Design and Analysis of Algorithms Assignment - 1

Name: Dhanraj Kore

Div: TY B

Roll No: 60

Batch : B-3

Quick Sort Implementation

Approach 1: Simple Quick Sort:

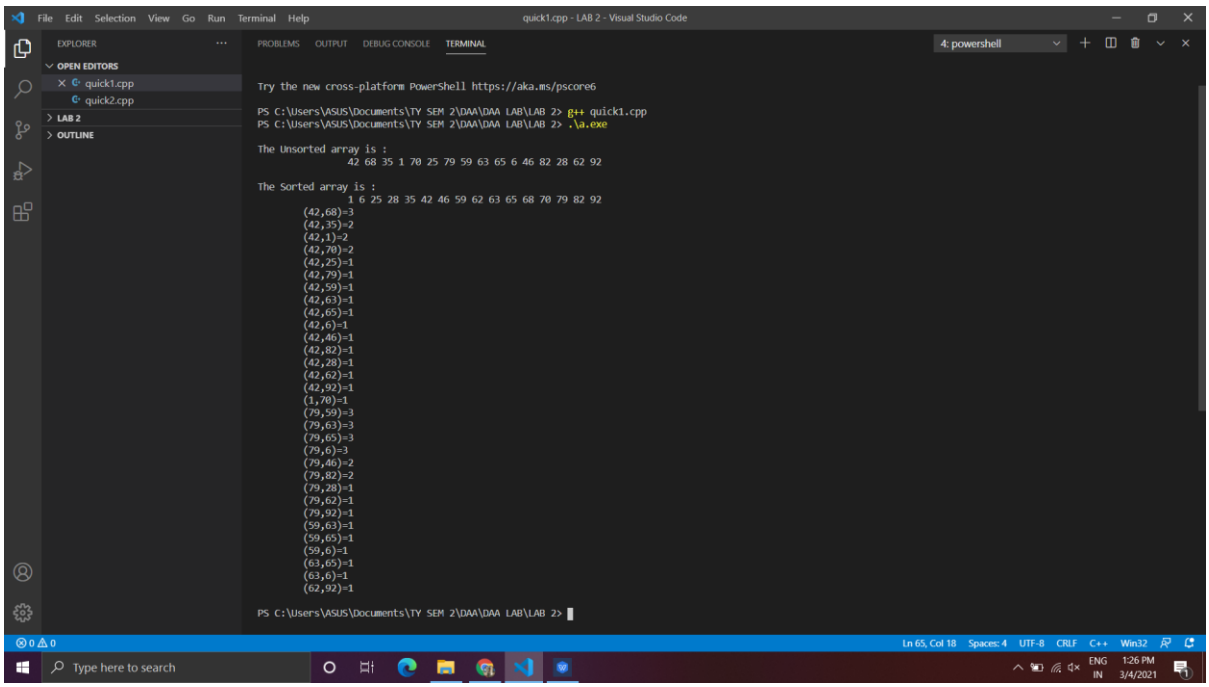
CODE:

```
#include <bits/stdc++.h>
using namespace std;
#define n 16
int Partition(int l, int h, int a[], int c[][n])
{
    int pivot = a[l];
    int i = l, j = h - 1;
    while (i < j)
    {
        while (a[i] <= pivot)
        {
            c[i][l]++;
            c[l][i]++;
            i++;
        }
        while (a[j] > pivot)
        {
            c[j][l]++;
            c[l][j]++;
            j--;
```

```
        }
        if (i < j)
            swap(a[i], a[j]);
    }
    swap(a[l], a[j]);
    return j;
}

void QuickSort(int a[], int c[][n], int l, int h)
{
    if (l < h)
    {
        int k = Partition(l, h, a, c);
        QuickSort(a, c, l, k);
        QuickSort(a, c, k + 1, h);
    }
}
int main()
{
    int a[n];
    int b[n];
    for (int i = 0; i < n; i++)
    {
        a[i] = rand() % 100 + 1;
    }
    cout << "\nThe Unsorted array is : " << endl
         << "\t\t";
    for (int i = 0; i < n; i++)
    {
        cout << a[i] << " ";
    }
    for (int i = 0; i < n; i++)
    {
        b[i] = a[i];
    }
    cout << endl;
```

```
    int c[n][n] = {0};
    QuickSort(a, c, 0, n);
    cout << "\nThe Sorted array is : " << endl
         << "\t\t";
    for (int i = 0; i < n; i++)
    {
        cout << a[i] << " ";
    }
    cout << endl;
    for (int i = 0; i < n; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            if (c[i][j] != 0)
            {
                cout << "\t(" << b[i] << "," << b[j] << ")=" << c[i][j] << endl;
            }
        }
    }
    cout << endl;
}
```

O/P:



Approach 2: Randomized Quick Sort:

