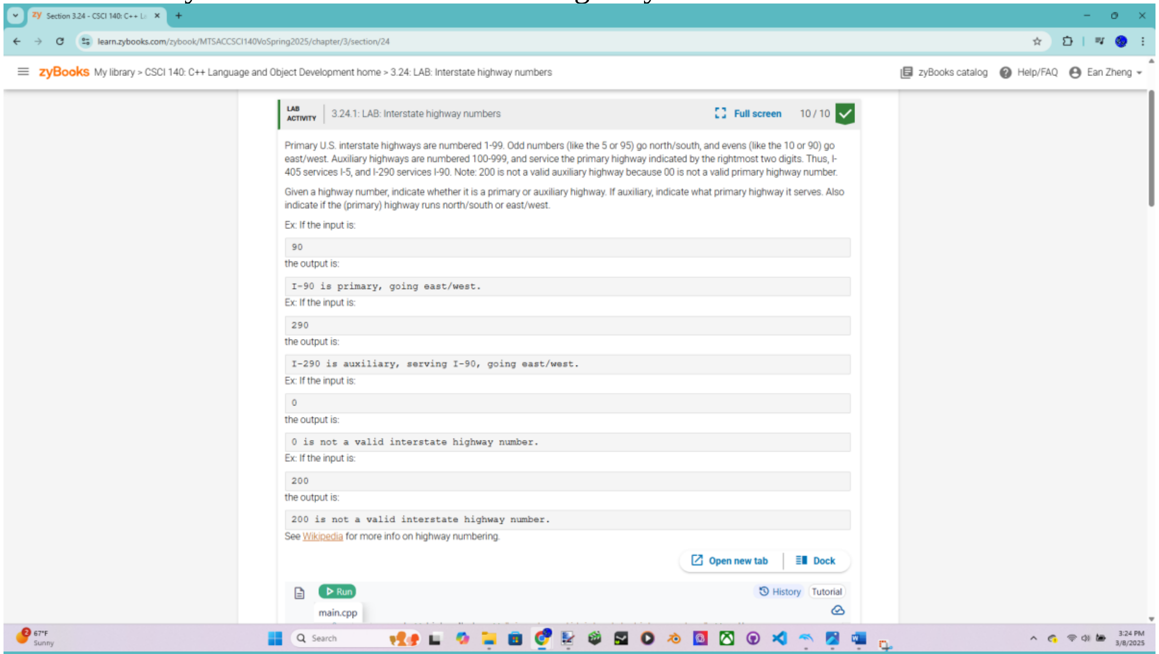
CSCI 140 PA 2 Submission

Due Date:3/10/2025 Late (date and time):_________________

Name: Ean Zheng

Exercise 1 – zyBook 3.24 LAB: Interstate highway numbers


Section 3.24 - CSCI 140: C++ L
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/24
zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.24: LAB: Interstate highway numbers
zyBooks catalog
Help/FAQ
Ean Zheng
LAB ACTIVITY
3.24.1: LAB: Interstate highway numbers
Full screen
10 / 10
Open new tab
Dock
Run
History
Tutorial
main.cpp
67°F Sunny
Search
3:24 PM 3/8/2025

Primary U.S. interstate highways are numbered 1-99. Odd numbers (like the 5 or 95) go north/south, and evens (like the 10 or 90) go east/west. Auxiliary highways are numbered 100-999, and service the primary highway indicated by the rightmost two digits. Thus, I-405 services I-5, and I-290 services I-90. Note: 200 is not a valid auxiliary highway because 00 is not a valid primary highway number.

Given a highway number, indicate whether it is a primary or auxiliary highway. If auxiliary, indicate what primary highway it serves. Also indicate if the (primary) highway runs north/south or east/west.

Ex: If the input is:

```
90
```

the output is:

```
I-90 is primary, going east/west.
```

Ex: If the input is:

```
290
```

the output is:

```
I-290 is auxiliary, serving I-90, going east/west.
```

Ex: If the input is:

```
0
```

the output is:

```
0 is not a valid interstate highway number.
```

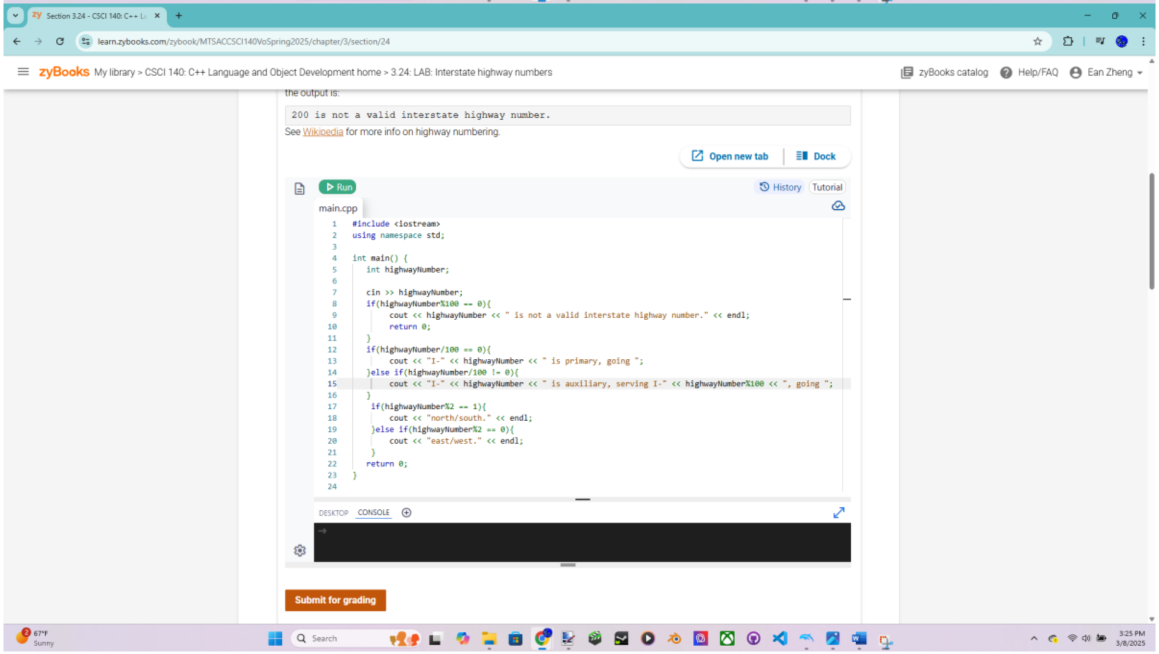
Ex: If the input is:

```
200
```

the output is:

```
200 is not a valid interstate highway number.
```

See Wikipedia for more info on highway numbering.

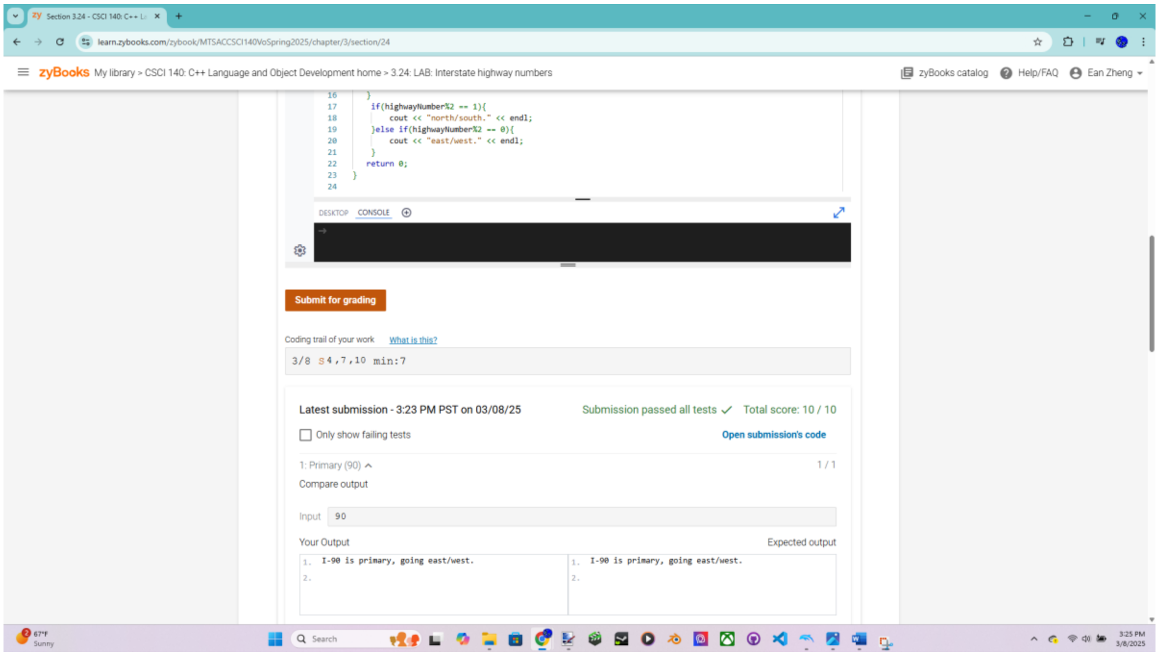

Section 3.24 - CSCI 140: C++ L
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/24
zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.24: LAB: Interstate highway numbers
zyBooks catalog
Help/FAQ
Ean Zheng
Open new tab
Dock
Run
History
Tutorial
main.cpp
DESKTOP
CONSOLE
Submit for grading
67°F Sunny
Search
3:25 PM 3/8/2025

