

**NAME:** Ahtisham Ahmed

**ROLL NO:** 014

**BATCH:** BSCS-Fa21

**LAB no: 10**

**DSA**

1. Counting Numbers

//COUNT NUMBER

#include <iostream>

using namespace std;

int count(int n){

    if(n>=0&&n<=9){

        return 1;

    }

    else{

        n = n / 10;

        return 1 + count(n);

    }

}

int main(){

    int n;

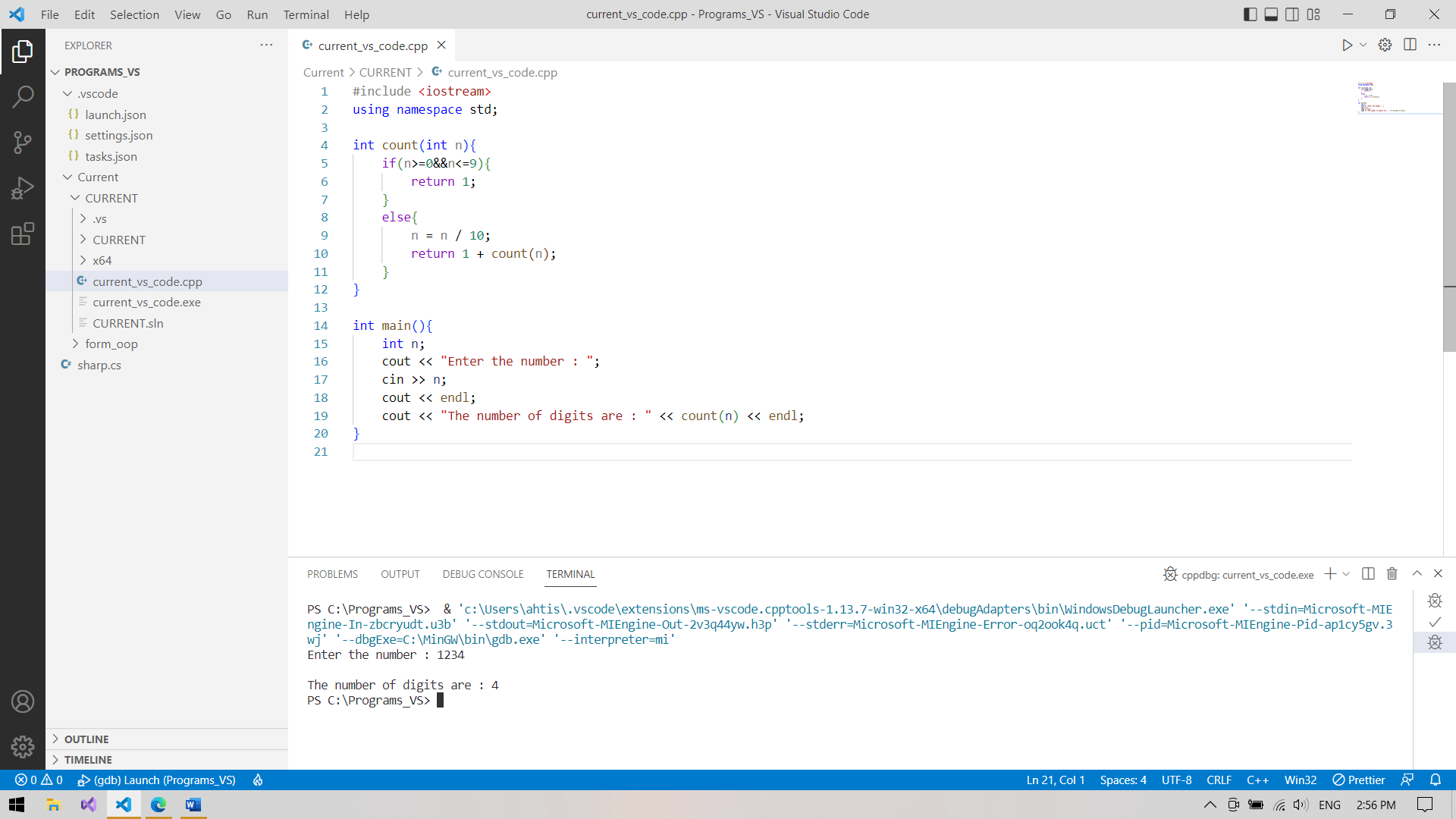
    cout << "Enter the number : ";

