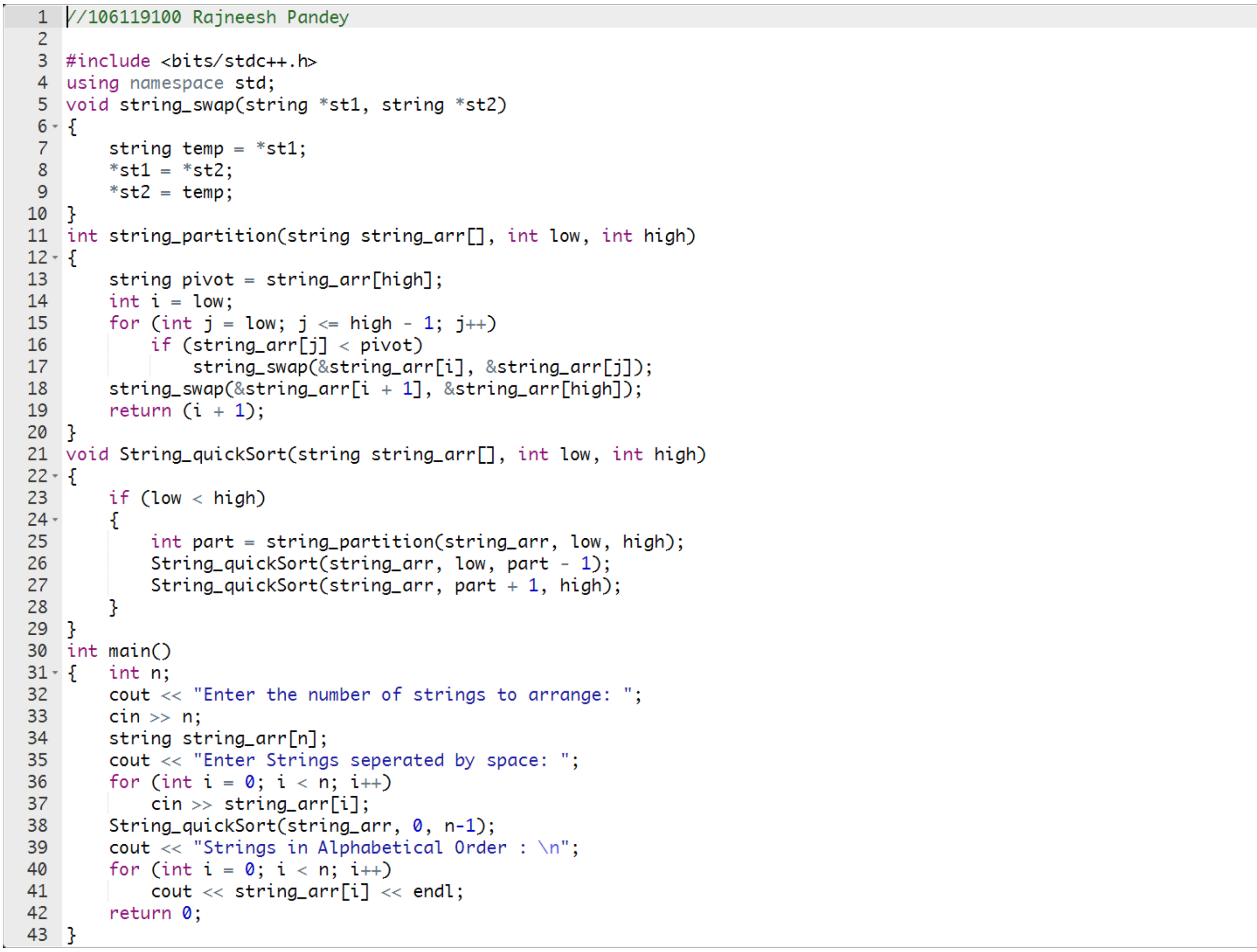
LAB-3 106119100 - RAJNEESH PANDEY

Q 1. Write a program to arrange the set of strings in alphabetical order using divide and conquer technique.