the output is:

```
200 is not a valid interstate highway number.
```

See Wikipedia for more info on highway numbering.

```cpp
#include <iostream>
using namespace std;

int main() {
   int highwayNumber;

   cin >> highwayNumber;
   if(highwayNumber%100 == 0){
      cout << highwayNumber << " is not a valid interstate highway number." << endl;
      return 0;
   }
   if(highwayNumber/100 == 0){
      cout << "I-" << highwayNumber << " is primary, going ";
   }else if(highwayNumber/100 != 0){
      cout << "I-" << highwayNumber << " is auxiliary, serving I-" << highwayNumber%100 << ", going ";
   }
    if(highwayNumber%2 == 1){
      cout << "north/south." << endl;
    }else if(highwayNumber%2 == 0){
      cout << "east/west." << endl;
    }
   return 0;
}

```


Section 3.24 - CSCI 140: C++ L
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/24
zyBooks catalog
Help/FAQ
Ean Zheng
DESKTOP
CONSOLE
67°F
Sunny
Search
3:25 PM
3/8/2025

zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.24: LAB: Interstate highway numbers

```
    }
    if(highwayNumber%2 == 1){
        cout << "north/south." << endl;
    }else if(highwayNumber%2 == 0){
        cout << "east/west." << endl;
    }
   return 0;
}
```

Submit for grading

Coding trail of your work What is this?

```
3/8 S4,7,10 min:7
```

Latest submission - 3:23 PM PST on 03/08/25

Submission passed all tests ✓ Total score: 10 / 10

□ Only show failing tests

Open submission's code

1: Primary (90) ^ 1 / 1

Compare output

Input `90`

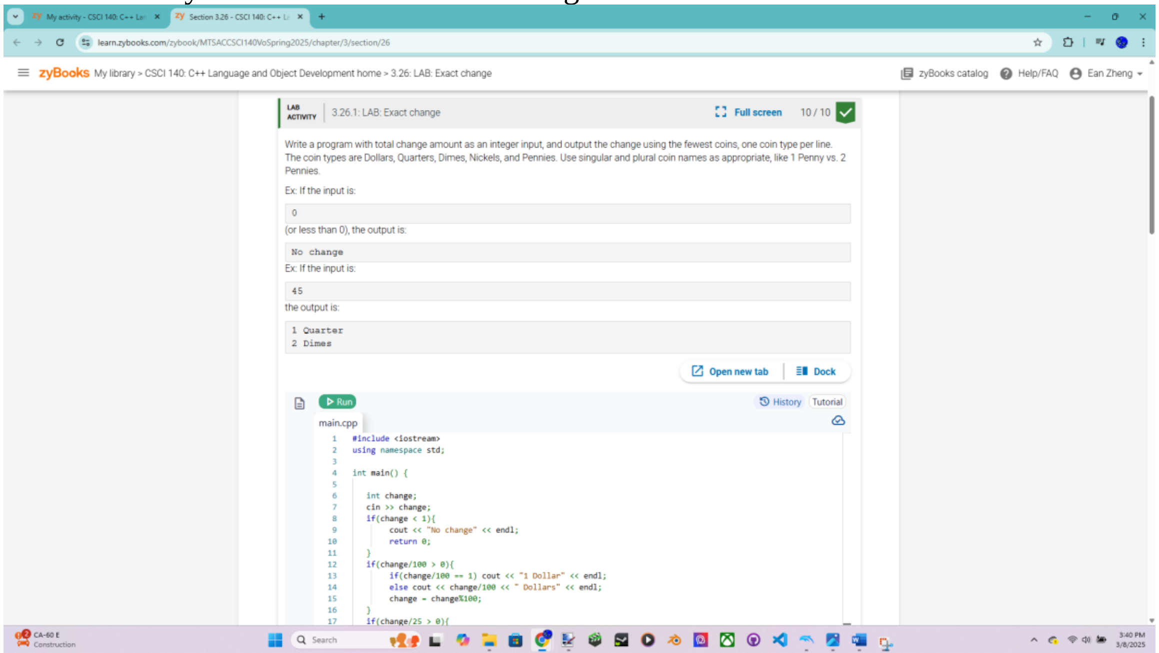
Your Output

```
1. I-90 is primary, going east/west.
2.
```

Expected output

```
1. I-90 is primary, going east/west.
2.
```

Exercise 2 – zyBook 3.26 LAB: Exact change


My activity - CSCI 140: C++ La
Section 3.26 - CSCI 140: C++ La
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/26
zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.26: LAB: Exact change
zyBooks catalog
Help/FAQ
Ean Zheng
LAB ACTIVITY
3.26.1: LAB: Exact change
Full screen
10 / 10
Open new tab
Dock
Run
History
Tutorial
main.cpp
Search
3:40 PM
3/8/2025

Write a program with total change amount as an integer input, and output the change using the fewest coins, one coin type per line. The coin types are Dollars, Quarters, Dimes, Nickels, and Pennies. Use singular and plural coin names as appropriate, like 1 Penny vs. 2 Pennies.

