**Practical-4**

**Name:** Abhijeet Vidwan Vyavhare

**Roll No:** 232

**PRN:** 202202040012

**Problem Statement:**

Write a program to implement sliding window mechanisms using

1. Stop and Wait ARQ

2. Go Back N ARQ

3. Selective Repeat ARQ

**Code:**

1. **Stop and Wait ARQ**

#include<iostream>

#include<cstdlib>

#include<ctime>

using namespace std;

// Simulate frame transmission and return true if the frame is received, false otherwise.

bool transmitFrame()

{

    // Simulate frame transmission and random acknowledgment (50% chance of success)

    return rand() % 2 == 0;

}

int main()

{

    srand(time(0)); // Seed the random number generator

    int totalFrames;

    cout<<"\n\n\n\t\t =========== STOP AND WAIT ARQ =============\n\n\n\n";

    cout << "--> Enter the total number of frames to simulate: ";

    cin >> totalFrames;

    int frameNumber = 0;

    cout << "\n==> Simulating Stop-and-Wait ARQ:\n\n";

    while (frameNumber < totalFrames)

    {

        cout << "\nSender sends frame " << frameNumber << endl;

        bool received = transmitFrame();

        if (received)

        {

            cout << "Receiver acknowledges frame " << frameNumber << endl;

            cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

            frameNumber++; // Move to the next frame

        }

        else

        {

            cout << "Receiver does not acknowledge frame " << frameNumber << ", retransmitting...\n";

        }

        cout << endl;

