Graph Theory - S003 (E021)

Solved Challenges **0/2**

Back To Challenges List

Minimum Swaps - Ascending Order

ID:11098 **Solved By 793 Users**

The program must accept N integers from 1 to N in any order as the input. The program must print the minimum number of swaps required to order those N integers in ascending order as the output.

Boundary Condition(s):
1 <= N <= 1000

Input Format:
The first line contains N.
The second line contains N integers separated by a space.

Output Format:
The first line contains the minimum number of swaps required.

Example Input/Output 1:
Input:
5
2 3 1 5 4

Output:
3

Explanation:
The integers 5 and 4 can be swapped.
Now the integers become 2 3 1 4 5.
Then the integers 2 and 1 can be swapped.
Now the integers become 1 3 2 4 5.
Then the integers 3 and 2 can be swapped.
Now the integers become 1 2 3 4 5.
So at least 3 swaps are required.
Hence 3 is printed.

Example Input/Output 2:
Input:
7
2 7 6 3 5 4 1

Output:
4

Max Execution Time Limit: 500 millisecs

Ambiance

C (gcc 8.x)

Reset

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

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

    int visited[n+1];
    for(int i=1;i<=n;i++)
        visited[i] = 0;

    int totalswaps=0;
    for(int index=1;index<=n;index++)
    {
        if(visited[arr[index]])
            continue;
        if(arr[index] == index)
        {
            visited[arr[index]] = 1;
            continue;
        }
        int edges=0,cycleIndex=index;
        while(!visited[arr[cycleIndex]])
        {
            visited[arr[cycleIndex]]=1;
            edges++;
            cycleIndex = arr[cycleIndex];
        }
        totalswaps += edges-1;
    }
    printf("%d",totalswaps);

}
```

Code did not pass the execution

TestCase ID: 63593

Input:

```
5
2 3 1 5 4
```

Expected Output:

```
3
```

Your Program Output:

```
4
```

Save Run

□ Run with a custom test case (Input/Output)