Ex: If the input is:

```
0
```

(or less than 0), the output is:

```
No change
```

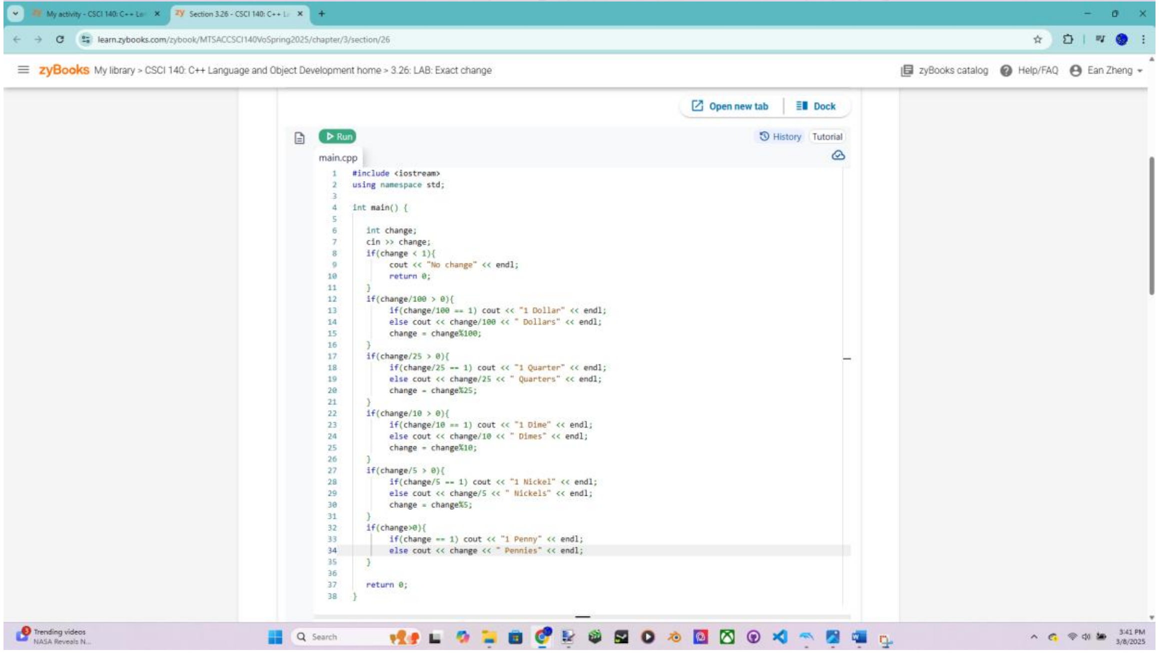
Ex: If the input is:

```
45
```

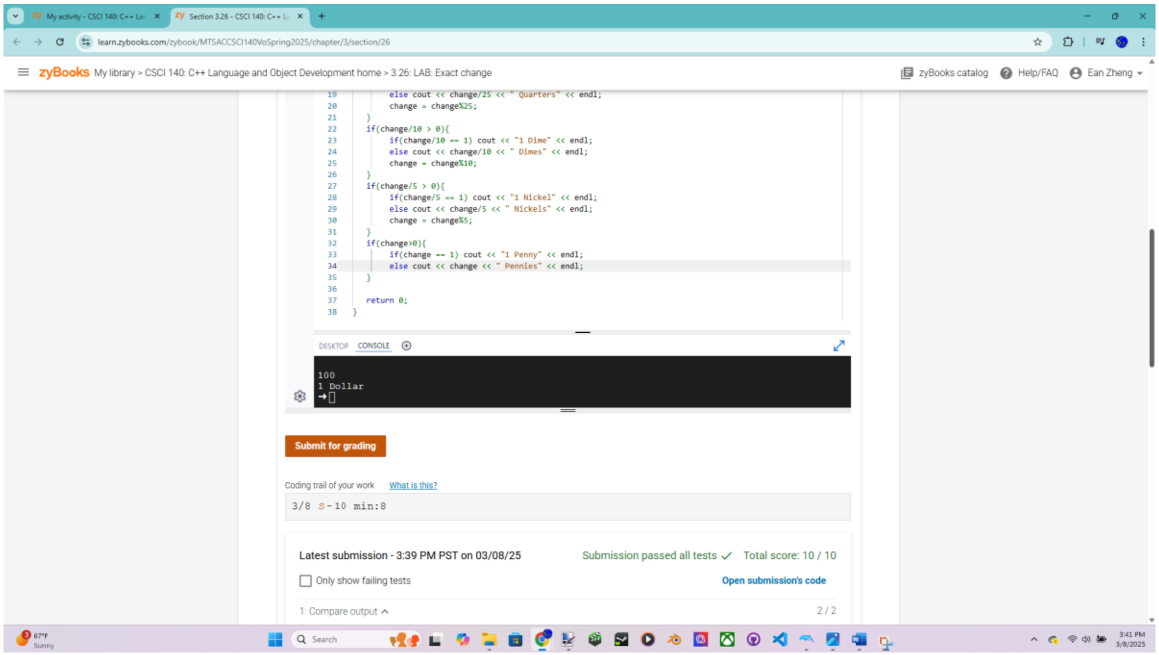
the output is:

```
1 Quarter
2 Dimes
```

```cpp
#include <iostream>
using namespace std;

int main() {

   int change;
   cin >> change;
   if(change < 1){
         cout << "No change" << endl;
         return 0;
   }
   if(change/100 > 0){
         if(change/100 == 1) cout << "1 Dollar" << endl;
         else cout << change/100 << " Dollars" << endl;
         change = change%100;
   }
   if(change/25 > 0){
```


My activity - CSCI 140: C++ La
Section 3.26 - CSCI 140: C++ La
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/26
zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.26: LAB: Exact change
zyBooks catalog
Help/FAQ
Ean Zheng
Open new tab
Dock
Run
History
Tutorial
main.cpp
Search
3:41 PM
3/8/2025

```cpp
#include <iostream>
using namespace std;

int main() {

   int change;
   cin >> change;
   if(change < 1){
         cout << "No change" << endl;
         return 0;
   }
   if(change/100 > 0){
         if(change/100 == 1) cout << "1 Dollar" << endl;
         else cout << change/100 << " Dollars" << endl;
         change = change%100;
   }
   if(change/25 > 0){
         if(change/25 == 1) cout << "1 Quarter" << endl;
         else cout << change/25 << " Quarters" << endl;
         change = change%25;
   }
   if(change/10 > 0){
         if(change/10 == 1) cout << "1 Dime" << endl;
         else cout << change/10 << " Dimes" << endl;
         change = change%10;
   }
   if(change/5 > 0){
         if(change/5 == 1) cout << "1 Nickel" << endl;
         else cout << change/5 << " Nickels" << endl;
         change = change%5;
   }
   if(change>0){
         if(change == 1) cout << "1 Penny" << endl;
         else cout << change << " Pennies" << endl;
   }

   return 0;
}
```


My activity - CSCI 140: C++ La
Section 3.26 - CSCI 140: C++ L
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/26
zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.26: LAB: Exact change
zyBooks catalog
Help/FAQ
Ean Zheng

```
            else cout << change/25 << " Quarters" << endl;
            change = change%25;
        }
        if(change/10 > 0){
            if(change/10 == 1) cout << "1 Dime" << endl;
            else cout << change/10 << " Dimes" << endl;
            change = change%10;
        }
        if(change/5 > 0){
            if(change/5 == 1) cout << "1 Nickel" << endl;
            else cout << change/5 << " Nickels" << endl;
            change = change%5;
        }
        if(change>0){
            if(change == 1) cout << "1 Penny" << endl;
            else cout << change << " Pennies" << endl;
        }

        return 0;
    }
```

```
100
1 Dollar
→
```

Coding trail of your work What is this?

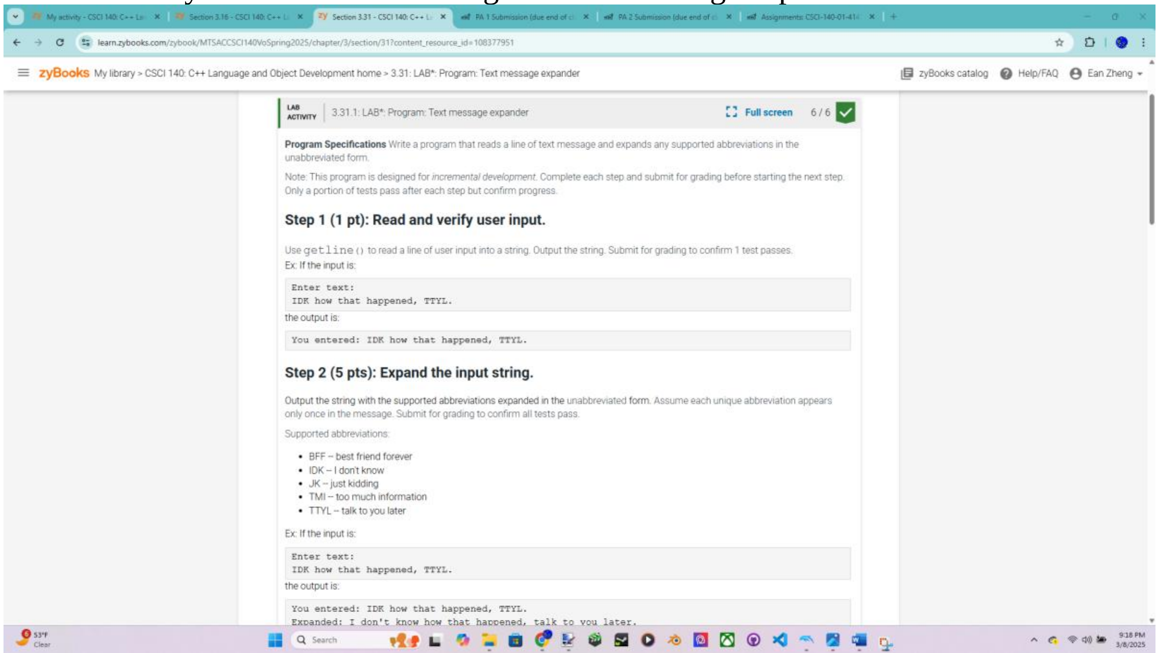
3/8 S-10 min:8

Latest submission - 3:39 PM PST on 03/08/25 Submission passed all tests ✓ Total score: 10 / 10

□ Only show failing tests

1: Compare output 2 / 2

Exercise 3 – zyBook 3.31 LAB*: Program: Text message expander


My activity - CSCI 140: C++ La
Section 3.16 - CSCI 140: C++ La
Section 3.31 - CSCI 140: C++ La
PA 1 Submission (due end of cl
PA 2 Submission (due end of cl
Assignments: CSCI-140-01-414
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/31?content_resource_id=108377951
zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.31: LAB*: Program: Text message expander
zyBooks catalog
Help/FAQ
Ean Zheng
LAB ACTIVITY
3.31.1: LAB*: Program: Text message expander
Full screen
6 / 6
53°F
Clear
Search
9:18 PM
3/8/2025

Program Specifications Write a program that reads a line of text message and expands any supported abbreviations in the unabbreviated form.

