

CMR Engineering College

Name :Heman Babu
Roll no:228r1a62b1
Branch :CMREC
Batch :2022-26

Email :228r1a62b1@cmrec.ac.in
Phone :7989053640
Department :CS
Degree :B.Tech CS

2022-26_I_CS_B_C Programming Lab

C PROGRAMMING_AUTOMATA FIX_Single Dimensional Arrays

Attempt : 1
Total Mark : 100
Marks Obtained : 98

Section 1 : Automata Fix

1. Write a program to check if the given arrays have common elements or not.

Answer

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

int disjoint_arrays(int arr1[], int arr2[], int n, int m)
{
    for(int i=0; i<n; i++)
        for(int j=0; j<m; j++) if(arr1[i]==arr2[j]) return -1;
    return 0;
}

int main()
{
    int m,n;

    scanf("%d",&n);

    scanf("%d",&m);
```

```

int arr1[n];
int arr2[m];
int i;

for(i=0;i<n;i++)
{
    scanf("%d",&arr1[i]);
}

for(i=0;i<m;i++)
{
    scanf("%d",&arr2[i]);
}
int res = disjoint_arrays(arr1,arr2,n,m);
if(res == -1)
    printf("There is common elements");
else
    printf("There is no common elements");
return 0;
}

```

Status : Correct

Marks : 10/10

2. Write a program to find the Sum of absolute differences of the given Array.

For example, consider the following array given as input : arr = {1, 3, 9, 6, 3}

$$|1-3| + |3-9| + |9-6| + |6-3| = 2 + 6 + 3 + 3 = 14$$

Answer

```

// You are using GCC
#include<stdio.h>
#include<stdlib.h>
int abs_sum(int a[],int n){
    int i,sum=0,diff;
    for(i=0;i<(n-1);i++){
        diff=a[i]-a[i+1];
        sum=sum+(diff>0?diff:-diff);
    }
    return sum;
}

```

```

}

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

    printf("%d",abs_sum(a, n));
    return 0;
}

```

Status : Correct

Marks : 10/10

3. Write a Program to check if all the numbers of an array can be made equal.

Example:

Input:

3

50 75 100

Output:

Yes ->{50 * 2 * 3, 75 * 2 * 2, 100 * 3} = {300, 300, 300}

Answer

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

```

int make_equal(int a[], int n)
{
    for(int i=0; i<n; i++){
        int factors[]={2,3,5,7};
        for(int j=0;j<4;j++){

```

```

        while(a[i]%factors[j]==0) a[i]/=factors[j];
    }
    if(i!=0&& a[i]!=a[0]) return 0;
}
return 1;
}

int main()
{
    int n, i;
    scanf("%d", &n);
    int a[n];
    for(i=0; i<n; i++)
        scanf("%d", &a[i]);
    if (make_equal(a, n) == 1)
        printf("Yes");
    else
        printf("No");
    return 0;
}

```

Status : Partially correct

Marks : 8/10

4. Write a program to find repeating element in an array.

Answer

```

// You are using GCC
#include<stdio.h>
void repeating_element(int arr[], int n)
{
    int i, j;

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

```

```

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

```

Status : Correct

Marks : 10/10

5. Write a program to Find all Triplets with the given sum in the given array.

Answer

```

// You are using GCC
#include<stdio.h>

void find_all_triplets(int arr[], int n, int sum)
{
    for (int i = 0; i < n; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            for (int k = j + 1; k < n; k++)
            {
                if (arr[i] + arr[j] + arr[k] == sum)
                {
                    printf("%d,%d,%d\n",arr[i],arr[j],arr[k]);
                }
            }
        }
    }
}

```

```

    }

}

int main()
{
    int n;

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

    find_all_triplets(arr, n, sum);
    return 0;
}

```

Status : Correct

Marks : 10/10

6. Write a program to Rearrange positive and negative numbers in an array.

Example:

Input: -1 1 -2 2 -3 3

Output: -1 -2 -3 1 2 3

Answer

