Question 1-a

#include <iostream>

using namespace std;

//Date:13 October

//Question: Implementation of Selection Sort

int main()

{

    int input,min,temp;

    cout << "Enter the number of inputs: " << endl;

    cin >> input;

    int arr[input];

    for (int i = 0; i < input; i++)

    {

        cin >> arr[i];

    }

    for (int i = 0; i < input-1; i++)

    {

        min = i;

        for (int j = i+1; j < input; j++)

        {

            if (arr[j] < arr[min])

            {

                min = j;

            }

        }

        if ( i != min )

        {

            temp = arr[i];

            arr[i] = arr[min];

            arr[min] = temp;

        }

    }

    for (int i = 0; i < input; i++)

    {

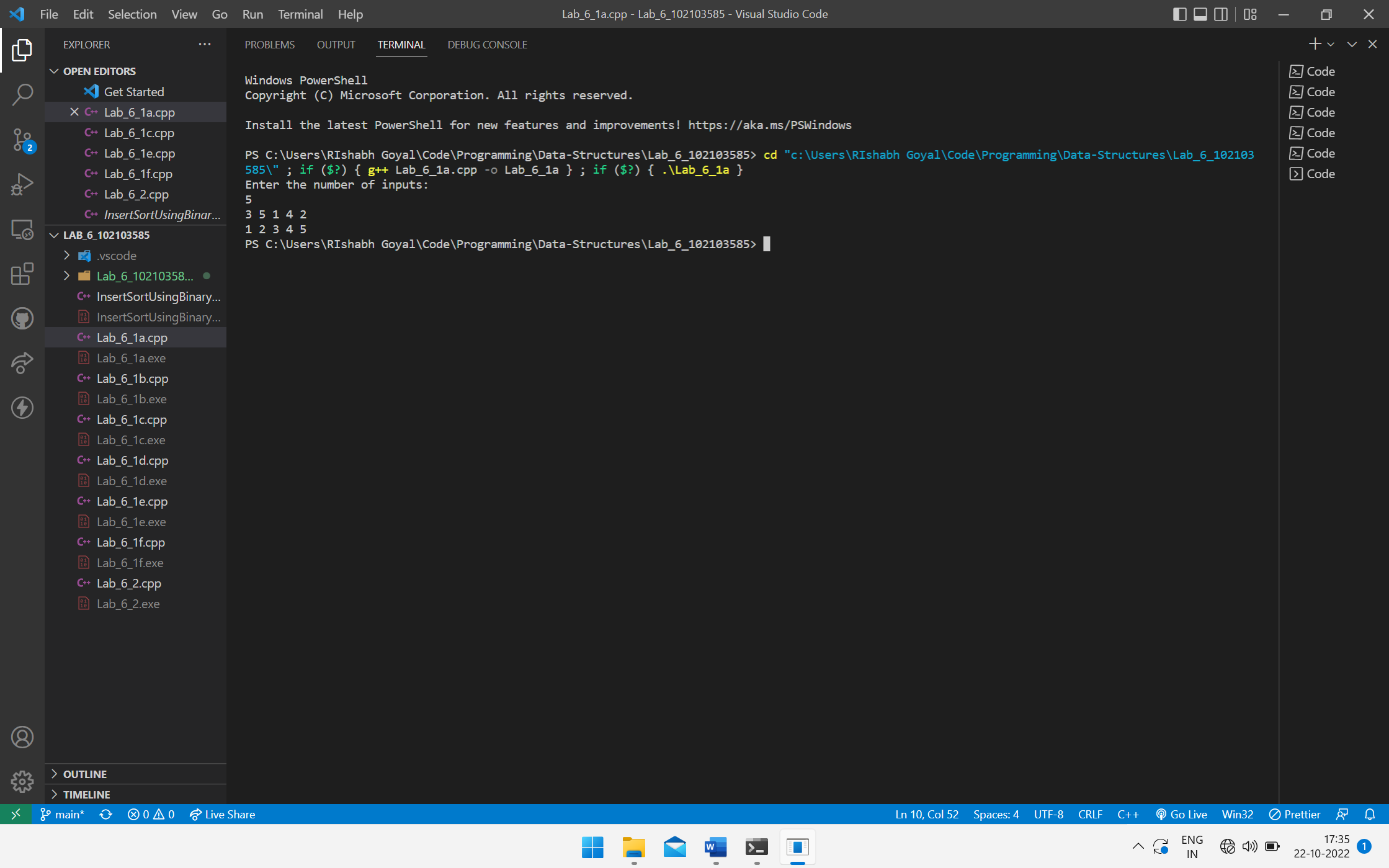
        cout << arr[i] << " ";