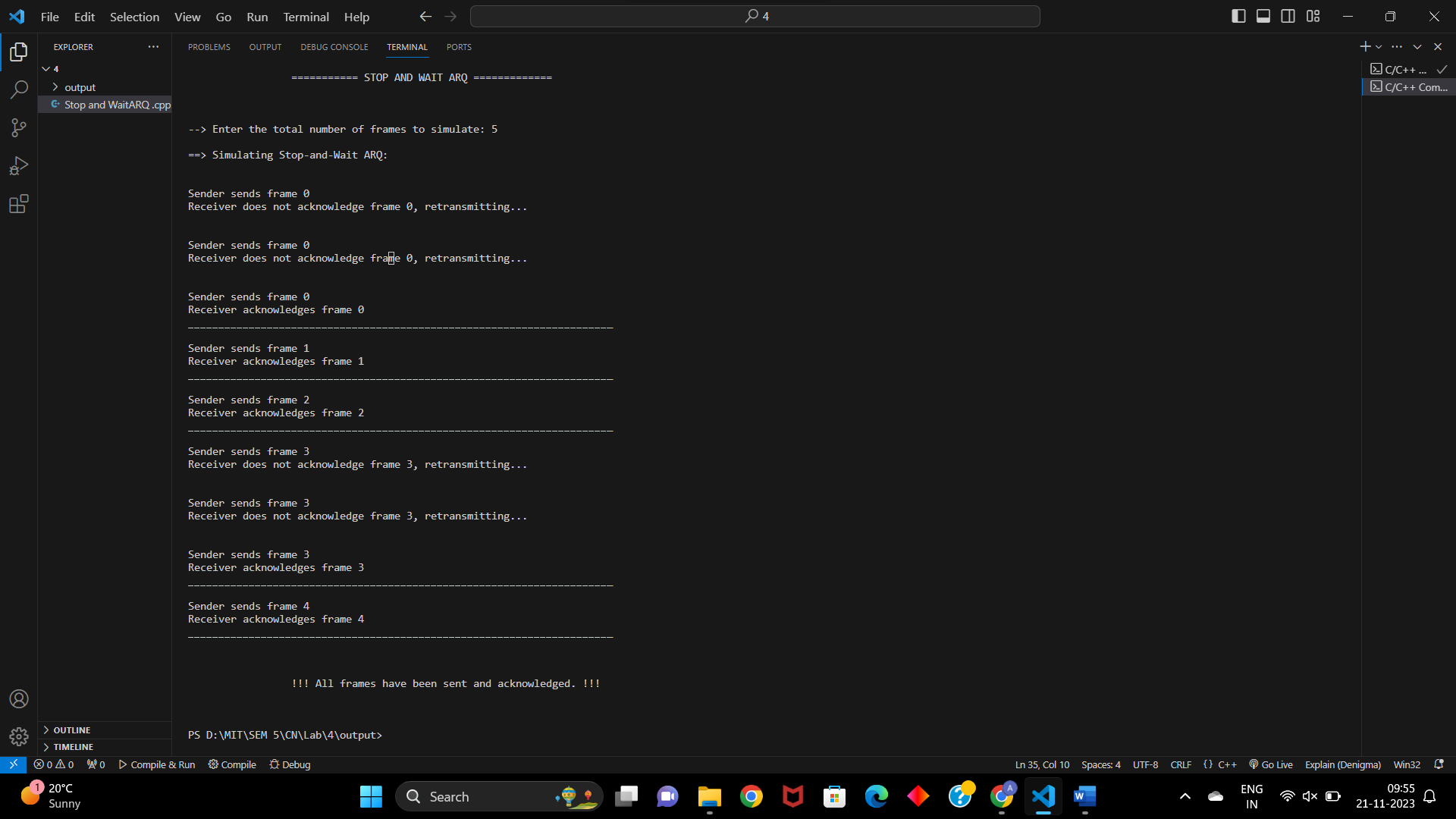
    }

    cout << "\n\n\n\t\t !!! All frames have been sent and acknowledged. !!!\n\n\n\n";

    return 0;

}

**OUTPUT:**

****

1. **Go Back N ARQ**

#include <bits/stdc++.h>

#include <ctime>

#define ll long long int

using namespace std;

void transmission(ll &i, ll &N, ll &tf, ll &tt)

{

    while (i <= tf)

    {

        int z = 0;

        for (int k = i; k < i + N && k <= tf; k++)

        {

            cout << "Sending Frame " << k << "..." << endl;

            tt++;

        }

        for (int k = i; k < i + N && k <= tf; k++)

        {

            int f = rand() % 2;

            if (!f)

            {

                cout << "\nAcknowledgment for Frame " << k << "...\n"

                     << endl;

                z++;

            }

            else

            {

                cout << "\nTimeout!!\n--> Frame Number : " << k << " Not Received" << endl;

                cout << "\*\* Retransmitting Window...\n"

                     << endl;

                break;

            }

        }

        cout << "\n";

        i = i + z;

    }

}

int main()

{

    ll tf, N, tt = 0;

    srand(time(NULL));

    cout << "\n\n\t\t ======== GO BACK N ARQ ========\n\n\n";

    cout << "--> Enter the Total number of frames : ";

    cin >> tf;

    cout << "\n --> Enter the Window Size : ";

    cin >> N;

    ll i = 1;

