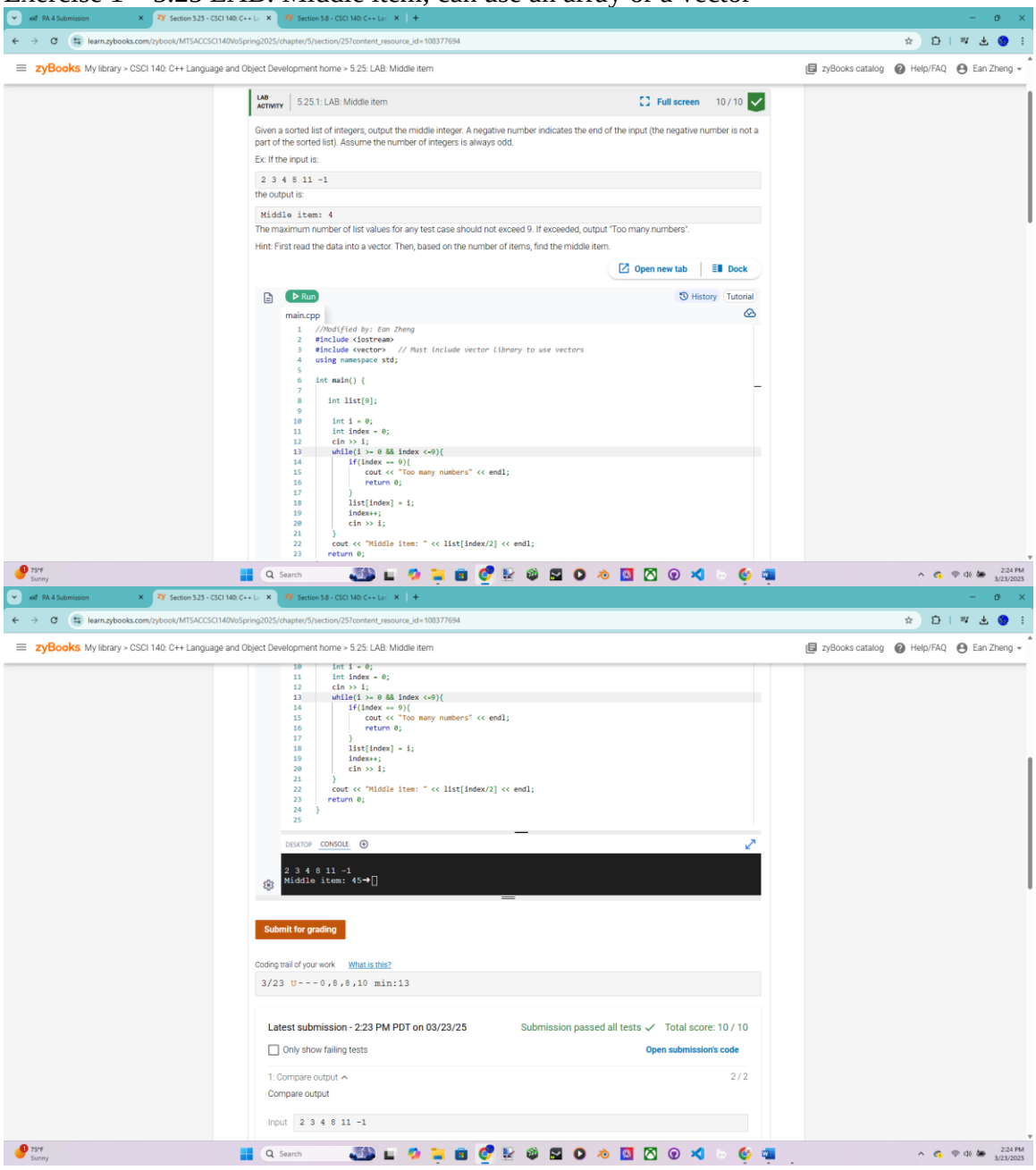
CSCI 140 PA 4 Submission

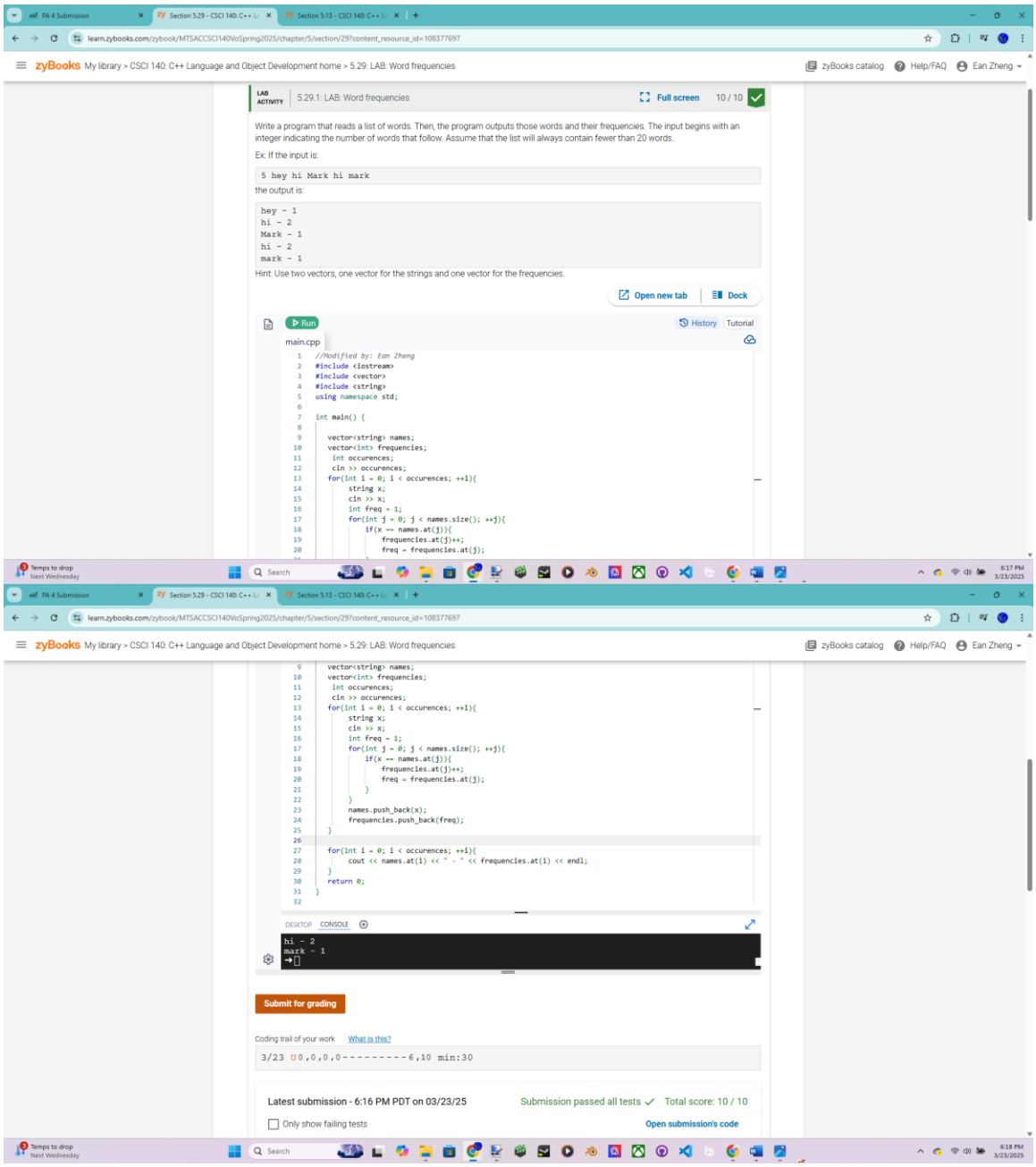
Due Date:3/24/2025 Late (date and time):_____