Note: This program is designed for *incremental development*. Complete each step and submit for grading before starting the next step. Only a portion of tests pass after each step but confirm progress.

Step 1 (1 pt): Read and verify user input.

Use `getline()` to read a line of user input into a string. Output the string. Submit for grading to confirm 1 test passes.

Ex: If the input is:

```
Enter text:
IDK how that happened, TTYL.
```

the output is:

```
You entered: IDK how that happened, TTYL.
```

Step 2 (5 pts): Expand the input string.

Output the string with the supported abbreviations expanded in the unabbreviated form. Assume each unique abbreviation appears only once in the message. Submit for grading to confirm all tests pass.

Supported abbreviations:

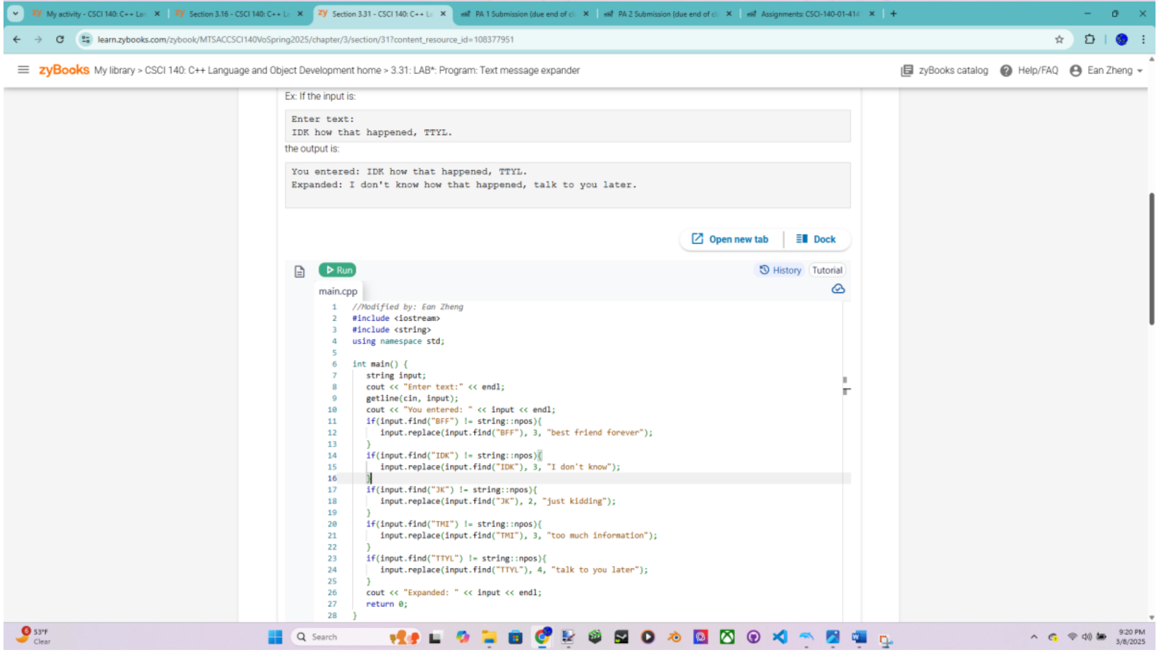
- BFF – best friend forever
- IDK – I don't know
- JK – just kidding
- TMI – too much information
- TTYL – talk to you later

Ex: If the input is:

```
Enter text:
IDK how that happened, TTYL.
```

the output is:

```
You entered: IDK how that happened, TTYL.
Expanded: I don't know how that happened, talk to you later.
```


My activity - CSCI 140: C++ La
Section 3.16 - CSCI 140: C++ La
Section 3.31 - CSCI 140: C++ La
PA 1 Submission (due end of cl
PA 2 Submission (due end of cl
Assignments: CSCI-140-01-414
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/31?content_resource_id=108377951
zyBooks My library > CSCI 140: C++ Language and Object Development home > 3.31: LAB*: Program: Text message expander
zyBooks catalog
Help/FAQ
Ean Zheng
Open new tab
Dock
Run
History
Tutorial
main.cpp
53°F
Clear
Search
9:20 PM
3/8/2025

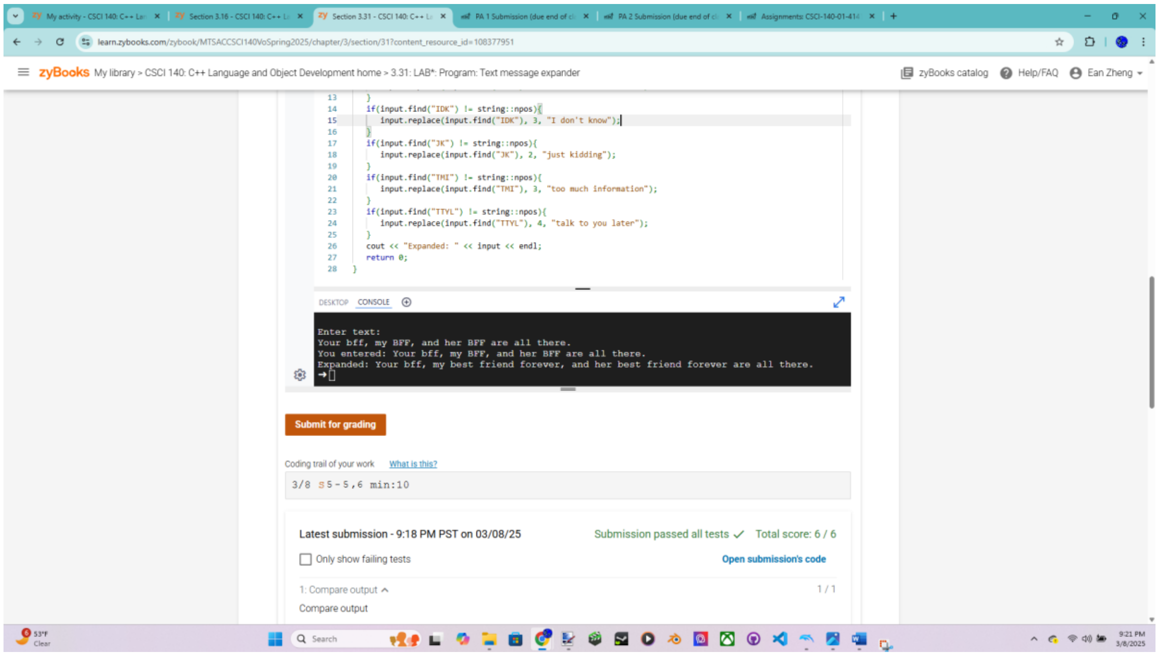
Ex: If the input is:

```
Enter text:
IDK how that happened, TTYL.
```

the output is:

```
You entered: IDK how that happened, TTYL.
Expanded: I don't know how that happened, talk to you later.
```

```cpp
//Modified by: Ean Zheng
#include <iostream>
#include <string>
using namespace std;

int main() {
   string input;
   cout << "Enter text:" << endl;
   getline(cin, input);
   cout << "You entered: " << input << endl;
   if(input.find("BFF") != string::npos){
      input.replace(input.find("BFF"), 3, "best friend forever");
   }
   if(input.find("IDK") != string::npos){
      input.replace(input.find("IDK"), 3, "I don't know");
   }
   if(input.find("JK") != string::npos){
      input.replace(input.find("JK"), 2, "just kidding");
   }
   if(input.find("TMI") != string::npos){
      input.replace(input.find("TMI"), 3, "too much information");
   }
   if(input.find("TTYL") != string::npos){
      input.replace(input.find("TTYL"), 4, "talk to you later");
   }
   cout << "Expanded: " << input << endl;
   return 0;
}
```



My activity - CSCI 140: C++ La
Section 3.16 - CSCI 140: C++ L
Section 3.31 - CSCI 140: C++ L
PA 1 Submission (due end of cl
PA 2 Submission (due end of cl
Assignments: CSCI-140-01-414
learn.zybooks.com/zybook/MTSACCSCI140VoSpring2025/chapter/3/section/31?content_resource_id=108377951
zyBooks
My library > CSCI 140: C++ Language and Object Development home > 3.31: LAB*: Program: Text message expander
zyBooks catalog
Help/FAQ
Ean Zheng

```
   }
   if(input.find("IDK") != string::npos){
      input.replace(input.find("IDK"), 3, "I don't know");
   }
   if(input.find("JK") != string::npos){
      input.replace(input.find("JK"), 2, "just kidding");
   }
   if(input.find("TMI") != string::npos){
      input.replace(input.find("TMI"), 3, "too much information");
   }
   if(input.find("TTYL") != string::npos){
      input.replace(input.find("TTYL"), 4, "talk to you later");
   }
   cout << "Expanded: " << input << endl;
   return 0;
}
```

```
Enter text:
Your bff, my BFF, and her BFF are all there.
You entered: Your bff, my BFF, and her BFF are all there.
Expanded: Your bff, my best friend forever, and her best friend forever are all there.
➜
```