    cin >> n;

    cout << endl;

    cout << "The number of digits are : " << count(n) << endl;

}



//Find number in array

#include <iostream>

using namespace std;

int find\_number(int arr[], int size, int n){

    if(size<0){

        cout << "Not Found" << endl;

        return 0;

    }

    else if(arr[size]==n){

        cout << "Found at Index" << endl;

        return size;

    }

    else{

        size--;

        return find\_number(arr, size, n);

    }

}

int main(){

    int n;

    int arr[5] = {1, 2, 3, 4, 5};

    int size = sizeof(arr) / sizeof(arr[0]);

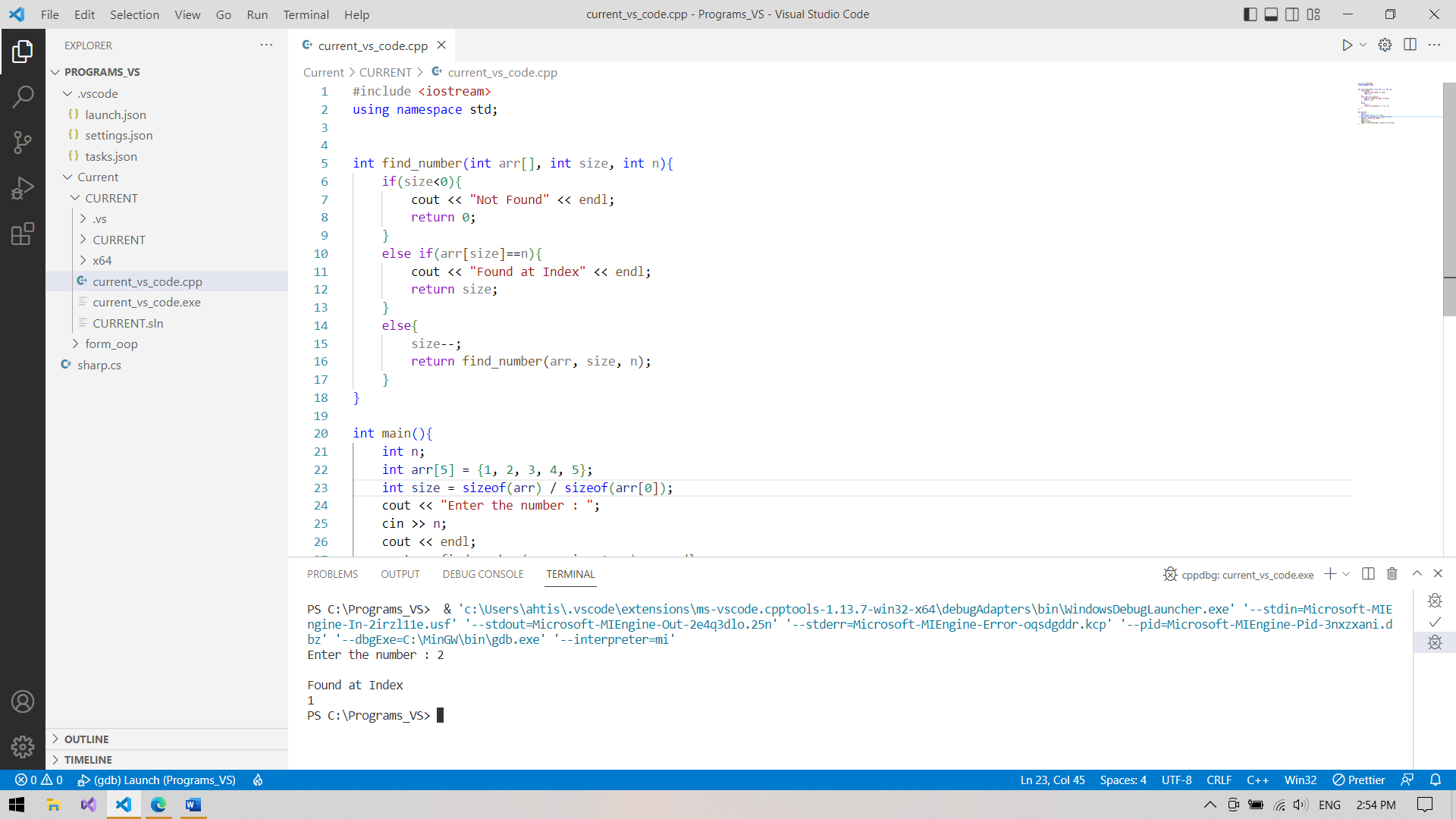
    cout << "Enter the number : ";