Name(s):Ean Zheng

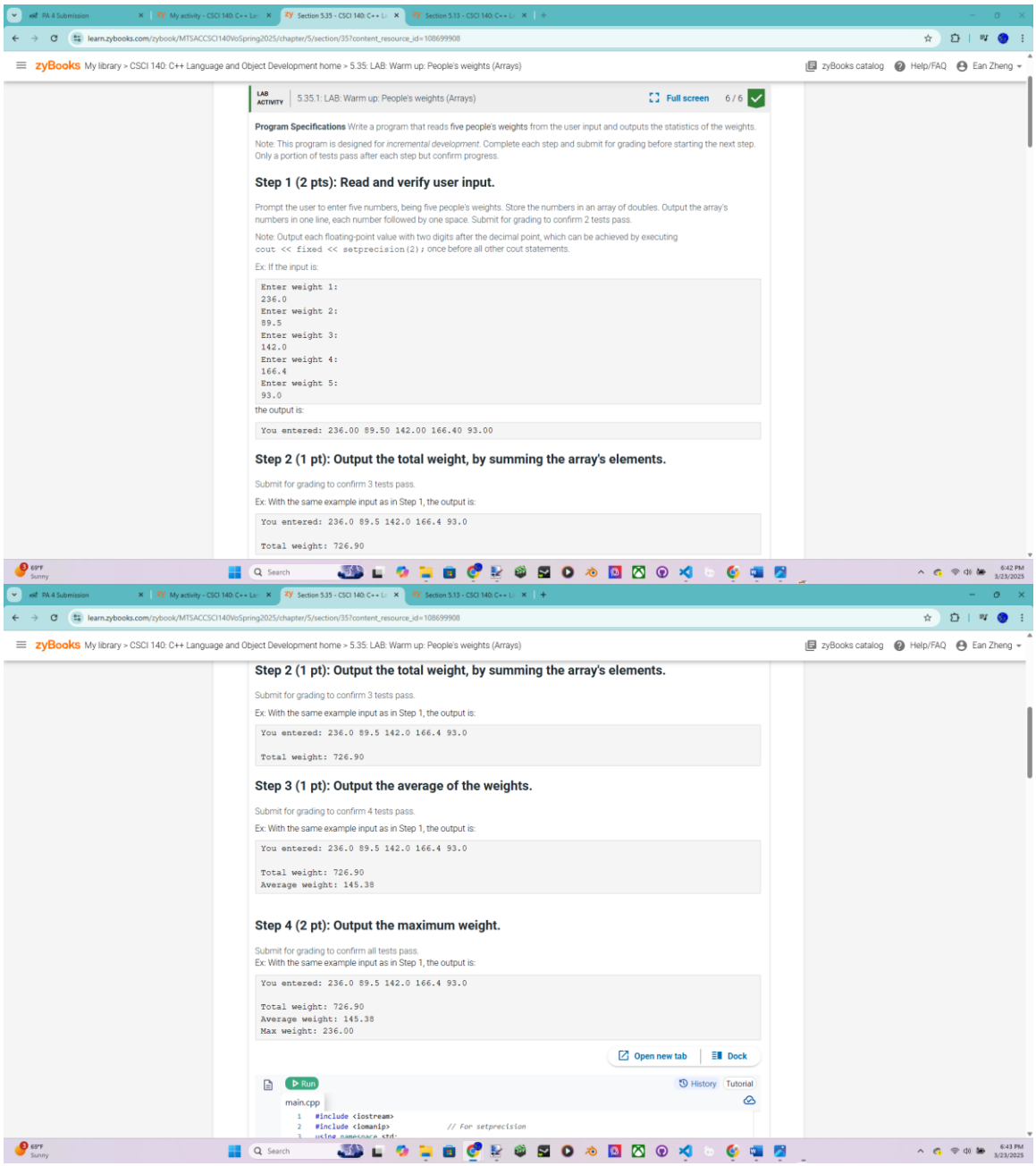
Exercise 1 – 5.25 LAB: Middle item; can use an array or a vector