    }

    return 0;

}

Output



Question 1-b

#include <iostream>

using namespace std;

//Date:13 October

//Question: Implementation of Insertion Sort

int main()

{

    int in, key, j, passes = 0, swaps = 0, comp = 0,passComp = 0,pos;

    cout << "Number of inputs = " << endl;

    cin >> in;

    int arr[in];

    for (int i = 0; i < in; i++)

    {

        cin >> arr[i];

    }

    for (int i = 1; i < in; i++)

    {

        key = arr[i];

        passes++;

        for (j = i - 1; j >= 0; j--)

        {

            if (arr[j] > key){

                arr[j+1] = arr[j];

                comp++;

                passComp++;

            }

            else{

                comp++;

                break;

            }

        }

        swaps = swaps + passComp;

        passComp = 0;

        arr[j+1] = key;

    }

    for (int i = 0; i < in; i++)

    {

        cout << arr[i] << " ";

    }

    cout << endl;

    cout << "Number of passes = " << passes << endl;

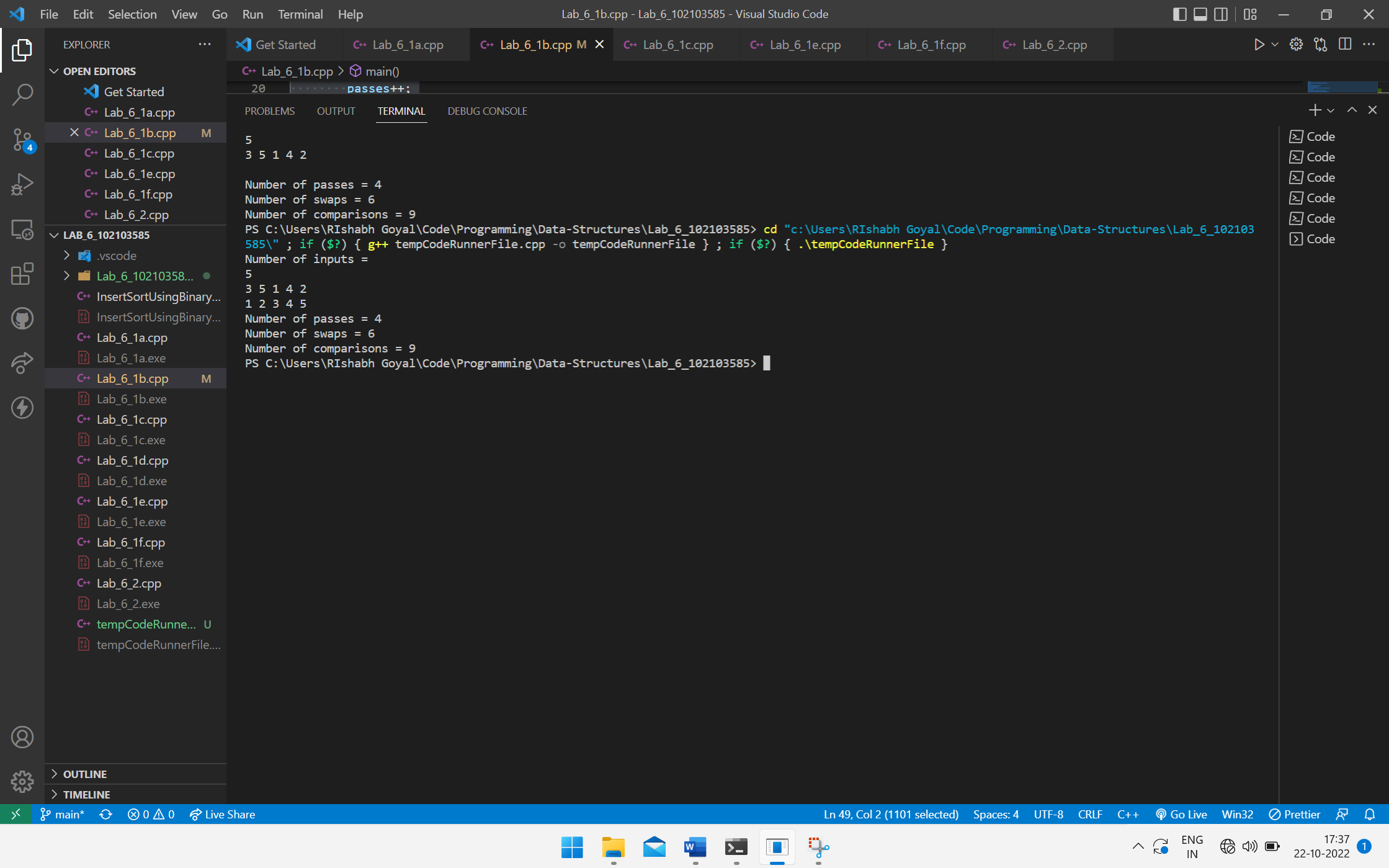
    cout << "Number of swaps = " << swaps << endl;