    cin >> n;

    cout << endl;

    cout << find\_number(arr, size-1, n) << endl;

}



LAB TASKS  
Question No. 1:  
A. Write code to insert, search and delete data in a hash table using hashing concepts.

//hash tables

#include <iostream>

using namespace std;

int search(int arr[], int num\_search, int size){

    int i = 0;

    while(i<size){

        if(arr[i] == num\_search){

            cout << "The number is found at index : " << i << endl;

            return i;

            break;

        }

        else{

            i++;

        }

    }

    if(arr[i]!=num\_search){

        cout << "Number not found" << endl;

    }

}

void delete\_number(int arr[], int index){

    arr[index] = -1;

}

int main() {

    int arr[20];

    for (int i = 0; i < 20; i++) {

        arr[i] = -1;

    }

    int num;

    cout << endl;

    int collision\_number = 0;

    int opt;

    while (1) {

        cout << endl;

        cout << "Enter 1 to input" << endl;

        cout << "Enter 2 to display" << endl;

        cout << "Enter 3 to search a number" << endl;

        cout << "Enter 4 to delete a number" << endl;

        cin >> opt;

        cout << endl;

        switch (opt) {

        case 1:

        {

            cout << "Enter the number : ";

            cin >> num;

            cout << endl;

            int temp = num % 20;

            int collision\_number = 0;

            if (arr[temp] == -1) {

                arr[temp] = num;

            }

            else {

                int i = 0;

                while(i<20)

                {

                    temp = num % 20;

                    collision\_number++;

                    temp = (temp + collision\_number) % 20;

                    if (arr[temp] == -1) {

                        arr[temp] = num;

                        break;

                    }

                    else{

                        i++;

                    }

                }

            }

            break;

        }

        case 2:

        {

            int i = 0;

            while (i <= 19) {

                cout << arr[i] << "\t";

                i++;

            }

            i = 0;

            break;

        }

        case 3:

        {

            int num\_search;

            cout << "Enter the number you want to search : ";

            cin >> num\_search;

            cout << endl;

            int size = sizeof(arr) / sizeof(arr[0]);

            search(arr, num\_search, size-1);

            break;

        }

        case 4:

        {

            int num\_search;

            cout << "Enter the number you want to delete : ";

            cin >> num\_search;

            cout << endl;

            int size = sizeof(arr) / sizeof(arr[0]);

            int index = search(arr, num\_search, size-1);

            delete\_number(arr, index);

            break;