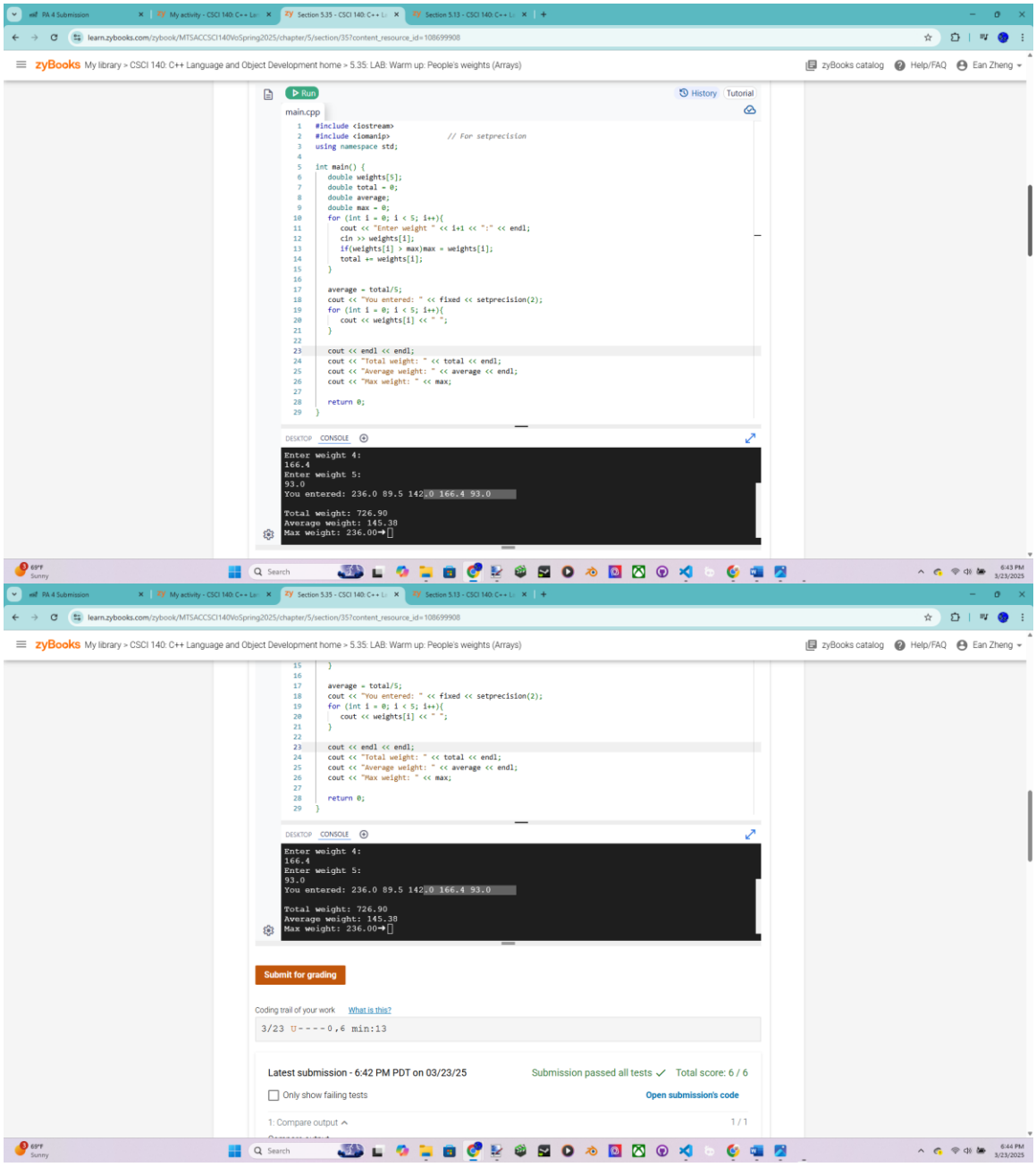


Exercise 2 – 5.29 LAB: Word frequencies



Exercise 3 – 5.35 LAB: Warm up: People's weights (Arrays)




zyBooks My library > CSCI 140: C++ Language and Object Development home > 5.35: LAB: Warm up: People's weights (Arrays)
zyBooks catalog
Help/FAQ
Ean Zheng
Run
History
Tutorial
main.cpp
#include <iostream>
#include <iomanip> // For setprecision
using namespace std;
int main() {
double weights[5];
double total = 0;
double average;
double max = 0;
for (int i = 0; i < 5; i++){
cout << "Enter weight " << i+1 << ":" << endl;
cin >> weights[i];
if(weights[i] > max)max = weights[i];
total += weights[i];
}
average = total/5;
cout << "You entered: " << fixed << setprecision(2);
for (int i = 0; i < 5; i++){
cout << weights[i] << " ";