    cout << "Number of comparisons = " << comp << endl;

    return 0;

}

Output



Question 1-c

#include <iostream>

using namespace std;

//Date:13 October

//Question: Implementation of Bubble Sort

int main()

{

    int input,temp;

    cout << "Enter the number of inputs: " << endl;

    cin >> input;

    int arr[input];

    for (int i = 0; i < input; i++)

    {

        cin >> arr[i];

    }

    for (int i = 0; i < input-1; i++)

    {

        for (int j = 0; j < input - i- 1; j++)

        {

            if (arr[j] > arr[j+1])

            {

                temp = arr[j];

                arr[j] = arr[j+1];

                arr[j+1] = temp;

            }

        }

    }

    for (int i = 0; i < input; i++)

    {

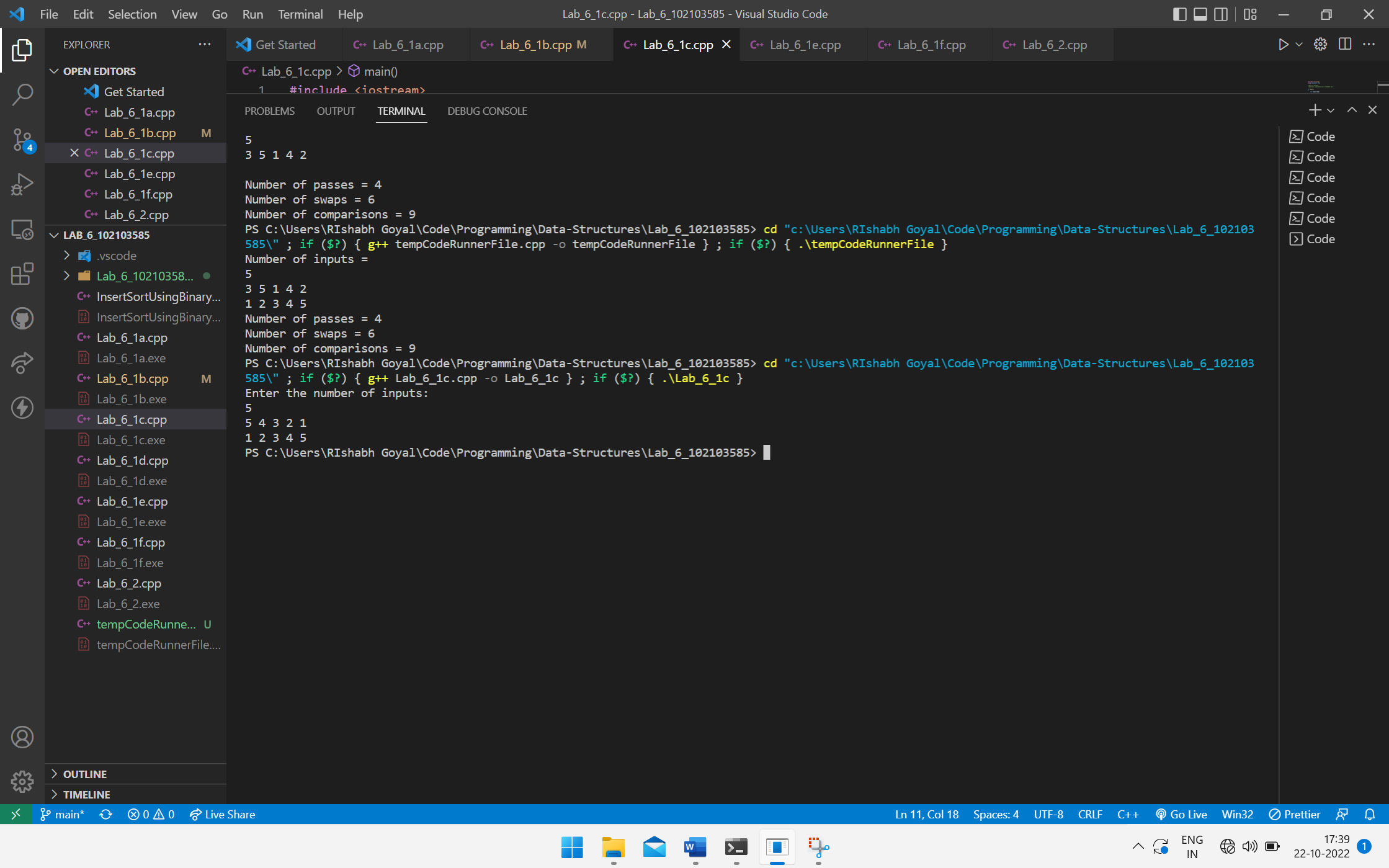
        cout << arr[i] << " ";