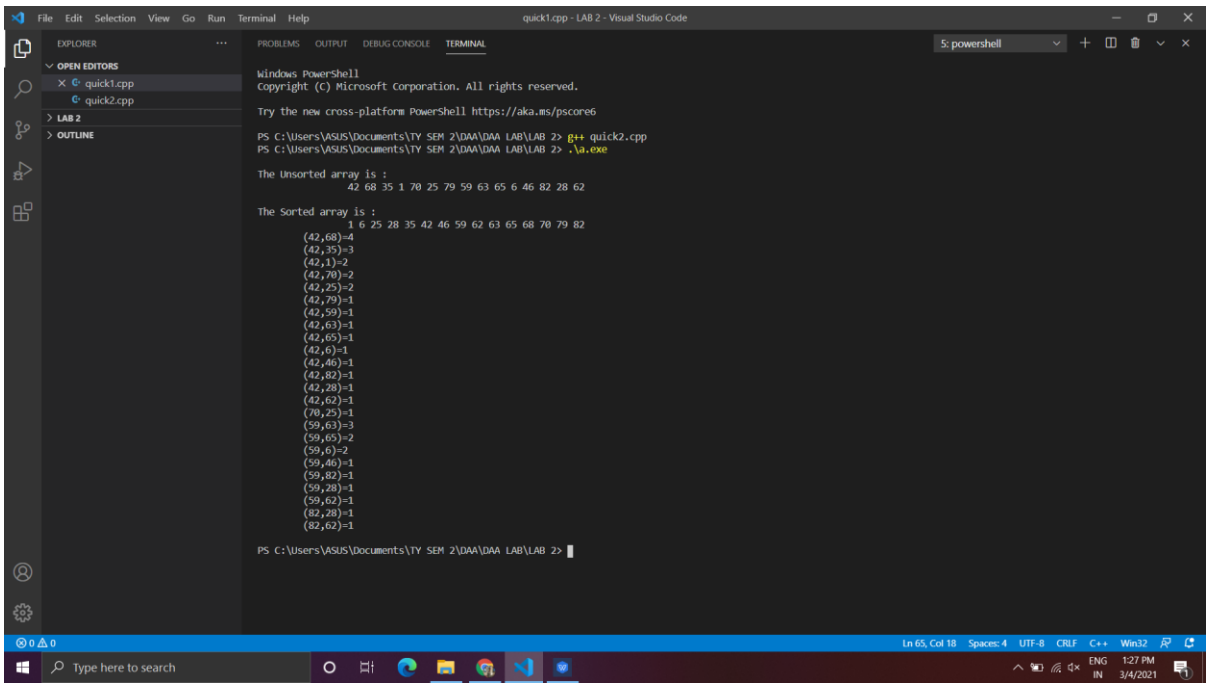
CODE:

```
#include <bits/stdc++.h>

using namespace std;

#define n 15


int Partition(int l, int h, int a[], int c[][n])

{

    int idx = (rand() % (h - l)) + l;

    swap(a[l], a[idx]);

    int pivot = a[l];

    //cout<<a[idx]<<endl;

    int i = l, j = h - 1;

    while (i < j)

    {
```

```
        while (a[i] <= pivot)
        {
            c[i][l]++;
            c[l][i]++;
            i++;
        }
        while (a[j] > pivot)
        {
            c[j][l]++;
            c[l][j]++;
            j--;
        }
        if (i < j)
            swap(a[i], a[j]);
    }
    swap(a[l], a[j]);
    return j;
}

void QuickSort(int a[], int c[][n], int l, int h)
{
    if (l < h)
    {
        int k = Partition(l, h, a, c);
        QuickSort(a, c, l, k);
        QuickSort(a, c, k + 1, h);
    }
}
int main()
{
    int a[n];
    int b[n];
    for (int i = 0; i < n; i++)
    {
        a[i] = rand() % 100 + 1;
    }
```

```
    cout << "\nThe Unsorted array is : " << endl
         << "\t\t";
    for (int i = 0; i < n; i++)
    {
        cout << a[i] << " ";
    }
    for (int i = 0; i < n; i++)
    {
        b[i] = a[i];
    }
    cout << endl;
    int c[n][n] = {0};
    QuickSort(a, c, 0, n);
    cout << "\nThe Sorted array is : " << endl
         << "\t\t";
    for (int i = 0; i < n; i++)
    {
        cout << a[i] << " ";
    }
    cout << endl;
    for (int i = 0; i < n; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            if (c[i][j] != 0)
            {
                cout << "\t(" << b[i] << "," << b[j] << ")=" << c[i][j] << endl;
            }
        }
    }
    cout << endl;
}
```

O/P:


File Edit Selection View Go Run Terminal Help
quick1.cpp - LAB 2 - Visual Studio Code
EXPLORER
OPEN EDITORS
quick1.cpp
quick2.cpp
LAB 2
OUTLINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
5: powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\ASUS\Documents\TY SEM 2\DAA\DAA LAB\LAB 2> g++ quick2.cpp
PS C:\Users\ASUS\Documents\TY SEM 2\DAA\DAA LAB\LAB 2> .\a.exe
The Unsorted array is :
42 68 35 1 70 25 79 59 63 65 6 46 82 28 62
The Sorted array is :
1 6 25 28 35 42 46 59 62 63 65 68 70 79 82
(42,68)=4
(42,35)=3
(42,1)=2
(42,70)=2
(42,25)=2
(42,79)=1
(42,59)=1
(42,63)=1
(42,65)=1
(42,6)=1
(42,46)=1
(42,82)=1
(42,28)=1
(42,62)=1
(70,25)=1
(59,63)=3
(59,65)=2
(59,6)=2
(59,46)=1
(59,82)=1
(59,28)=1
(59,62)=1
(82,28)=1
(82,62)=1
PS C:\Users\ASUS\Documents\TY SEM 2\DAA\DAA LAB\LAB 2>
Ln 65, Col 18 Spaces: 4 UTF-8 CRLF C++ Win32
Type here to search
ENG IN
1:27 PM
3/4/2021