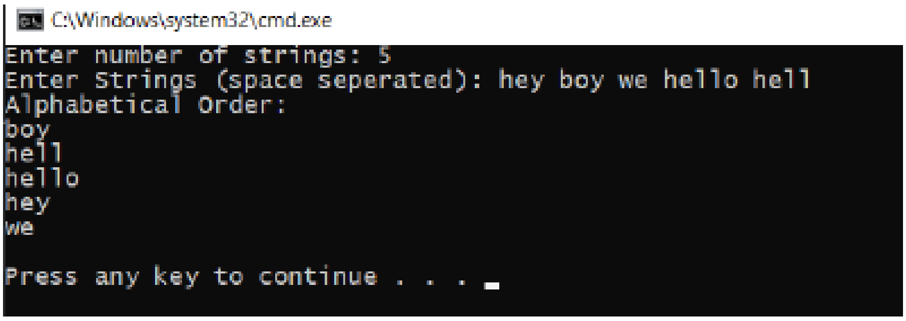
Solution: Alphabetical order is same as sorting the strings. Hence to use divide and conquer techniques we can either sort the array of strings using merge sort or quicksort techniques. The comparators > and < is overloaded in-built in C++ to compare two strings. Here quick sort is used to solve the problem.



```
//106119100 Rajneesh Pandey

#include <bits/stdc++.h>
using namespace std;
void string_swap(string *st1, string *st2)
{
    string temp = *st1;
    *st1 = *st2;
    *st2 = temp;
}
int string_partition(string string_arr[], int low, int high)
{
    string pivot = string_arr[high];
    int i = low;
    for (int j = low; j <= high - 1; j++)
        if (string_arr[j] < pivot)
            string_swap(&string_arr[i], &string_arr[j]);
    string_swap(&string_arr[i + 1], &string_arr[high]);
    return (i + 1);
}
void String_quickSort(string string_arr[], int low, int high)
{
    if (low < high)
    {
        int part = string_partition(string_arr, low, high);
        String_quickSort(string_arr, low, part - 1);
        String_quickSort(string_arr, part + 1, high);
    }
}
int main()
{   int n;
    cout << "Enter the number of strings to arrange: ";
    cin >> n;
    string string_arr[n];
    cout << "Enter Strings seperated by space: ";
    for (int i = 0; i < n; i++)
        cin >> string_arr[i];
    String_quickSort(string_arr, 0, n-1);
    cout << "Strings in Alphabetical Order : \n";
    for (int i = 0; i < n; i++)
        cout << string_arr[i] << endl;
    return 0;
}
```

OUTPUT

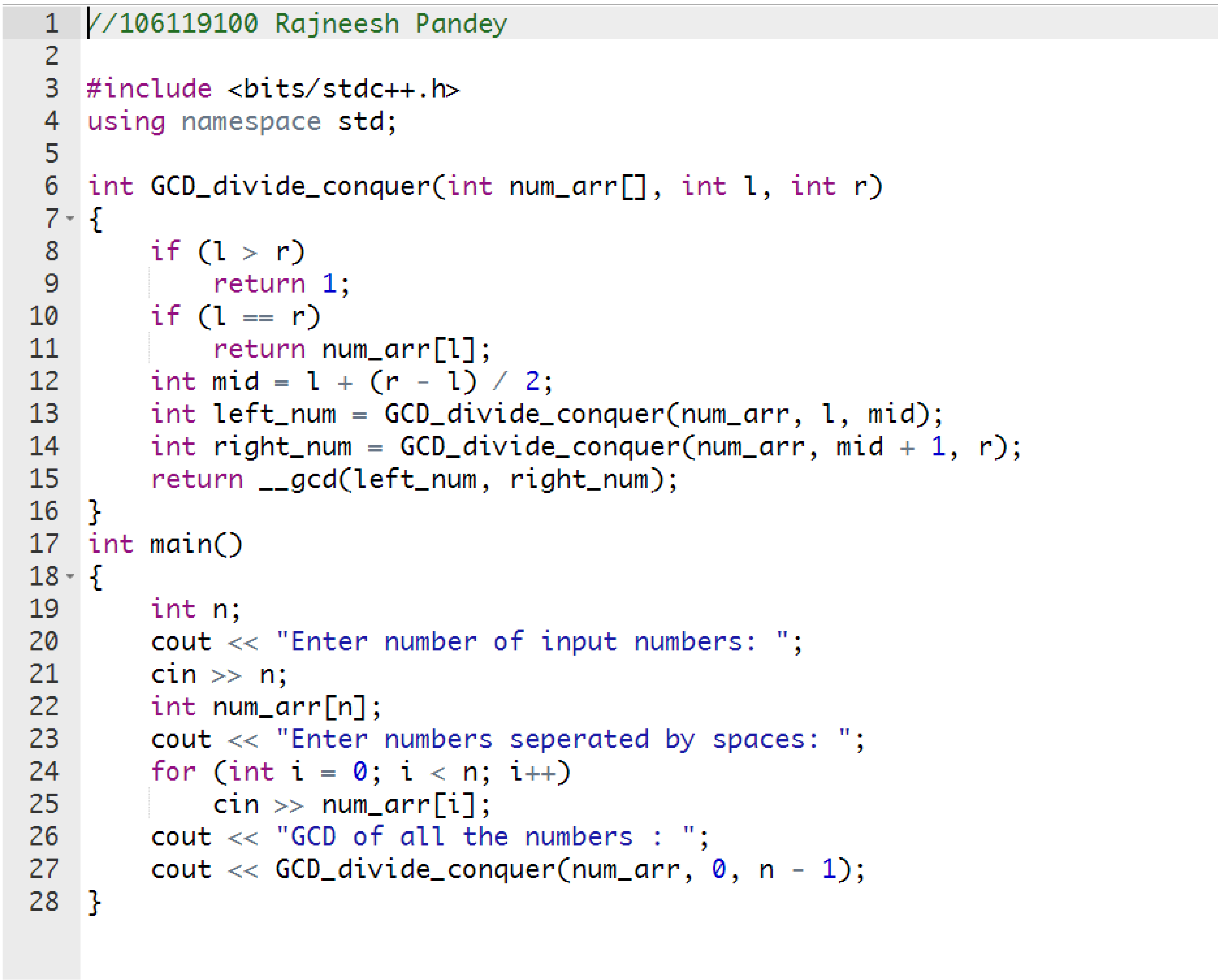


```
Enter number of strings: 5
Enter Strings (space seperated): hey boy we hello hell
Alphabetical Order:
boy
hell
hello
hey
we

Press any key to continue . . .
```