}
cout << endl << endl;
cout << "Total weight: " << total << endl;
cout << "Average weight: " << average << endl;
cout << "Max weight: " << max;
return 0;
}
DESKTOP
CONSOLE
Enter weight 4:
166.4
Enter weight 5:
93.0
You entered: 236.0 89.5 142.0 166.4 93.0
Total weight: 726.90
Average weight: 145.38
Max weight: 236.00

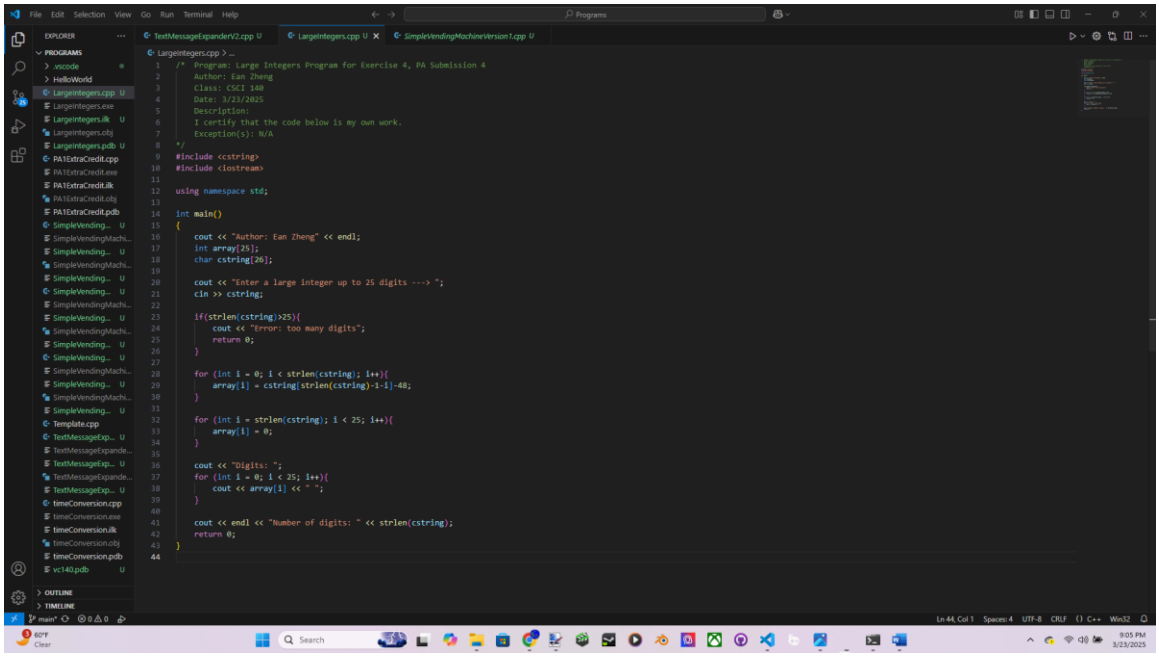
zyBooks My library > CSCI 140: C++ Language and Object Development home > 5.35: LAB: Warm up: People's weights (Arrays)
zyBooks catalog
Help/FAQ
Ean Zheng
Submit for grading
Coding trail of your work
What is this?
3/23 U----0,6 min:13
Latest submission - 6:42 PM PDT on 03/23/25
Submission passed all tests ✓
Total score: 6 / 6
Only show failing tests
Open submission's code
1: Compare output
1 / 1

