Please make sure to write your name and the date in the top left corner. You may use any course materials to answer the following questions and you may collaborate with others, but the work shown must be your own.

1 Spot The Error

Problem 1.1. The code snippet below is supposed to determine if a variable stores a value that is greater than, less than, or equal to 8. Identify the error(s) and write the correct line(s).

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

int main()
{
    int num = 6;

    if (num > 8) {
        cout << "The number is greater than 8." ;
    }
    else if (num = 8) {
        cout << "The number is equal to 8.";
    }
    else {
        cout << "The number is less than 8.";
    }

    return 0;
}
```

else if (num == 8) {

Problem 1.2. The code snippet below is supposed to determine if a variable stores a value for an angle that is obtuse, right, or acute. Identify the error(s) and write the correct line(s).

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

int main()
{
    int angle = 120;
    if (x>90) {
        cout<<"It is an obtuse angle." ;
    }
    elif(x=90) {
        cout<<"It is a right angle.";
    }
    else{
        cout<<"It is an acute angle.";
    }
}
```

else if (x == 90) {

Problem 1.3. The code snippet below is supposed to determine if a variable stores a value that is equal to zero or not. Identify the error(s) and write the correct line(s).

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

int main()
{
    int num = 7;

    if !(num) {
        cout << "The number is zero.";
    }
    else {
        cout << "The number is not zero.";
    }

    return 0;
}
```

(handwritten) if (! num) {

Problem 1.4. The code snippet below is supposed to determine if a variable stores a value that is equal to zero or not. Identify the error(s) and write the correct line(s).

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

int main()
{
    int num = 0;

    else {
        cout << "This is the 'else' block.";
    }
    if (num == 0) {
        cout << "The number is zero.";
    }
    else {
        cout << "The number is not zero.";
    }

    return 0;
}
```

(handwritten) } Delete

Problem 1.5. The following code snippet is expected to accept a user provided integer and then state whether that number is even or odd. Identify the error(s) and write the correct line(s).

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

int main()
{
    int num;
    cout << "Provide an integer:" << endl;
    cin >> num;

    if (num/2){
        cout << "The number is even." << endl;
    }
    else {
        cout << "The number is odd." << endl;
    }

    return 0;
}
```

(handwritten) if ((num % 2) == 0)

Problem 1.6. The following code snippet is expected to accept a user provided character and then state whether the corresponding grade passes or not. Identify the error(s) and write the correct line(s).

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

int main()
{
    char grade;
    cout << "Provide a grade (A, B, C, D, or F):" << endl;
    cin >> grade;

    if (grade == 'A' || 'B' || 'C'){
        cout << "This is a passing grade." << endl;
    }
    else if (grade == 'D'){
        cout << "This grade passes with conditions." << endl;
    }
    else {
        cout << "This is a failing grade." << endl;
    }

    return 0;
}
```

(handwritten) if (grade=='A' || grade=='B' || grade=='C') {

2 Hydration App

Your goal is to drink 3 liters(L) of water every day but you aren't great at remembering to do it! So you decide to create a hydration app that tracks your water intake every day and will alert you based on how much you drank for the day. The program first asks how much water you drank for the day, which is in liters, and then displays a message based on whether you have hit your goal for the day. Next, it will also tell you how much water you have left to drink.

The following are the possible messages you will get based on your intake:

- If you're drinking 1.5L or less, then you get

```
You're very, very dehydrated! Get that water in! You have
X liters left to drink.
```

- If you're drinking more than 1.5L but less than 3L, then you get

```
You're doing great, but you're still halfway to your goal!
Get that water in! You have X liters left to drink
```

- If you're drinking 3L or more, then you get

```
You've hit your goal for the day! Great job getting hydrated!
```

Note that **X** is the amount of water left after subtracting your intake.
Below are some sample runs. User input is shown in bold.

Sample Run 2.0.1
How much water did you drink today? (in liters)
0.75
You're very, very dehydrated! Get that water in! You have 2.25 liters left to drink.

Problem 2.1. Write out the steps you would use to solve this problem by hand as pseudocode.

Ask how much water has been drinked
Get input in liters
Calculate how much water needs to be drinked
if water < 1.5
    You're hydrated. Drink (water left)
else if water <= 3
    Good, drink (water left)
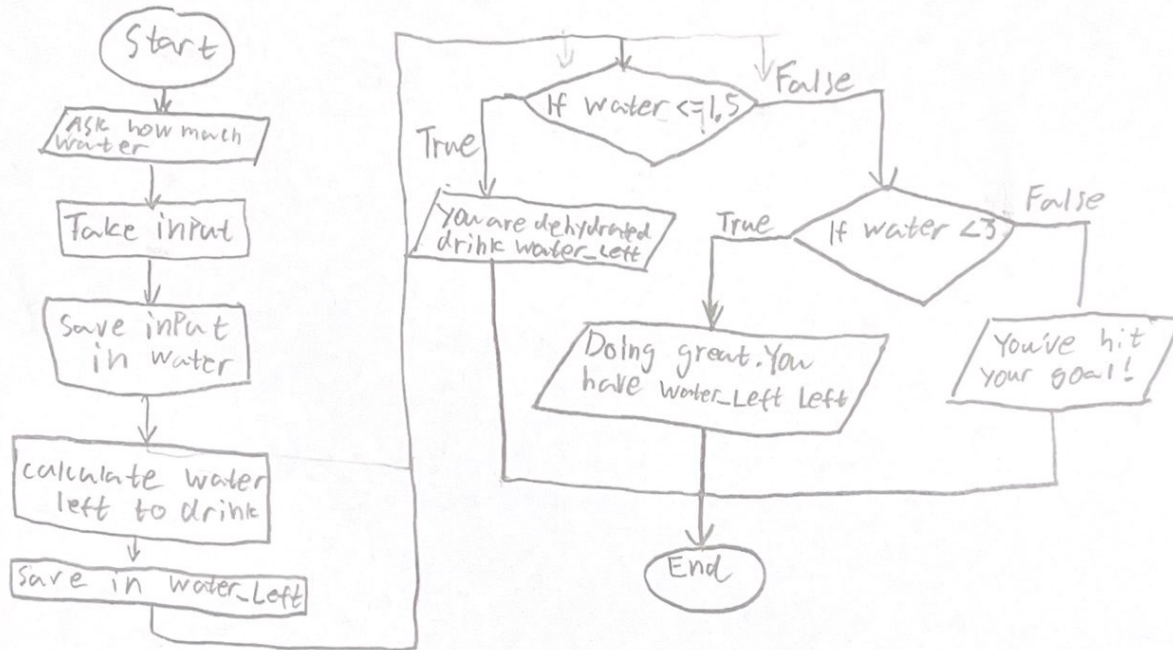else if
    Good job, you're done.

Problem 2.2. Depict your pseudocode using a flowchart.

Problem 2.3. Pick a random number between 0 and 3.0 for a sample run. Follow the steps you wrote for this number to find your end result, and verify it.

yes

Problem 2.4. Identify two possible values that are "boundaries" in this problem that you will have to test. What should happen for these values?

1.5 - Should get "dehydrated", gets "dehydrated"
3 - Should get "goal hit" - gets "goal hit"

Problem 2.5. Translate your pseudocode into a c++ program to solve the above code.

Submission Instructions: Create a zip file that contains your solution .cpp file for question 2 as well as photos of this handout and submit on Canvas.