2. Write a program to find GCD of ‘n’ numbers using divide and conquer technique.

Solution. GCD (a, b, c, ….) = GCD (GCD (a, b), c) ….

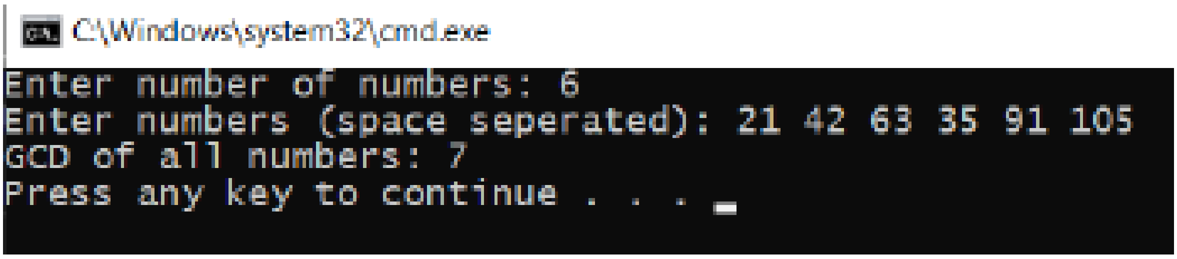


```cpp
//106119100 Rajneesh Pandey

#include <bits/stdc++.h>
using namespace std;

int GCD_divide_conquer(int num_arr[], int l, int r)
{
    if (l > r)
        return 1;
    if (l == r)
        return num_arr[l];
    int mid = l + (r - l) / 2;
    int left_num = GCD_divide_conquer(num_arr, l, mid);
    int right_num = GCD_divide_conquer(num_arr, mid + 1, r);
    return __gcd(left_num, right_num);
}
int main()
{
    int n;
    cout << "Enter number of input numbers: ";
    cin >> n;
    int num_arr[n];
    cout << "Enter numbers seperated by spaces: ";
    for (int i = 0; i < n; i++)
        cin >> num_arr[i];
    cout << "GCD of all the numbers : ";
    cout << GCD_divide_conquer(num_arr, 0, n - 1);
}
```

OUTPUT:



```
Enter number of numbers: 6
Enter numbers (space seperated): 21 42 63 35 91 105
GCD of all numbers: 7
Press any key to continue . . .
```