# 5-Implementation of Quick Sort

| | |
|---|---|
| **Started on** | Tuesday, 30 September 2025, 12:16 PM |
| **State** | Finished |
| **Completed on** | Tuesday, 30 September 2025, 12:17 PM |
| **Time taken** | 33 secs |
| **Marks** | 1.00/1.00 |
| **Grade** | **10.00** out of 10.00 (**100**%) |

**Question 1** Correct Mark 1.00 out of 1.00 Flag question

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 | 12 34 67 78 98 |

| 67 34 12 98 78 | |
|---|---|

**Answer:**

```c
#include <stdio.h>

void swap(int *a, int *b) {
    int t = *a;
    *a = *b;
    *b = t;
}

int partition(int arr[], int low, int high) {
    int pivot = arr[high];
    int i = low - 1;
    for (int j = low; j < high; j++) {
        if (arr[j] < pivot) {
            i++;
            swap(&arr[i], &arr[j]);
        }
    }
    swap(&arr[i + 1], &arr[high]);
    return i + 1;
}

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

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

    quickSort(arr, 0, n - 1);

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

    return 0;
}

```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✔ | 5 | 12 34 67 78 98 | 12 34 67 78 98 | ✔ |

| | | | | |
|---|---|---|---|---|
| ✔ | 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.

Finish review

Back to Course