    }

    return 0;

}

Output



Question 1-d

#include <iostream>

using namespace std;

//Date:13 October

//Question: Implementation of Merge Sort

int comp = 0;

void Merge(int \* arr, int low, int notMid, int high)

{

    int index1= low,index2 = notMid,index3 = 0;

    int tempArr[high+1];

    while (index1 < notMid && index2 <= high)

    {

        if (arr[index1] <= arr[index2])

        {

            tempArr[index3] = arr[index1];

            comp++;

            index1++;

            index3++;

        }

        else{

            tempArr[index3] = arr[index2];

            comp++;

            index2++;

            index3++;

        }

    }

    if (index1 == notMid)

    {

        while (index2 <= high)

        {

            tempArr[index3] = arr[index2];

            index2++;

            index3++;

        }

    }

    if (index2 > high)

    {

        while (index1 !=  notMid)

        {

            tempArr[index3] = arr[index1];

            index1++;

            index3++;

        }

    }

    for (int i = low; i <= high; i++)

    {

        arr[i] = tempArr[i-low];

    }

}

void MergeSort(int \* arr,int low, int high)

{

    if (low < high)

    {

        int mid = (low + high)/2;

        MergeSort(arr,low,mid);

        MergeSort(arr,mid+1,high);

        Merge(arr,low,mid+1,high);

    }

}

int main()

{

    int input;

    cout << "Number of items in array: " << endl;

    cin >> input;

    int arr[input];

    for (int i = 0; i < input; i++)

    {

        cin >> arr[i];

    }

    MergeSort(arr,0,input-1);

    for (int i = 0; i < input; i++)

    {

        cout << arr[i] << " ";

