

1905387  
CHAUDHARY HAMDAN  
ALGO LAB 1  
Date : 09-07-2021

Q1.

WAP in C to input n integers in to an array. Let us assume that there can be duplicates elements. Write a program to search an element in the array in such a way that we get the highest frequency if there are duplicate elements.

Code:

```
#include<stdio.h>

int main() {

    int n;
    scanf("%d", &n);
    int arr[n], arr2[n];
    for (int i = 0; i < n; i++) {
        scanf("%d", (arr + i));
        arr2[i] = arr[i];
    }

    int freq = 0;
    for (int i = 0; i < n; i++) {
        int c = 1;
        if (arr[i] != -1) {
            for (int j = i + 1; j < n; j++) {
                if (arr[i] == arr[j]) {
                    arr[j] = -1;
                    c++;
                }
            }
            if (c > freq) {
                freq = c;
            }
        }
    }
}
```

```

    printf("%d\n", freq);

    for (int i = 0; i < n; i++) {
        int c = 0;
        for (int j = i; j < n; j++) {
            if (arr2[i] == arr2[j]) {
                c++;
            }
        }
        if (c == freq) {
            printf("Number with highest frequency: %d with count = %d\n", arr[i],
freq);
        }
    }

    return 0;
}

```

Output:

```

C:\Users\KIIT\Desktop\Algo Lab>lab1q1
10
1 4 2 3 5 6 3 2 4 5
2
Number with highest frequency: 4 with count = 2
Number with highest frequency: 2 with count = 2
Number with highest frequency: 3 with count = 2
Number with highest frequency: 5 with count = 2

```

Q2.

WAP for finding i and j in an array A for any key such that  $A[j]^2 + A[i]^2 == \text{key}$ .

Code:

```
#include<stdio.h>

int main() {

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

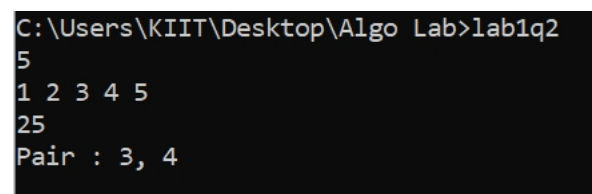
    int key, flag = 1;
    scanf("%d", &key);

    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            int a = arr[i] * arr[i] + arr[j] * arr[j];
            if (a == key) {
                flag = 0;
                printf("Pair : %d, %d\n", arr[i], arr[j]);
            }
        }
    }

    if (flag) {
        printf("No pairs found\n");
    }

    return 0;
}
```

Output:



```
C:\Users\KIIT\Desktop\Algo Lab>lab1q2
5
1 2 3 4 5
25
Pair : 3, 4
```

Q3.

Suppose an array A has n distinct integers. Increasing sequence is given as A[0].....A[k] and decreasing sequence is given as A[k+1].....A[n-1]. Write a program for finding k.

Code:

```
#include<stdio.h>

int main() {

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

        if (i && (arr[i] < arr[i - 1])) {
            printf("Index : %d", (i - 1));
            break;
        }
    }

    return 0;
}
```

Output:

```
C:\Users\KIIT\Desktop\Algo Lab>lab1q3
10
-2 3 5 13 20 17 12 6 1 0
Index : 4
```

Q4.

WAP to display an array of  $n$  integers ( $n > 1$ ) in  $O(n)$  time, where at every index of the array should contain the product of all elements in the array except the element at the given index. No additional array declaration is allowed.

*Example:* Input : 10, 4, 1, 6, 2

Output : 48,120,480,80,240

Code:

```
#include<stdio.h>

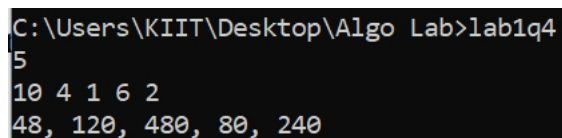
int main() {

    int n, tot = 1;
    scanf("%d", &n);
    int arr[n];
    for (int i = 0; i < n; i++) {
        scanf("%d", (arr + i));
        tot *= arr[i];
    }

    for (int i = 0; i < n - 1; i++) {
        printf("%d, ", tot / arr[i]);
    }
    printf("%d\n", tot / arr[n - 1]);

    return 0;
}
```

Output:



```
C:\Users\KIIIT\Desktop\Algo Lab>lab1q4
5
10 4 1 6 2
48, 120, 480, 80, 240
```