Coding trail of your work What is this?

```
3/8 S5-5,6 min:10
```

Latest submission - 9:18 PM PST on 03/08/25 Submission passed all tests ✓ Total score: 6 / 6

□ Only show failing tests

1: Compare output ^ 1 / 1

Compare output

Exercise 4 – Simple Vending Machine Version 1 – more points for this exercise

Your program is supposed to read an integer value between 0 and 100 (inclusive), representing the amount of a purchase in cents from a vending machine. Produce an error message if the input value is not in that range. If the input is valid, determine the amount of change that would be returned from one dollar, and print the number of quarters, dimes, and nickels. Since pennies are not available, round the changes to the nearest multiple of 5 as needed. Assume that there are exactly 2 quarters, 2 dimes, and 2 nickels in the machine at the beginning. It is important to maximize the coins with the highest value first and utilize the next denomination if the current denomination is exhausted. Reject a valid purchase if it cannot be processed (i.e., not enough available coins to make the change). Follow the format below and you must plan and write down the pseudocode before attempting your code on the computer.

Pseudocode below if applicable:

Initiate purchaseamount and change int variables, and initiate quarters, dimes, and nickels int variables with value 0.

Announce program name and author, say default amounts of quarters, dimes, and nickles, add a blank line, and prompt user to input value with valid range.

When user inputs value, assign purchaseamount to inputed value and show it.

Check if purchaseamount isn't in range, if so, cout invalid amount error and return 0.

Assign change to 100 minus input.

If modulo of change by 5 isn't 0, print change variable, then see if modulo of change by 5 is less than or equal to 2, or greater than or equal to 3. If less than or equal to 2, subtract modulo of change by 5 from change. If greater than or equal to 5, add 5 subtracted by module of change by 5 to change. (module of change by 5 is parenthesized to assure correct operation.) After that, print current change variable as rounded change.

Initiate remainingchange variable as change

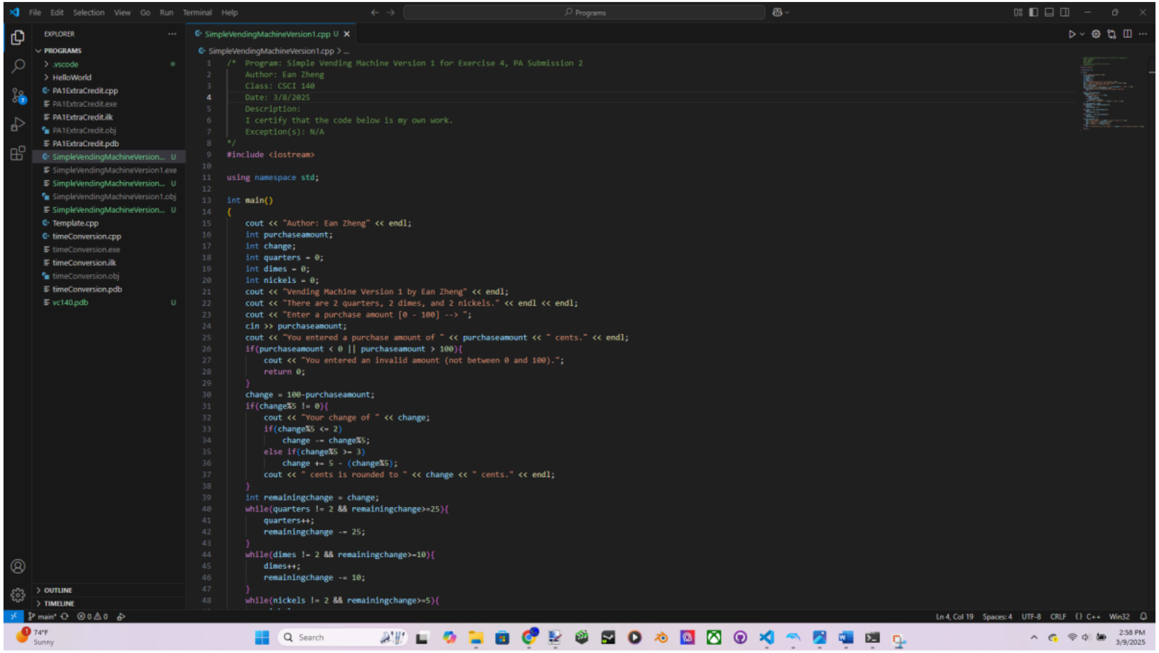
Make while loop for while quarters isn't 2 and while remainingchange is bigger than or equal to 25, in the loop add 1 to quarter then subtract 25 from remainingchange

Do same while loops 2 times except for dimes and nickels. Compare remainingchange bigger than or equal to 10 in while condition and subtract 10 from remainingchange inside loop for dimes. Compare remainingchange bigger than or equal to 5 in while condition and subtract 5 from remainingchange inside loop for nickels.

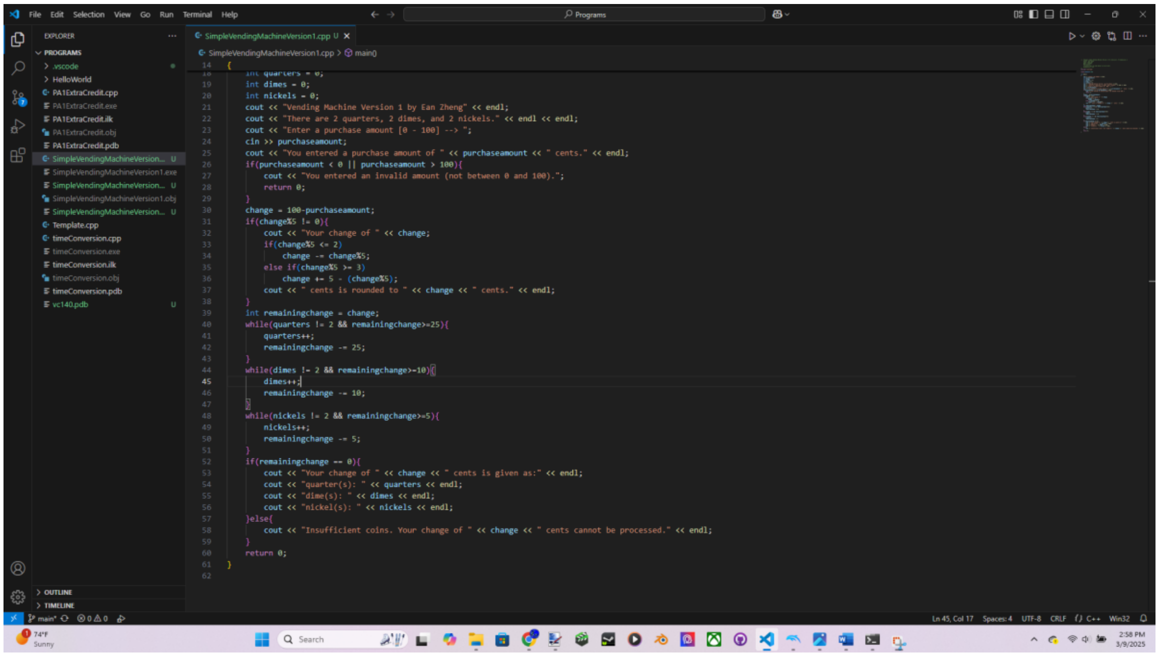
Check if remainingchange is 0. If so, print change variable, and that it is given as the quarters, dimes, and nickels variables in separate lines. If not, print insufficient coins meesage with change variable, and that it can’t be processed.