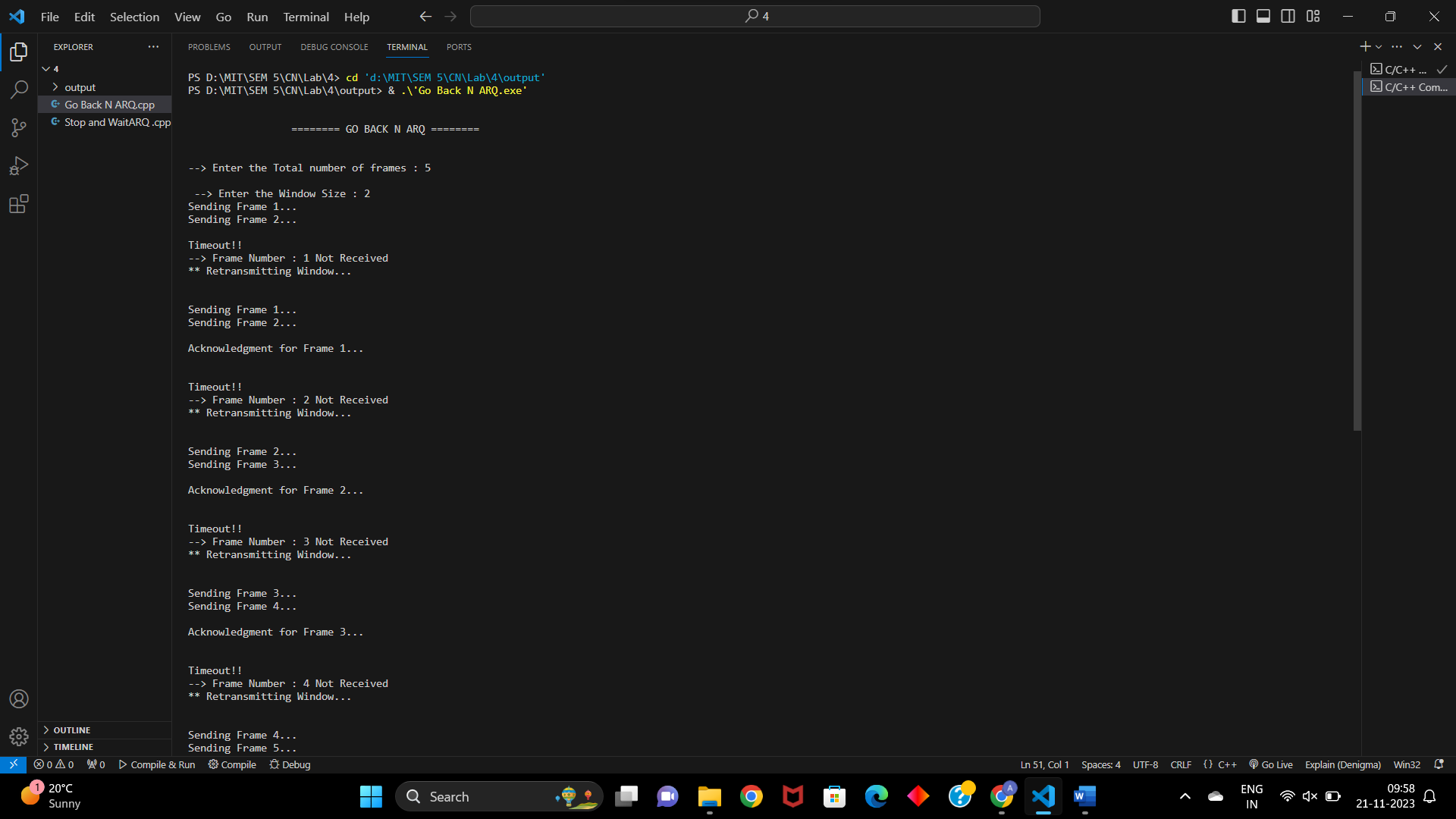
    transmission(i, N, tf, tt);