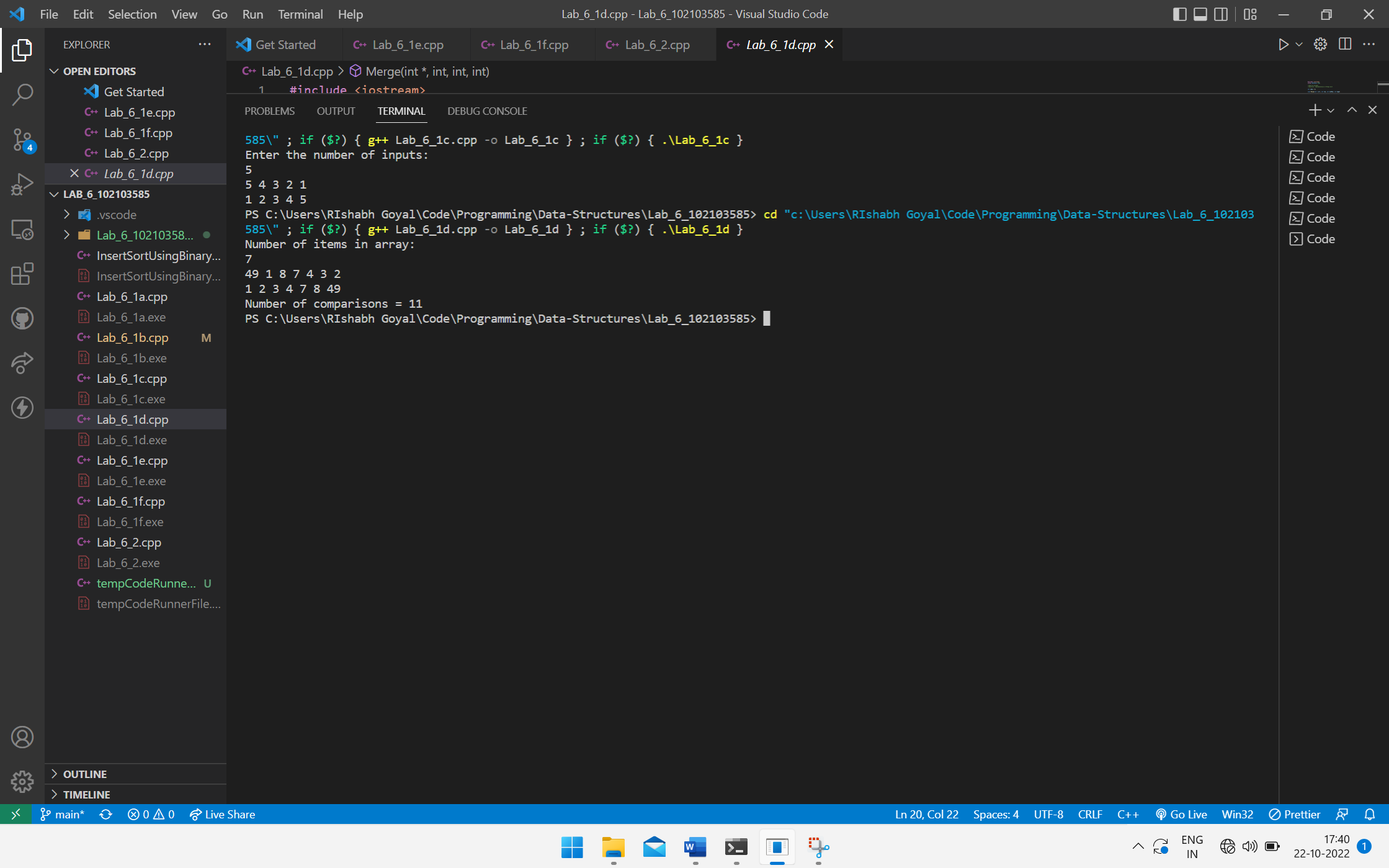
    }

    cout << endl << "Number of comparisons = " << comp << endl;

    return 0;

}

Output



Question 1-e

#include <iostream>

using namespace std;

//Date:13 October

//Question: Implementation of Quick Sort

int partition(int \* arr,int low,int high)

{

    int key = arr[high] , i = low - 1, temp;

    for (int j = low; j < high; j++)

    {

        if (arr[j] < key)

        {

            i++;

            temp = arr[i];

            arr[i] = arr[j];

            arr[j] = temp;

        }

    }

    temp = arr[i+1];

    arr[i+1] = key;

    arr[high] = temp;

    return i+1;

}

void quickSort(int \* arr, int low, int high)

{

    if (low < high)

    {

        int pos = partition(arr,low,high);

        quickSort(arr,low,pos-1);

        quickSort(arr,pos+1,high);

    }

}

int main()

{

    int in;

    cout << "Enter the number of inputs: " << endl;

    cin >> in;

    int arr[in];

    for (int i = 0; i < in; i++)

    {

        cin >> arr[i];

    }

    quickSort(arr,0,in-1);

    for (int i = 0; i < in; i++)