Source code below:

```
/*  Program: Simple Vending Machine Version 1 for Exercise 4, PA Submission 2
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/8/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int purchaseamount;
    int change;
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
        if(change%5 <= 2)
            change -= change%5;
        else if(change%5 >= 3)
            change += 5 - (change%5);
        cout << " cents is rounded to " << change << " cents." << endl;
    }
    int remainingchange = change;
    while(quarters != 2 && remainingchange>=25){
        quarters++;
        remainingchange -= 25;
    }
    while(dimes != 2 && remainingchange>=10){
        dimes++;
        remainingchange -= 10;
    }
    while(nickels != 2 && remainingchange>=5){
        nickels++;
        remainingchange -= 5;
    }
    if(remainingchange == 0){
        cout << "Your change of " << change << " cents is given as:" << endl;
        cout << "quarter(s): " << quarters << endl;
        cout << "dime(s): " << dimes << endl;
        cout << "nickel(s): " << nickels << endl;
    }else{
        cout << "Insufficient coins. Your change of " << change << " cents cannot be processed." << endl;
    }
    return 0;
}
```



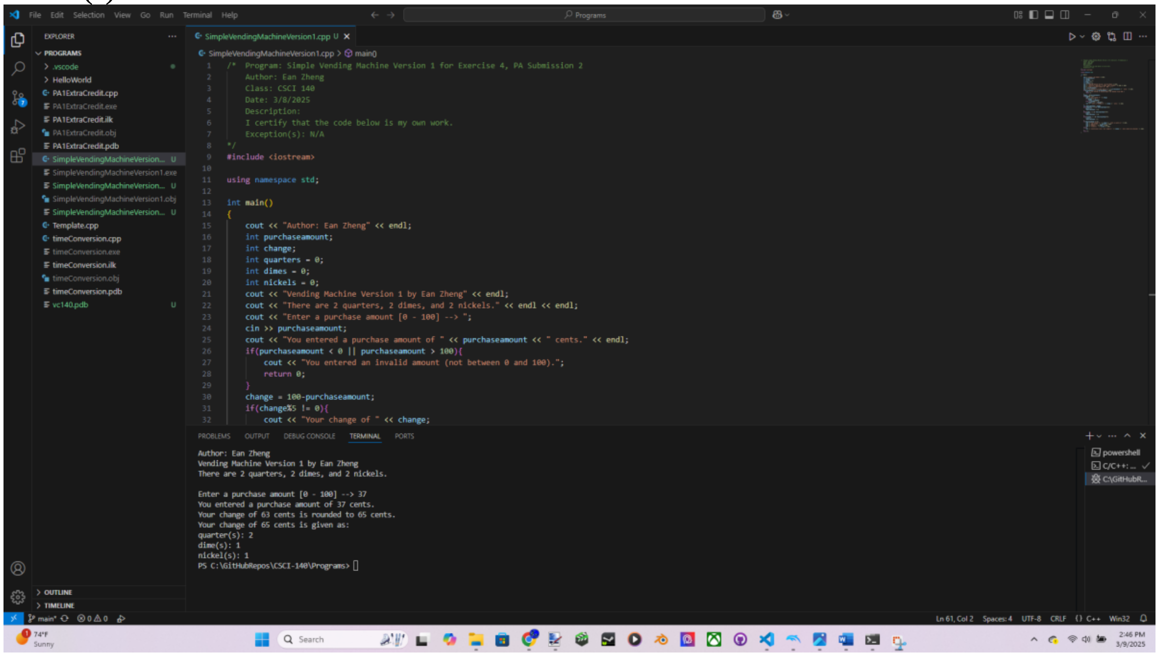
```
/*  Program: Simple Vending Machine Version 1 for Exercise 4, PA Submission 2
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/8/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int purchaseamount;
    int change;
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
        if(change%5 <= 2)
            change -= change%5;
        else if(change%5 >= 3)
            change += 5 - (change%5);
        cout << " cents is rounded to " << change << " cents." << endl;
    }
    int remainingchange = change;
    while(quarters != 2 && remainingchange>=25){
        quarters++;
        remainingchange -= 25;
    }
    while(dimes != 2 && remainingchange>=10){
        dimes++;
        remainingchange -= 10;
    }
    while(nickels != 2 && remainingchange>=5){
```



```cpp
{
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
        if(change%5 <= 2)
            change -= change%5;
        else if(change%5 >= 3)
            change += 5 - (change%5);
        cout << " cents is rounded to " << change << " cents." << endl;
    }
    int remainingchange = change;
    while(quarters != 2 && remainingchange>=25){
        quarters++;
        remainingchange -= 25;
    }
    while(dimes != 2 && remainingchange>=10){
        dimes++;
        remainingchange -= 10;
    }
    while(nickels != 2 && remainingchange>=5){
        nickels++;
        remainingchange -= 5;
    }
    if(remainingchange == 0){
        cout << "Your change of " << change << " cents is given as:" << endl;
        cout << "quarter(s): " << quarters << endl;
        cout << "dime(s): " << dimes << endl;
        cout << "nickel(s): " << nickels << endl;
    }else{
        cout << "Insufficient coins. Your change of " << change << " cents cannot be processed." << endl;
    }
    return 0;
}
```

Input/output below:

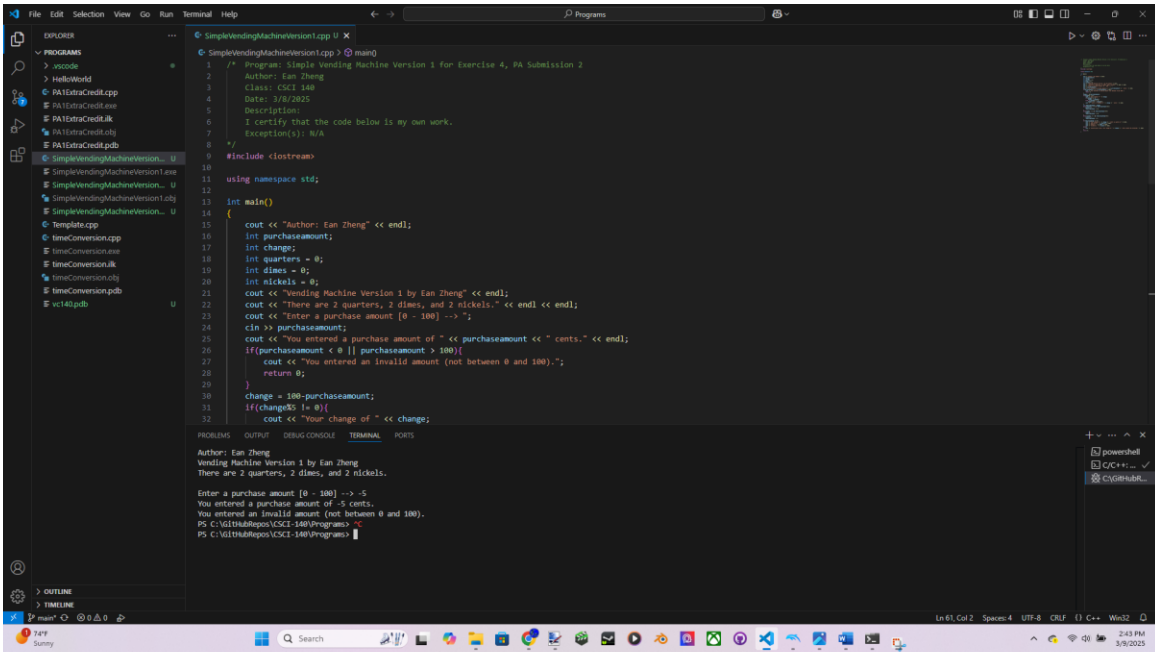
```
Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 37
You entered a purchase amount of 37 cents.
Your change of 63 cents is rounded to 65 cents.
Your change of 65 cents is given as:
quarter(s): 2
dime(s): 1
nickel(s): 1
```



```cpp
/*  Program: Simple Vending Machine Version 1 for Exercise 4, PA Submission 2
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/8/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int purchaseamount;
    int change;
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
```

```
Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 37
You entered a purchase amount of 37 cents.
Your change of 63 cents is rounded to 65 cents.
Your change of 65 cents is given as:
quarter(s): 2
dime(s): 1
nickel(s): 1
PS C:\GitHubRepos\CSCI-140\Programs> 
```

```
Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> -5
You entered a purchase amount of -5 cents.
You entered an invalid amount (not between 0 and 100).
```

