Red Bird Racing EVRT Software Training

Homework 2

Last updated: 2025-10-06

Authored by: CHEUNG Pui Ki (Planeson)

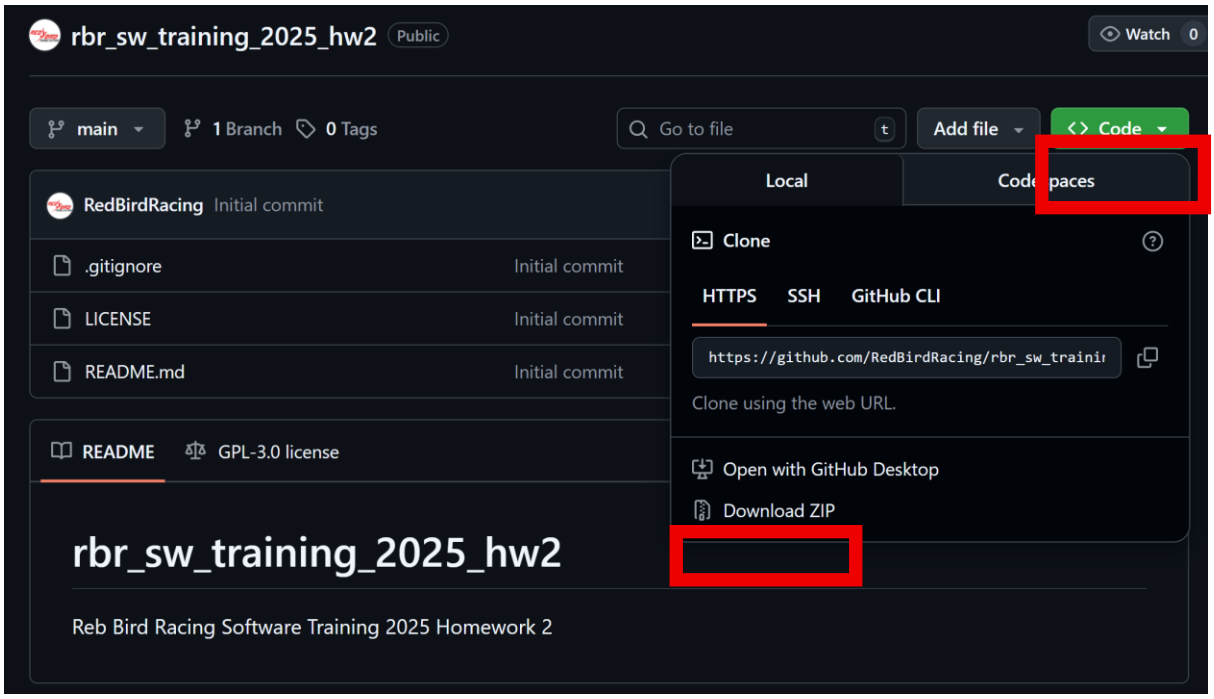
Deadline: 2025-10-12 23:59:59 HKT (Sunday Night)

Contents

Instructions

Go to the GitHub page of the skeleton code, go to Code and download as ZIP.

https://github.com/RedBirdRacing/rbr_sw_training_2025_hw2



Complete the homework as instructed, then create your own repository, and push your work to said repository. Once you are done, send a link to Carson Cheung (@Planeson, +852 9437 6620) via WhatsApp. I will grade your work and give feedback. If you wish to resubmit after you made edits, simply send me a message informing me that you have edited the code and would like another review. The score you get for this homework will the highest score you get out of all attempts. You get unlimited attempts before the deadline, so you are suggested to submit early and keep correcting your code.

If you have any questions, regarding the homework, the duties of the software team, software tutorial contents, etc, tag and ask me in the group.

Testing your code

You can install C++, for instructions, refer to the COMP2011 page. You can login to this site even if you haven't studied COMP2011, as long as you have a CSE account.