    {

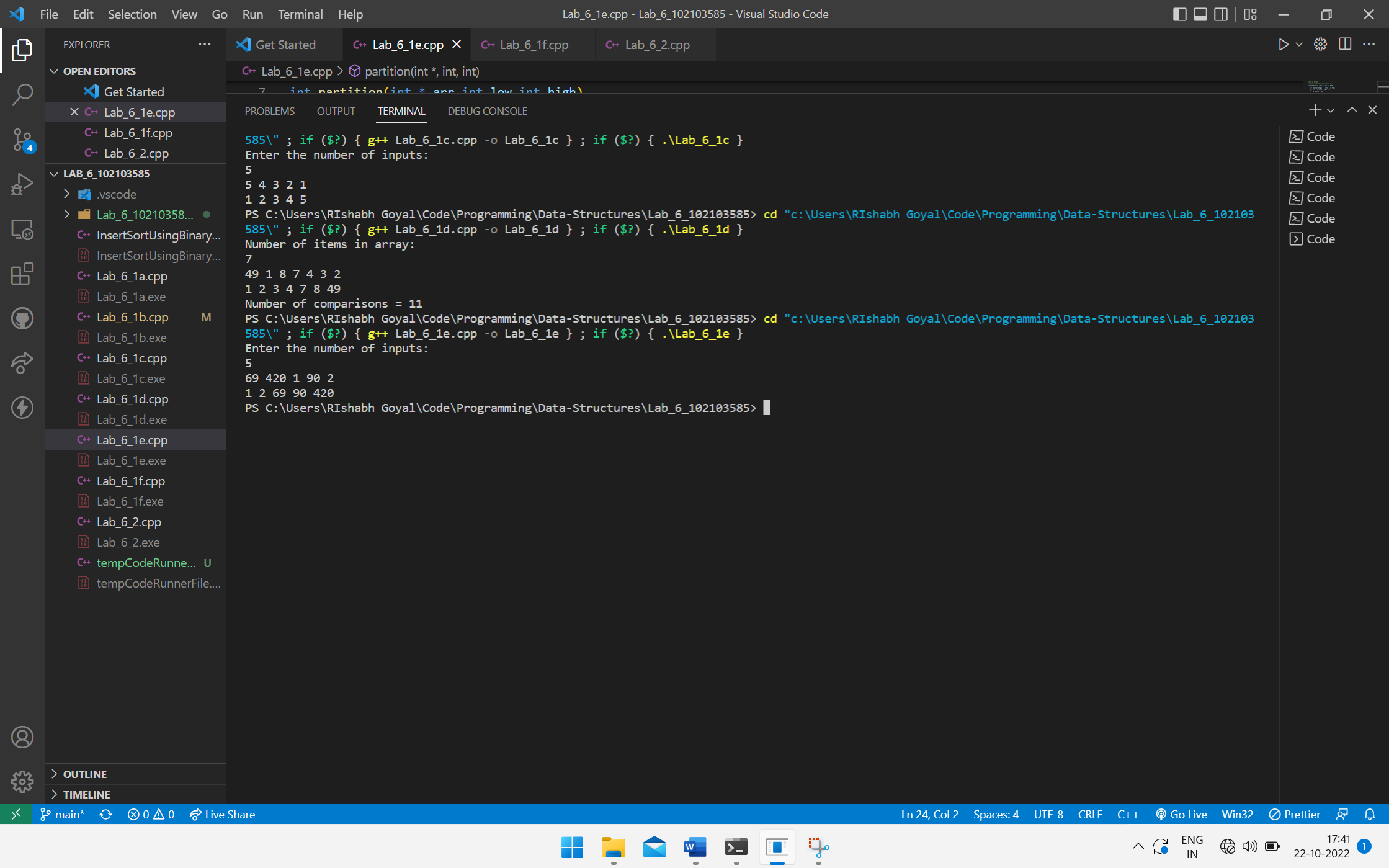
        cout << arr[i] << " ";

    }

    return 0;

}

Output



Question 1-f

#include <iostream>

using namespace std;

//Date:13 October

//Question: Implementation of Counting Sort

int main()

{

    int input,k,num = 1, i = 0;

    cout << "Enter the number of inputs: " << endl;

    cin >> input;

    int inArr[input];

    cout << "Enter the maximum number k that is present in the array: " << endl;

    cin >> k;

    int freqArr[k+1];

    int outArr[input];

    for (int i = 0; i < input; i++)

    {

        cin >> inArr[i];

    }

    for (int i  = 0; i <= k ; i++)

    {

        freqArr[i] = 0;

    }

    for (int i = 0; i < input; i++)

    {

        freqArr[inArr[i]]++;

    }

    while (i < input)

    {

        if (freqArr[num])

        {

            outArr[i] = num;

            freqArr[num]--;

            if (!freqArr[num])

                num++;

            i++;

        }

    }

    for (int i = 0; i < input; i++)

    {

        cout << outArr[i] << " ";

    }

    return 0;

}

Output