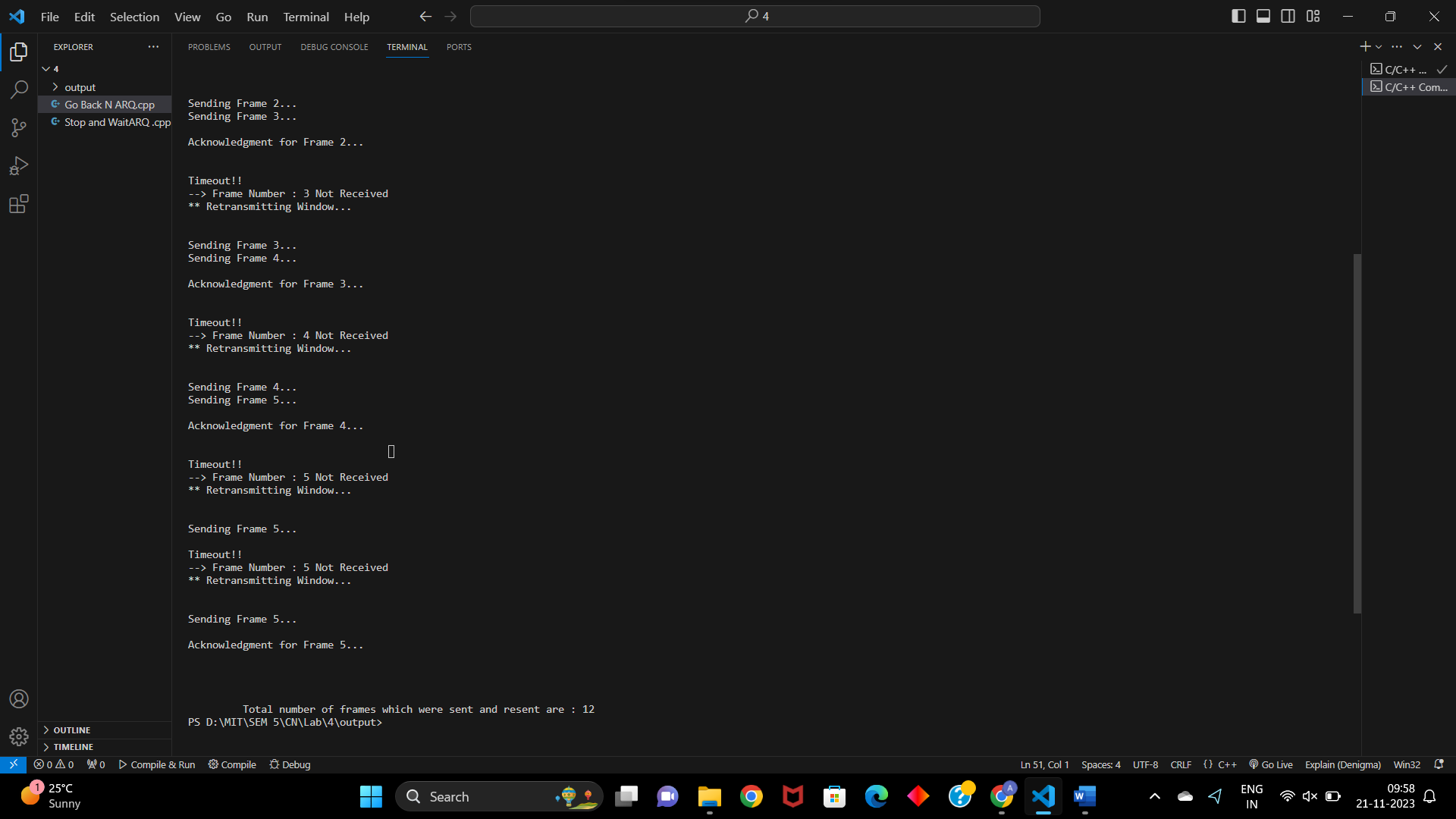
    cout << "\n\n\t Total number of frames which were sent and resent are : " << tt << endl;

    return 0;

}

**OUTPUT:**

****

****

1. **Selective Repeat ARQ**

#include <iostream>

#include <vector>

#include <cstdlib>

#include <ctime>

using namespace std;

// Define the maximum window size and the total number of frames

const int MAX\_WINDOW\_SIZE = 4;

const int TOTAL\_FRAMES = 10;

int main()

{

    int windowSize;

    int totalFramesSent = 0;

    cout << "\n\n\n\t\t ========== SELECTIVE REPEAT ARQ ============\n\n\n";

    cout << "--> Enter window size: ";

    cin >> windowSize;

    vector<bool> acknowledged(TOTAL\_FRAMES, false);

    srand(time(0)); // Seed the random number generator

    cout << "\n-->Simulating Selective Repeat ARQ:\n\n";

    while (totalFramesSent < TOTAL\_FRAMES)

    {

        cout << "Sender's Window: ";

        // Send frames within the current window

        for (int i = totalFramesSent; i < min(totalFramesSent + windowSize, TOTAL\_FRAMES); i++)

        {

            if (!acknowledged[i])

            {

                cout << i << " ";

                // Simulate frame transmission and random acknowledgment

                if (rand() % 2 == 0)

                {

                    acknowledged[i] = true;

                    cout << "(Sent) ";

                }

            }

        }

        cout << "\nReceiver's Window: ";

        // Simulate receiver's acknowledgment

        for (int i = totalFramesSent; i < min(totalFramesSent + windowSize, TOTAL\_FRAMES); i++)

        {

            if (acknowledged[i])

            {

                cout << i << " ";

            }

        }

        cout << "\n\n";