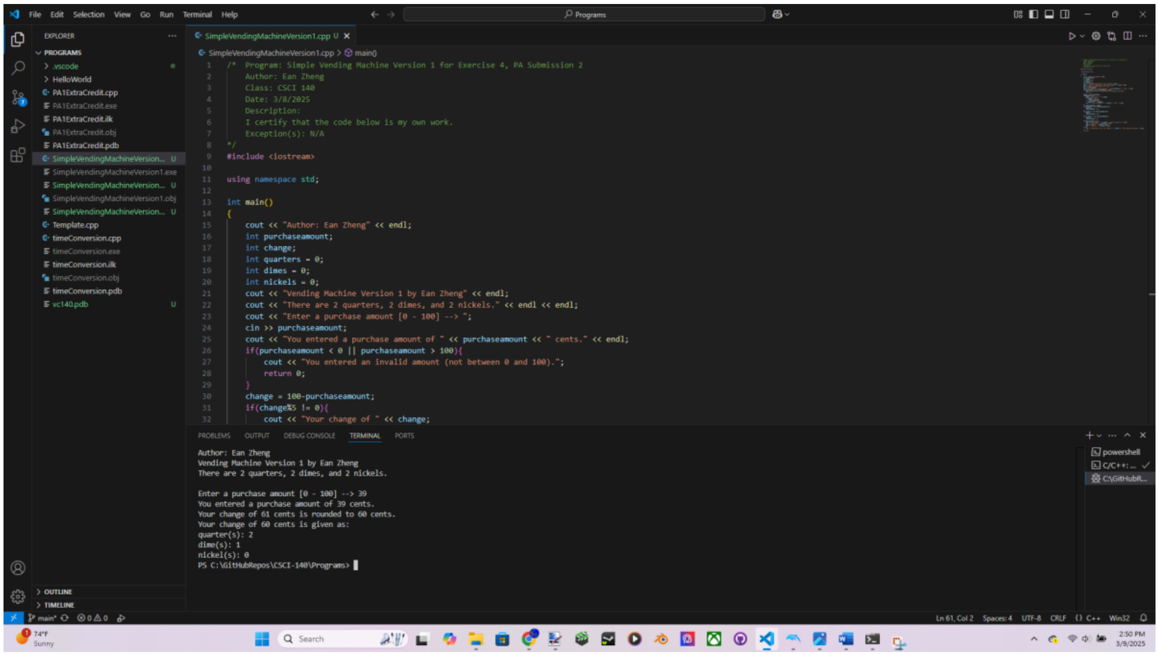


```
/*  Program: Simple Vending Machine Version 1 for Exercise 4, PA Submission 2
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/8/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int purchaseamount;
    int change;
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
```

```
Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> -5
You entered a purchase amount of -5 cents.
You entered an invalid amount (not between 0 and 100).
PS C:\GitHubRepos\CSCI-140\Programs> ^C
PS C:\GitHubRepos\CSCI-140\Programs>
```

Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 39
You entered a purchase amount of 39 cents.
Your change of 61 cents is rounded to 60 cents.
Your change of 60 cents is given as:
quarter(s): 2
dime(s): 1
nickel(s): 0

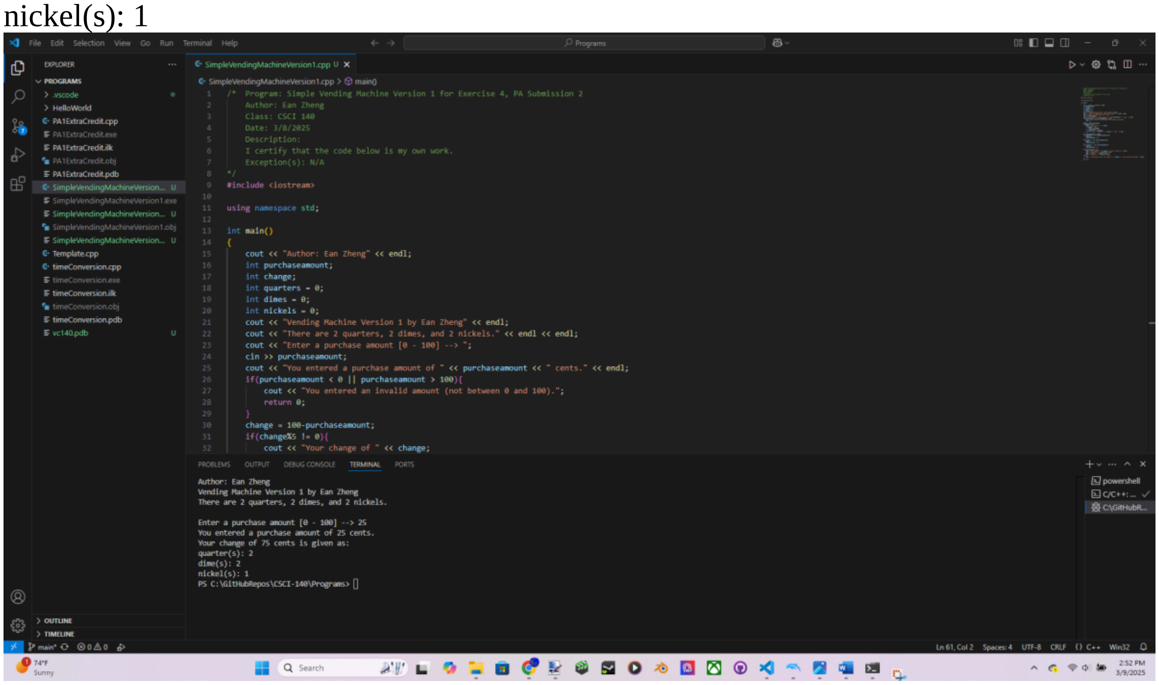


```
/*  Program: Simple Vending Machine Version 1 for Exercise 4, PA Submission 2
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/8/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int purchaseamount;
    int change;
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
```

```
Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 39
You entered a purchase amount of 39 cents.
Your change of 61 cents is rounded to 60 cents.
Your change of 60 cents is given as:
quarter(s): 2
dime(s): 1
nickel(s): 0
PS C:\GitHubRepos\CSCI-140\Programs>
```

Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 25
You entered a purchase amount of 25 cents.
Your change of 75 cents is given as:
quarter(s): 2
dime(s): 2
nickel(s): 1

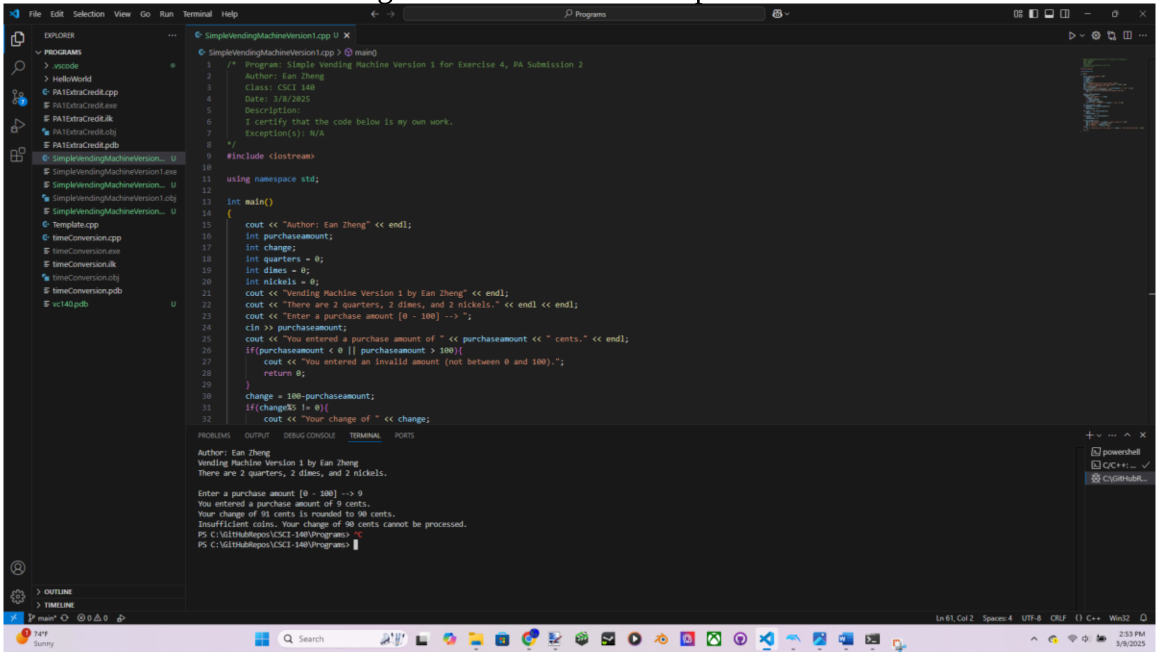


```
/*  Program: Simple Vending Machine Version 1 for Exercise 4, PA Submission 2
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/8/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int purchaseamount;
    int change;
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
```

```
Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 25
You entered a purchase amount of 25 cents.
Your change of 75 cents is given as:
quarter(s): 2
dime(s): 2
nickel(s): 1
PS C:\GitHubRepos\CSCI-140\Programs> []
```

Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 9
You entered a purchase amount of 9 cents.
Your change of 91 cents is rounded to 90 cents.
Insufficient coins. Your change of 90 cents cannot be processed.



```cpp
/*  Program: Simple Vending Machine Version 1 for Exercise 4, PA Submission 2
    Author: Ean Zheng
    Class: CSCI 140
    Date: 3/8/2025
    Description:
    I certify that the code below is my own work.
    Exception(s): N/A
*/
#include <iostream>

using namespace std;

int main()
{
    cout << "Author: Ean Zheng" << endl;
    int purchaseamount;
    int change;
    int quarters = 0;
    int dimes = 0;
    int nickels = 0;
    cout << "Vending Machine Version 1 by Ean Zheng" << endl;
    cout << "There are 2 quarters, 2 dimes, and 2 nickels." << endl << endl;
    cout << "Enter a purchase amount [0 - 100] --> ";
    cin >> purchaseamount;
    cout << "You entered a purchase amount of " << purchaseamount << " cents." << endl;
    if(purchaseamount < 0 || purchaseamount > 100){
        cout << "You entered an invalid amount (not between 0 and 100).";
        return 0;
    }
    change = 100-purchaseamount;
    if(change%5 != 0){
        cout << "Your change of " << change;
```

```
Author: Ean Zheng
Vending Machine Version 1 by Ean Zheng
There are 2 quarters, 2 dimes, and 2 nickels.

Enter a purchase amount [0 - 100] --> 9
You entered a purchase amount of 9 cents.
Your change of 91 cents is rounded to 90 cents.
Insufficient coins. Your change of 90 cents cannot be processed.
PS C:\GitHubRepos\CSCI-140\Programs> ^C
PS C:\GitHubRepos\CSCI-140\Programs>
```

Question 1: What is the main difference between an expression and a condition? Can you use an expression when a condition is required in C++ (e.g., use a + b as a condition)? Why or why not?

An expression is a combination of values/literals with operators/function calls that is also a process these values go through. The expression takes all the values, subjects them to its process, delivers a final result value, and assigns it to wherever it's at. A condition, on the other hand, is a comparison statement that evaluates to either true or false. You can use an expression that evaluates to true or false, because where a condition is required only takes true or false values, and anything that evaluates to true or false is good there. Although most expressions don't evaluate to true or false and can't be used where a condition is, there are some that do. As such, they can be used where conditions are.

Question 2: It is easy to convert a switch statement to a nested if statement and it can be quite difficult to convert a nested if statement to a switch statement. Provide an example or situation to show that it can be quite difficult to convert a nested if statement to a switch statement.

A nested if statement can have two conditions that must both be fulfilled. For example, a nested if statement is composed of an if statement outside another, and both of them have two different conditions that must be fulfilled. If someone tries to convert it into a switch statement, they can't include both conditions in one switch case. If they try to put a switch case below another, the case below can be executed skipping the one above, as long as it's true. Therefore, this will cause only one of two conditions to be required for the code in the nested if statement to execute.

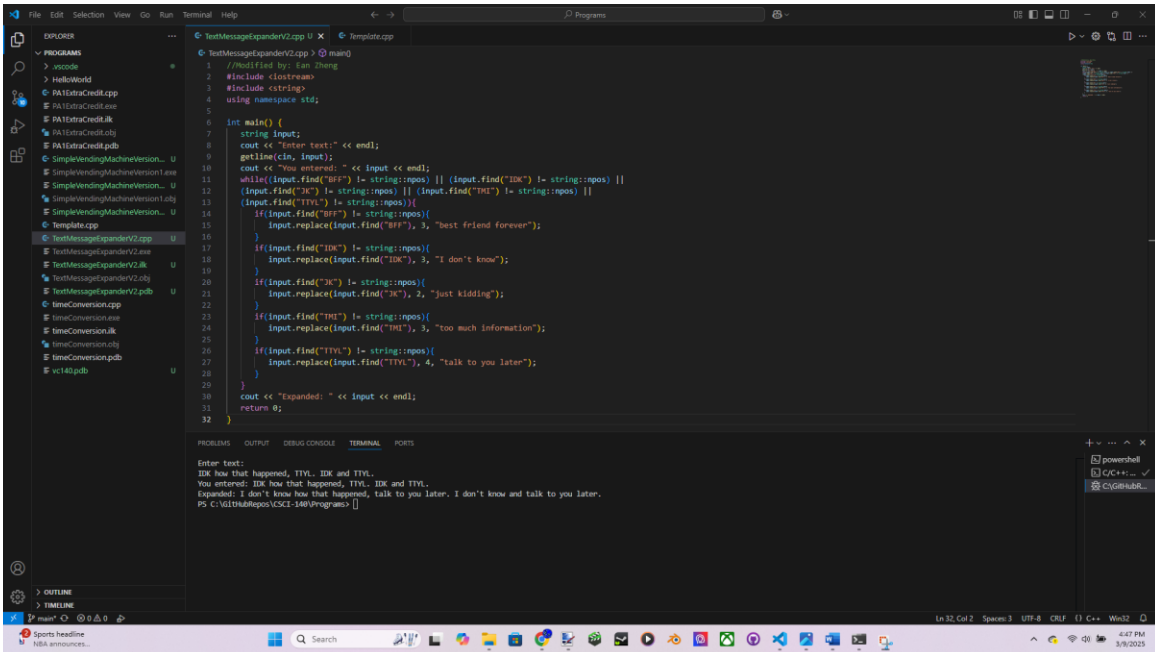
Extra Credit (2 points): Text message expander V2; copy your working code from exercise 3 to your IDE and modify it so it would expand all abbreviations. For example:
Input: IDK how that happened, TTYL. IDK and TTYL.
Output: I don't know how that happened, talk to you later. I don't know and talk to you later.

Source Code:

```
//Modified by: Ean Zheng
#include <iostream>
#include <string>
using namespace std;

int main() {
   string input;
   cout << "Enter text:" << endl;
   getline(cin, input);
   cout << "You entered: " << input << endl;
   while((input.find("BFF") != string::npos) || (input.find("IDK") != string::npos) ||
   (input.find("JK") != string::npos) || (input.find("TMI") != string::npos) ||
   (input.find("TTYL") != string::npos)){
      if(input.find("BFF") != string::npos){
         input.replace(input.find("BFF"), 3, "best friend forever");
      }
      if(input.find("IDK") != string::npos){
         input.replace(input.find("IDK"), 3, "I don't know");
      }
      if(input.find("JK") != string::npos){
         input.replace(input.find("JK"), 2, "just kidding");
      }
      if(input.find("TMI") != string::npos){
         input.replace(input.find("TMI"), 3, "too much information");
      }
      if(input.find("TTYL") != string::npos){
         input.replace(input.find("TTYL"), 4, "talk to you later");
      }
   }
   cout << "Expanded: " << input << endl;
   return 0;
}
```

Input/Output:

Enter text:
IDK how that happened, TTYL. IDK and TTYL.
You entered: IDK how that happened, TTYL. IDK and TTYL.
Expanded: I don't know how that happened, talk to you later. I don't know and talk to you later.



```cpp
//Modified by: Ean Zheng
#include <iostream>
#include <string>
using namespace std;

int main() {
   string input;
   cout << "Enter text:" << endl;
   getline(cin, input);
   cout << "You entered: " << input << endl;
   while((input.find("BFF") != string::npos) || (input.find("IDK") != string::npos) ||
   (input.find("JK") != string::npos) || (input.find("TMI") != string::npos) ||
   (input.find("TTYL") != string::npos)){
      if(input.find("BFF") != string::npos){
         input.replace(input.find("BFF"), 3, "best friend forever");
      }
      if(input.find("IDK") != string::npos){
         input.replace(input.find("IDK"), 3, "I don't know");
      }
      if(input.find("JK") != string::npos){
         input.replace(input.find("JK"), 2, "just kidding");
      }
      if(input.find("TMI") != string::npos){
         input.replace(input.find("TMI"), 3, "too much information");
      }
      if(input.find("TTYL") != string::npos){
         input.replace(input.find("TTYL"), 4, "talk to you later");
      }
   }
   cout << "Expanded: " << input << endl;
   return 0;
}
```

```
Enter text:
IDK how that happened, TTYL. IDK and TTYL.
You entered: IDK how that happened, TTYL. IDK and TTYL.
Expanded: I don't know how that happened, talk to you later. I don't know and talk to you later.
PS C:\GitHubRepos\CSCI-140\Programs> 
```