Exercise 4 – Large Integers – more points for this exercise

If you need to work with very large integers such as applications in data encryption and decryption, you may not be able to use data type int or even the new long long int. A string or an array of integers can be used to store a large integer, but it is easier to perform basic arithmetic operations with an array or vector of integers. There are many ways you can store the digits in an array or vector: {1, 2, 3, 4, 0 ... 0}, {4, 3, 2, 1, 0 ... 0}, or {0 ... 0, 1, 2, 3, 4,}. For this exercise, the integer 1234 MUST be stored in an int array a or int vector a by setting a[0] to 4, a[1] to 3, a[2] to 2, and a[3] to 1 and the number of digits (4) can be stored in a separate variable (the second way). Unused digits will be initialized to 0 and you might need a variable such as numDigits or lastIndex to keep track of the number of actual digits.

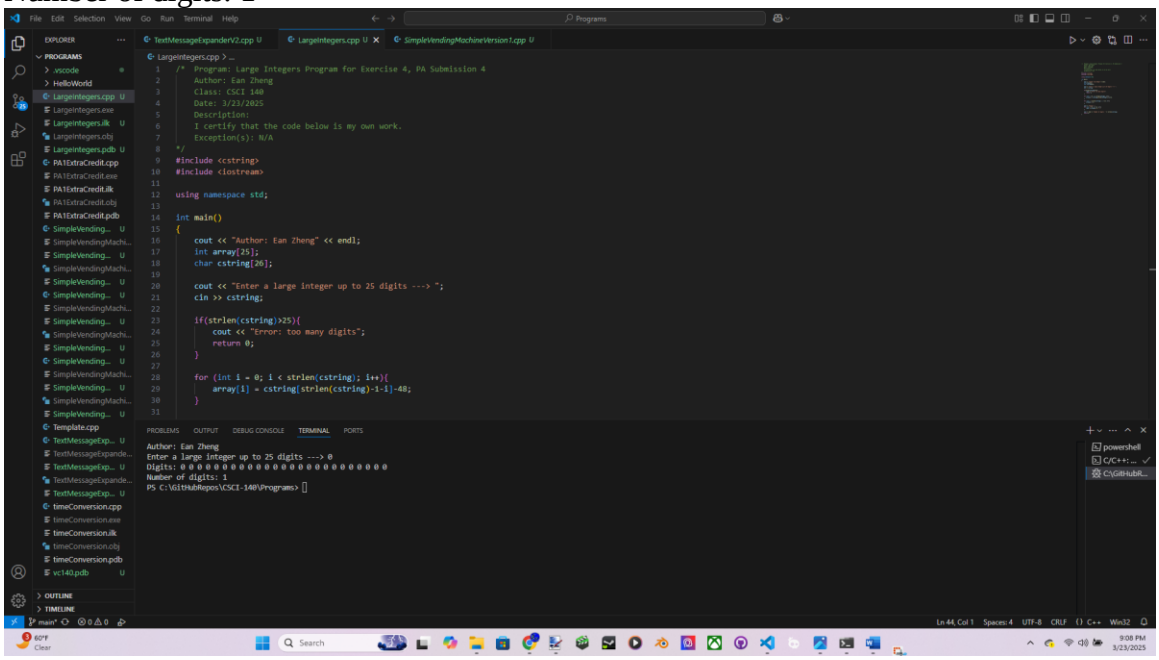
Source code below:

```
/*  Program: Large Integers Program for Exercise 4, PA Submission 4
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/23/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <cstring>
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int array[25];
    char cstring[26];

    cout << "Enter a large integer up to 25 digits ---> ";
    cin >> cstring;

    if(strlen(cstring)>25){
        cout << "Error: too many digits";
        return 0;
    }

    for (int i = 0; i < strlen(cstring); i++){
        array[i] = cstring[strlen(cstring)-1-i]-48;
    }

    for (int i = strlen(cstring); i < 25; i++){
        array[i] = 0;
    }

    cout << "Digits: ";
    for (int i = 0; i < 25; i++){
        cout << array[i] << " ";
    }

    cout << endl << "Number of digits: " << strlen(cstring);
    return 0;
}
```

