**Name: Rohit Ghosh.**

**Roll Number: 21051420.**

**Date: 25/07/2022.**

**Class Assignment**

**Program 5**

**Question:**

Given an unsorted array arr [] and two numbers x and y, find the minimum distance between x and y in arr []. The array might also contain duplicates. You may assume that both x and y are different and present in arr [].

Input: arr [] = {3,5,4,2,6,5,6,6,5,4,8,3}, x=3, y=6

Output: Minimum distance between 3 and 6 is 4.

**Code:**

#include<stdio.h>

#include<stdlib.h>

typedef struct

{

int a,b,size,c;

}use;

//We made the structure

int main()

{

use\* e1=(use\*) malloc(sizeof(use\*));

printf("Enter the size of array\n");

scanf("%d",&e1->size);

int arr[e1->size];

//array created

e1->c=-1;

//this element stores the distance

printf("Enter the elements of the array\n");

for(int i=0;i<e1->size;i++)

{

scanf("%d",&arr[i]);

}

//got the elements of the array

int \*ptr=arr;

//created the pointer

printf("Enter the two elements\n");

scanf("%d",&e1->a);

scanf("%d",&e1->b);

//got the limits

printf("\nThe elements are: \n");

for(int i=0;i<e1->size;i++)

{

if(\*(ptr+i)==e1->a)

{

for(int j=i;j<=e1->size;j++)

{

e1->c++;

if(\*(ptr+j)==e1->b)

{

break;

}

}

}

}

//got the required distance

printf("Distance: %d",e1->c);

return 0;

}

**Output:**

