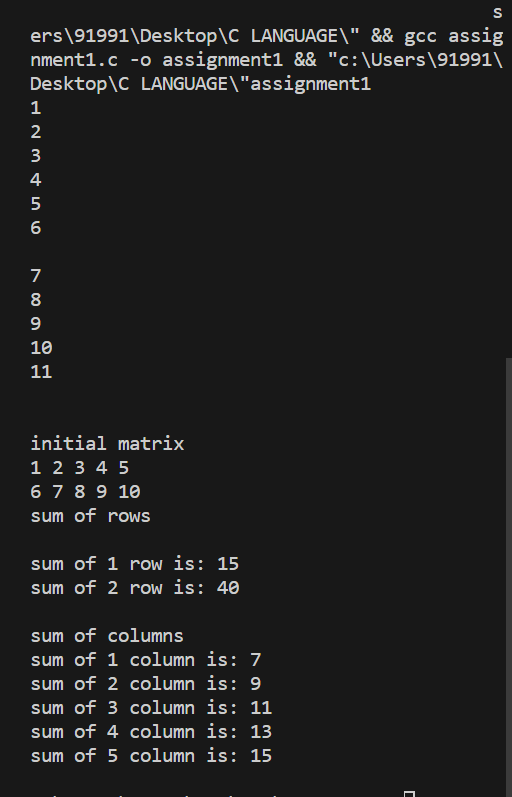
        }

        printf("sum of %d column is: %d\n",c+1,columnsum);

    }

}

Output



int main(){

}