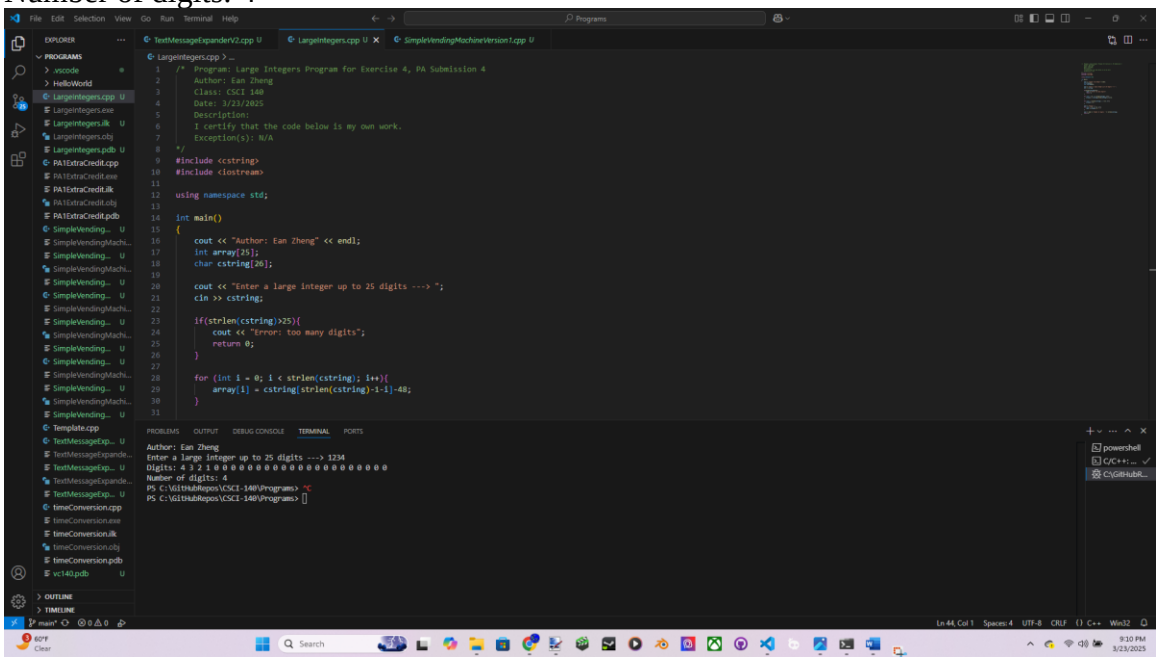

File Edit Selection View Go Run Terminal Help
Programs
EXPLORER
PROGRAMS
TextMessageExpanderV2.cpp U
LargeIntegers.cpp U
SimpleVendingMachineVersion1.cpp U
LargeIntegers.cpp > ...
/* Program: Large Integers Program for Exercise 4, PA Submission 4
Author: Ean Zheng
Class: CSCI 140
Date: 3/23/2025
Description:
I certify that the code below is my own work.
Exception(s): N/A
*/
#include <cstring>
#include <iostream>
using namespace std;
int main()
{
cout << "Author: Ean Zheng" << endl;
int array[25];
char cstring[26];
cout << "Enter a large integer up to 25 digits ---> ";
cin >> cstring;
if(strlen(cstring)>25){
cout << "Error: too many digits";
return 0;
}
for (int i = 0; i < strlen(cstring); i++){
array[i] = cstring[strlen(cstring)-1-i]-48;
}
for (int i = strlen(cstring); i < 25; i++){
array[i] = 0;
}
cout << "Digits: ";
for (int i = 0; i < 25; i++){
cout << array[i] << " ";
}
cout << endl << "Number of digits: " << strlen(cstring);
return 0;
}
OUTLINE
TIMELINE
Ln 44, Col 1 Spaces: 4 UTF-8 CRLF C++ Win32
Search
9:05 PM
3/23/2025

Input/output below:

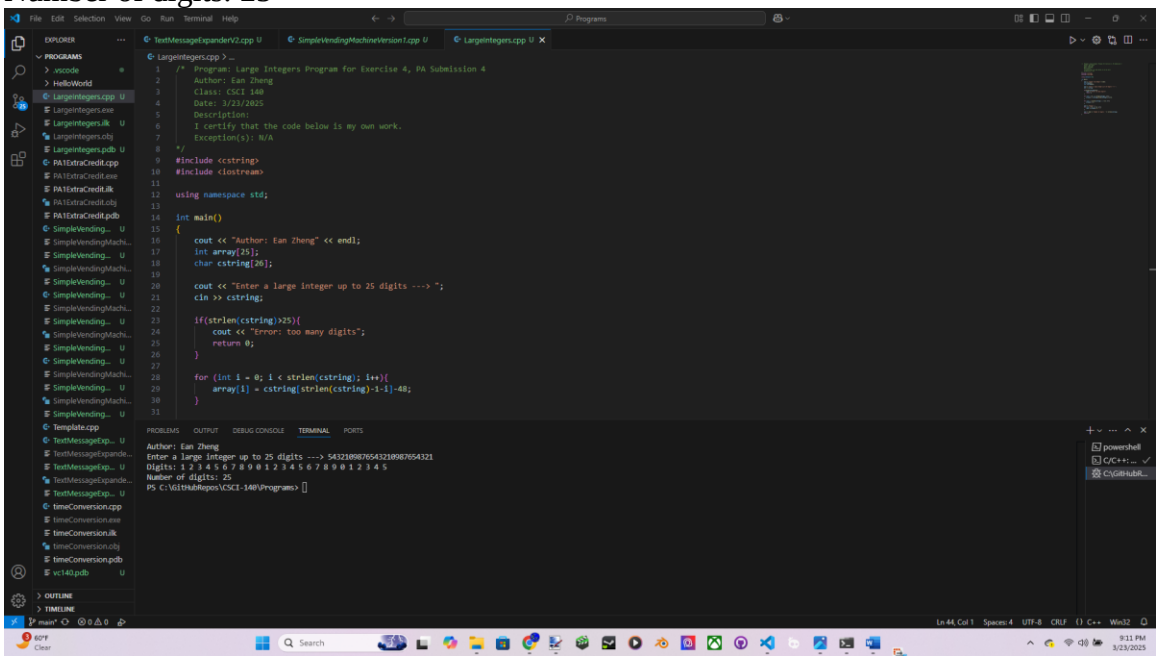
Author: Ean Zheng
Enter a large integer up to 25 digits ---> 0
Digits: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Number of digits: 1



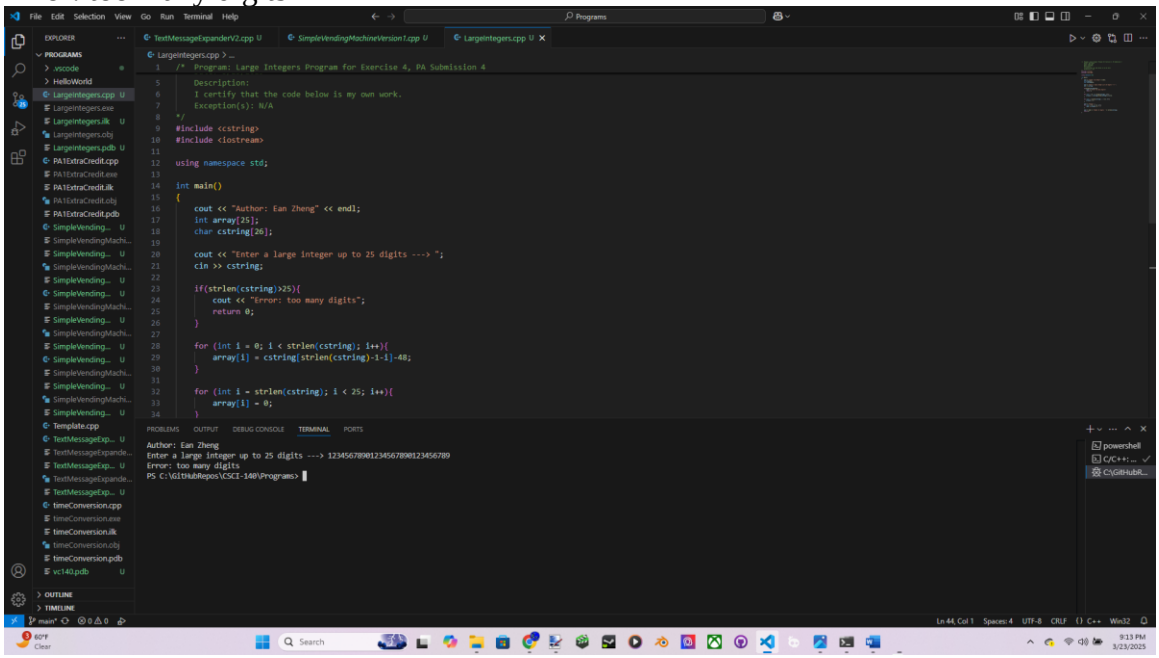
Author: Ean Zheng
Enter a large integer up to 25 digits ---> 1234
Digits: 4 3 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Number of digits: 4



Author: Ean Zheng
Enter a large integer up to 25 digits ---> 5432109876543210987654321
Digits: 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
Number of digits: 25



Author: Ean Zheng
Enter a large integer up to 25 digits ---> 12345678901234567890123456789
Error: too many digits



Question 1: Although C++ vectors can be more productive (getting things done quicker), list some good reasons for learning and using arrays instead of C++ vectors.