```

#include <stdio.h>
#include <stdlib.h>
void rearrange_alternate_positions(int arr[], int n)
{
    int temp[n];

```

```

int index=0;
for(int j=0; j<n; j++){
    if(arr[j]<1){
        temp[index]=arr[j];
        index++;
    }
}
for(int j=0; j<n; j++){
    if(arr[j]>0){
        temp[index]=arr[j];
        index++;
    }
}
for(int j=0; j<n; j++){
    arr[j]=temp[j];
}
}
int main()
{
    int n,i;

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

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

    rearrange_alternate_positions(arr, n);

    for(i = 0; i < n; i++)
    {
        printf("%d ",arr[i]);
    }

    return 0;
}

```

Status : Correct

Marks : 10/10

7. Write a program to check if two arrays are equal or not.

Answer

```
#include<stdio.h>

int sort(int arr[], int n)
{
    int i,j;
    for (i = 0; i < n-1; i++)
    {
        for (j = 0; j < n-i-1; j++)
        {
            if (arr[j] > arr[j+1])
            {
                int temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;
            }
        }
    }
}

int arrays_equal(int arr1[], int arr2[], int n, int m)
{
    sort(arr1, n);
    sort(arr2, m);
    if(n!=m) return 0;
    for(int i=0;i<n;i++){
        if(arr1[i]!=arr2[i])    return 0;
    }
    return 1;
}

int main()
{
    int n, m;
    scanf("%d",&n);
    scanf("%d",&m);
    int arr1[n];
    int arr2[m];
    int i;
    for(i = 0; i < n; i++)
```



```

{
    scanf("%d",&arr1[i]);
}
for(i = 0; i < m; i++)
{
    scanf("%d",&arr2[i]);
}
if(arrays_equal(arr1, arr2, n, m) == 0)
{
    printf("Not same");
}
else
    printf("Same");

return 0;
}

```

Status : Correct

Marks : 10/10

8. Write a program to find the frequency of each element of an array.

Answer

```

#include <stdio.h>

int main()
{
    int arr[100], freq[100];
    int size, i, j, count;
    scanf("%d", &size);

    for(i=0; i<size; i++){
        scanf("%d", &arr[i]);
        freq[i] = -1;
    }

    for(i=0; i<size; i++){
        if(freq[i]!=0){
            for(j=0; j<size; j++){
                if(arr[j]==arr[i]){
                    freq[j]=0;
                    freq[i]++;
                }
            }
        }
    }
}

```

```

    }
    }
}
for(i=0; i<size; i++)
{
    if(freq[i] != 0)
    {
        printf("%d - %d\n", arr[i], freq[i]);
    }
}
}

```

Status : Correct

Marks : 10/10

9. Write a Program to remove duplicate elements from the sorted array.

Example:

Input: arr = {1, 2, 3, 4, 4}

Output: arr = {1, 2, 3, 4}

Answer

```

// You are using GCC
#include<stdio.h>

```

```

int remove_duplicate_elements(int arr[], int n)
{

```

```

    if (n==0 || n==1)
        return n;

```

```

    int temp[n];

```

```

    int j = 0;
    int i;
    temp[j++] = arr[0];
    for (i=0; i<n; i++)
        if (arr[i] != arr[i+1])

```

```

    temp[j++] = arr[i+1];

    for (i=0; i<j; i++)
        arr[i] = temp[i];

    return i-1;
}

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

    for (i=0; i<n; i++)
        printf("%d ",arr[i]);

    return 0;
}

```

Status : Correct

Marks : 10/10

10. Write a Program to find the sum of perfect square elements in an array.

Answer

```

// You are using GCC
#include<stdio.h>
#include<math.h>

int isPerfectSquare(int number)
{

```

```

int iVar;
float fVar;

fVar=sqrt((double)number);
iVar=fVar;
if(iVar==fVar)
    return number;
else
    return 0;
}

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

```

Status : Correct

Marks : 10/10