| | |
|---|---|
| **Started on** | Friday, 25 October 2024, 1:51 PM |
| **State** | Finished |
| **Completed on** | Monday, 18 November 2024, 8:52 PM |
| **Time taken** | 24 days 7 hours |
| **Marks** | 1.00/1.00 |
| **Grade** | **10.00** out of 10.00 (**100**%) |

Question **1**

Correct

Mark 1.00 out of 1.00

Write a Program to Implement the Quick Sort Algorithm

Input Format:
The first line contains the no of elements in the list-n
The next n lines contain the elements.

Output:
Sorted list of elements

**For example:**

| Input | Result |
|---|---|
| 5<br>67 34 12 98 78 | 12 34 67 78 98 |

**Answer:**

```c
#include <stdio.h>

void partition(int arr[], int low, int high, int *pi) {
    int p = arr[low];
    int i = low + 1;
    int j = high;

    while (i <= j) {
        while (i <= high && arr[i] <= p) {
            i++;
        }

        while (j >= low && arr[j] > p) {
            j--;
        }

        if (i < j) {

            int temp = arr[i];
            arr[i] = arr[j];
            arr[j] = temp;
        }
    }

    arr[low] = arr[j];
    arr[j] = p;
    *pi = j;
}

void qsort(int arr[], int low, int high) {
    if (low < high) {
        int pi;
        partition(arr, low, high, &pi);
        qsort(arr, low, pi - 1);
        qsort(arr, pi + 1, high);
    }
}

int main() {
    int n;
    ;
    scanf("%d", &n);


    int arr[n];
```

```
    int arr[n];

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

    qsort(arr, 0, n - 1);

```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✔ | 5<br>67 34 12 98 78 | 12 34 67 78 98 | 12 34 67 78 98 | ✔ |
| ✔ | 10<br>1 56 78 90 32 56 11 10 90 114 | 1 10 11 32 56 56 78 90 90 114 | 1 10 11 32 56 56 78 90 90 114 | ✔ |
| ✔ | 12<br>9 8 7 6 5 4 3 2 1 10 11 90 | 1 2 3 4 5 6 7 8 9 10 11 90 | 1 2 3 4 5 6 7 8 9 10 11 90 | ✔ |

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 4-Two Elements sum to x

Jump to...

1-DP-Playing with Numbers ►