Arrays are more compact and use less space, and their set size are desirable for some cases. They also don't require additional stl headers, reducing compile time. Fixed size arrays can also be directly embedded, improving memory locality and reducing the amount of heap allocations that are needed.

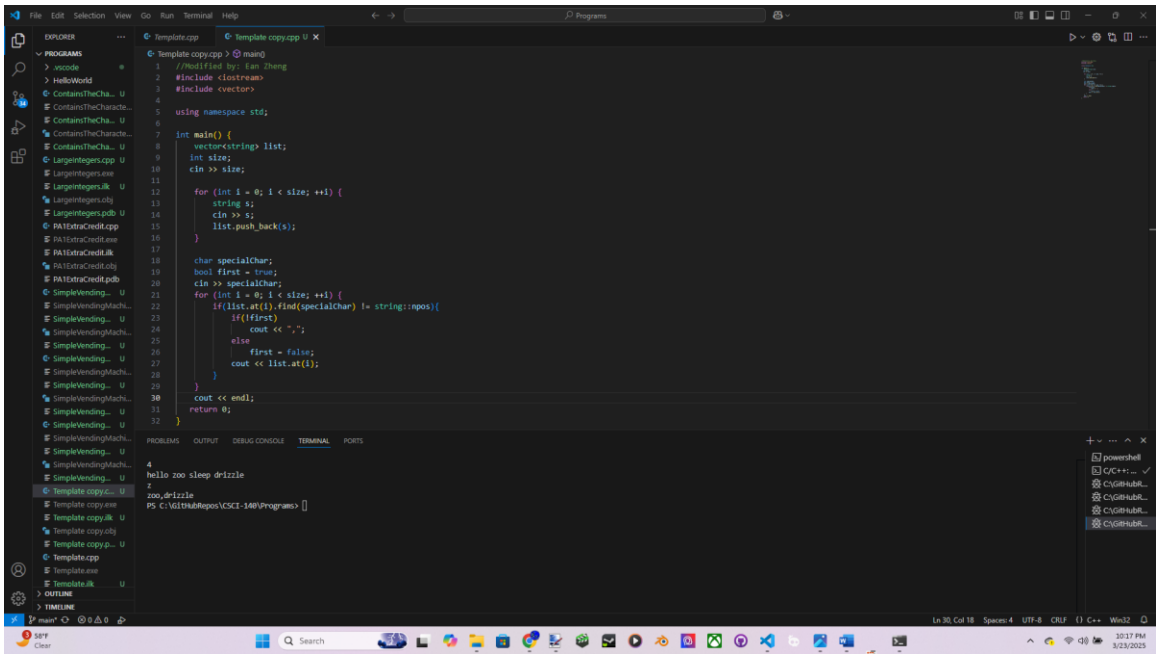
Question 2: Given two int arrays with the same size, arrX and arrY, you simply cannot copy array arrY to array arrX with "arrX = arrY;"? Explain why not. How would you go about copying arrY to arrX?

You can't copy one same-size array to another with a simple assignment because they have multiple values/instances. Simple assignment only works on two simple and exact instances, and don't work for multiple at once. I would copy arrY to arrX with a for loop that indexes and accesses the values of the two arrays, and assigns them each one by one until the number of iterations reaches the arrays' sizes.

Extra Credit (2 points): Start with 5.30 (LAB: Contains the character) in zyBook first and then modify it so that it does not display the trailing comma. You can copy the source code from zyBook to your development environment and then modify it. Make sure to provide the source code and sample input/output.
Ex: If the input is:
4
hello zoo sleep drizzle
z
then the output is:
zoo,drizzle

Source Code:

```
//Modified by: Ean Zheng
#include <iostream>
#include <vector>

using namespace std;

int main() {
    vector<string> list;
   int size;
   cin >> size;

    for (int i = 0; i < size; ++i) {
        string s;
        cin >> s;
        list.push_back(s);
    }

    char specialChar;
    bool first = true;
    cin >> specialChar;
    for (int i = 0; i < size; ++i) {
        if(list.at(i).find(specialChar) != string::npos){
            if(!first)
                cout << ",";
            else
                first = false;
            cout << list.at(i);
        }
    }
    cout << endl;
   return 0;
}
```

Input/Output:

4
hello zoo sleep drizzle
z
zoo,drizzle