        // Move the window

        totalFramesSent += windowSize;

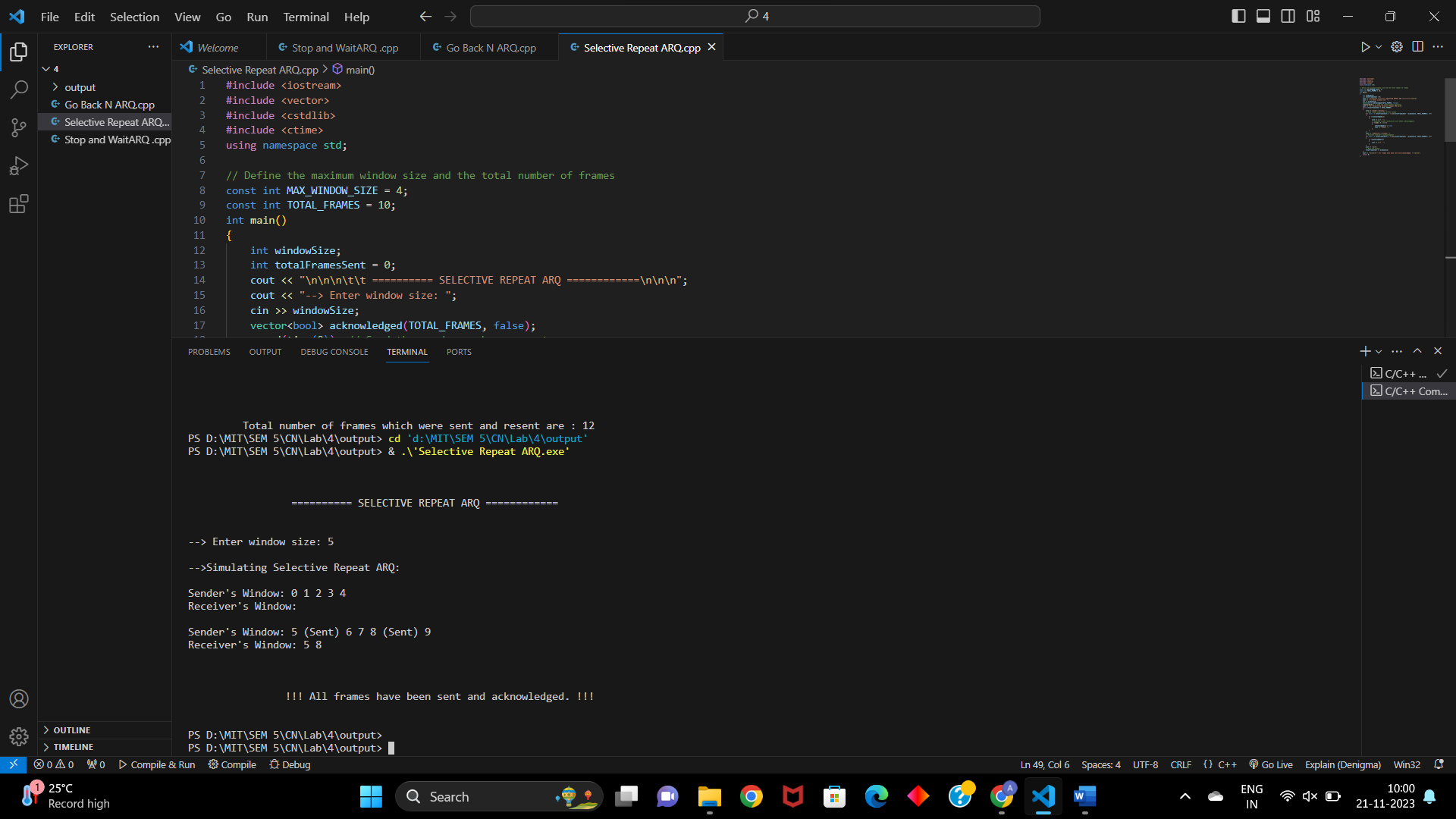
    }

    cout << "\n\n\t\t!!! All frames have been sent and acknowledged. !!!\n\n\n";

    return 0;

}

**OUTPUT:**

****