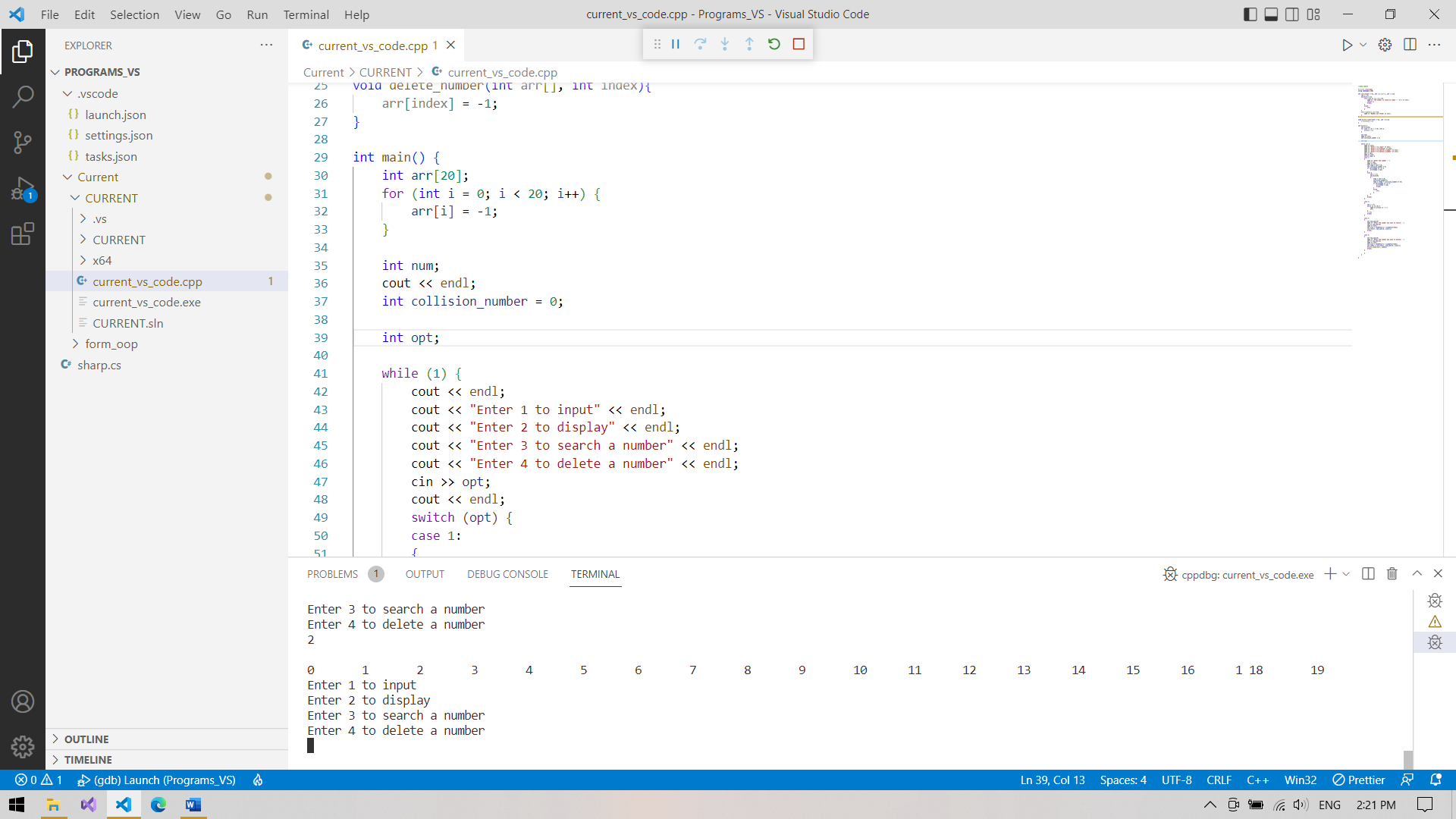
        }

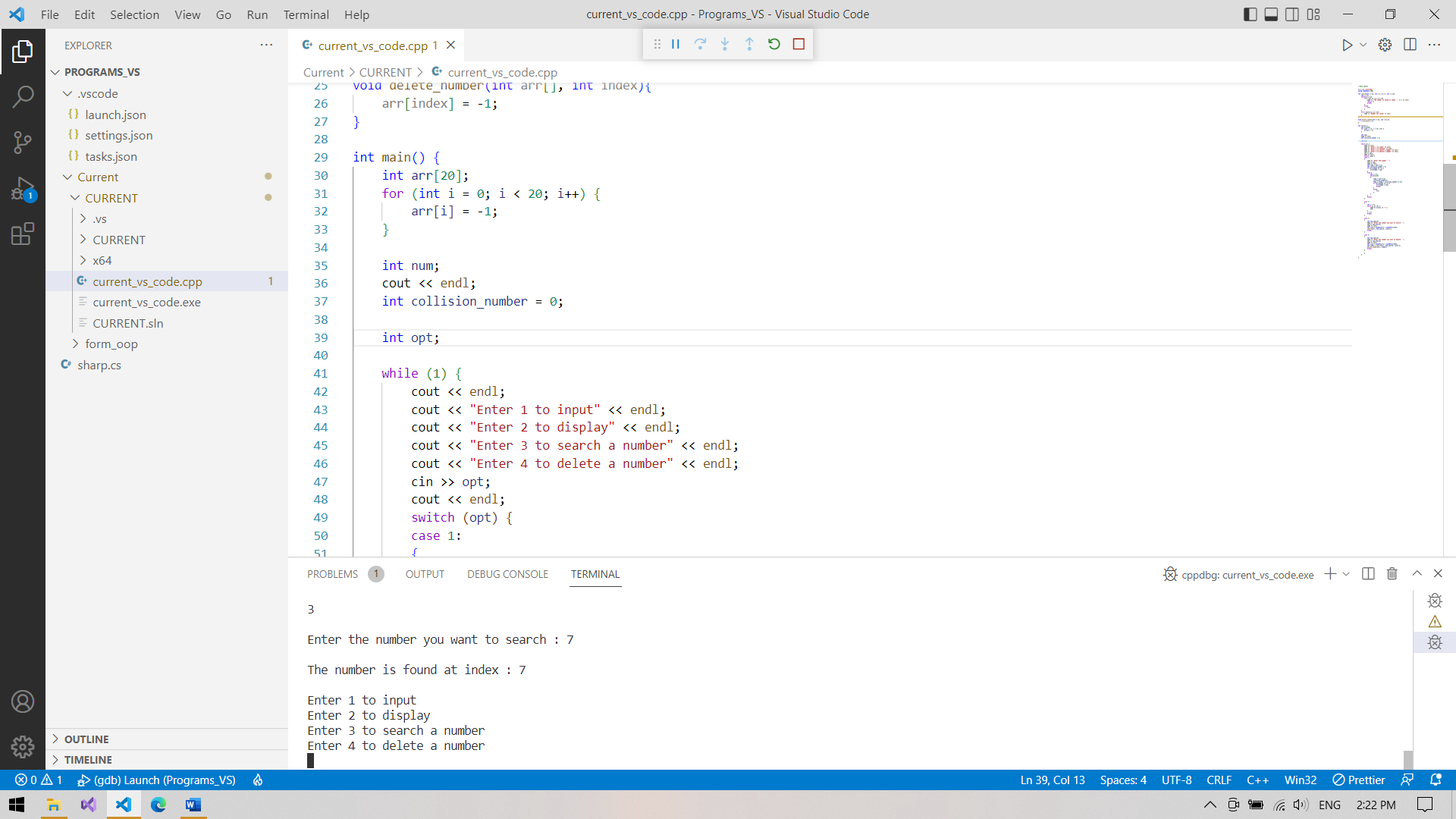
        }

    }

}

**INITIAL ARRAY:**



**SEARCHING 7:**

**DELETING 7:**

**-1 appeared instead of 7 because it is deleted.**