https://course.cse.ust.hk/comp2011/vscode/

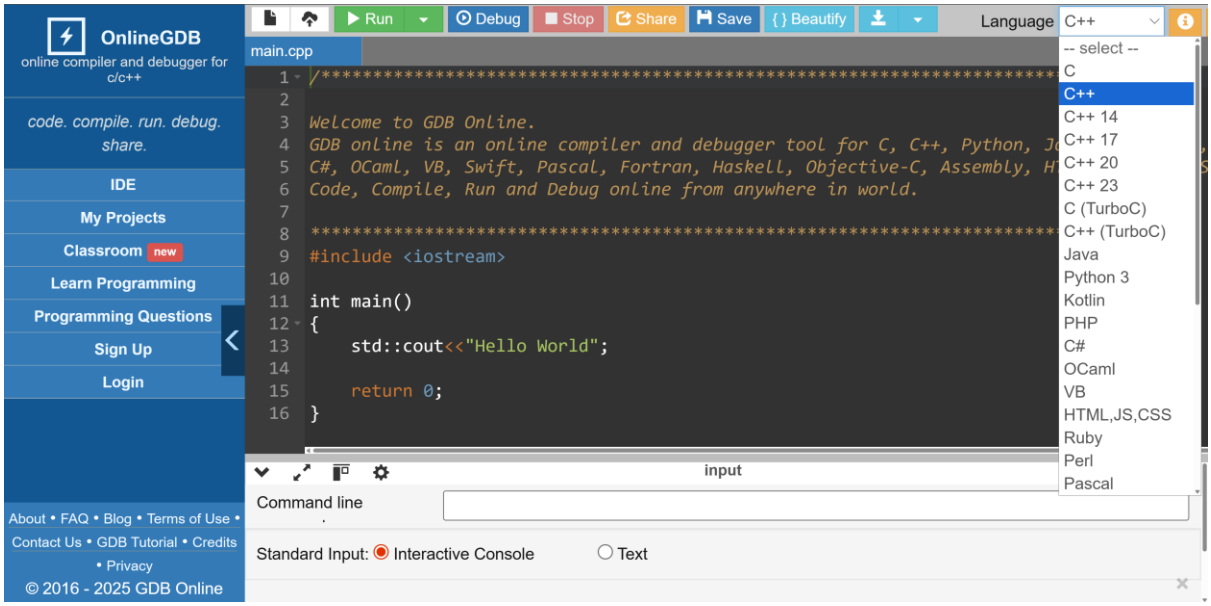
The easiest way to run your program is to run the following two command each time:

```
g++ FILENAME.cpp -o FILENAME
./FILENAME
```

For instance, to compile q1.cpp into a file call q1(q1.exe on Windows), then run q1(.exe),

```
g++ q1.cpp -o q1
./q1
```

If you don't wish to install VSCode, you can also test your code on OnlineGDB.



Homework content:

Homework 2 focuses on debugging. Answer the following questions in their respective files.

q1.cpp

1. Look at the following code. Give your answers at the top of the q1.cpp file. State what kind of error the following code produces (compilation error, runtime error, logic error), then explain why in no less than 10 words, per error. An example is given to you for your reference. Note that there can be multiple errors with one statement, but in this question, each part only contains one error.

a.
```cpp
#include <iostream>
using namespace std;
int main() {
    char inp[8];
    cin >> inp;
    cout << inp << endl;
    return 0;
}
```
Part a is given to you as an example. You only need to know that `char inp[8]` can hold 8 characters to understand why this is an error.

b.
```cpp
int main() {

    int a = "hello" / "world";
    return 0;
}
```

c.
```cpp
#include <iostream>
using namespace std;
int main()
{
    int myNum = 5;
    if (myNum < 10)
    {
        cout << "myNum is greater than 10" << endl;
    }
    else
    {
        cout << "myNum is 10 or less" << endl;
    }
}
```

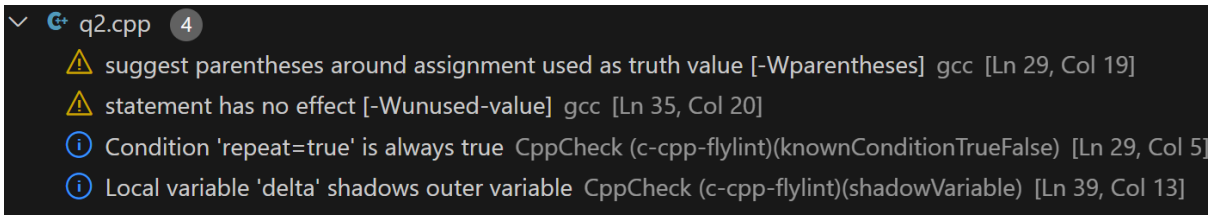
d.
```
int main()
{
    int a = 4;
    int b = 0;
    int c = a / b;
}
```

2. Explain the what the following code is trying to do. Find the bug making the results wrong. Document why and how it happens. Write a simple fix for it.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 3;
    int b = 4;
    int percent = (a / b) * 100;
    cout << percent << "%" << endl;
    return 0;  // returning 0 for main() just means the program ended successfully
}
```

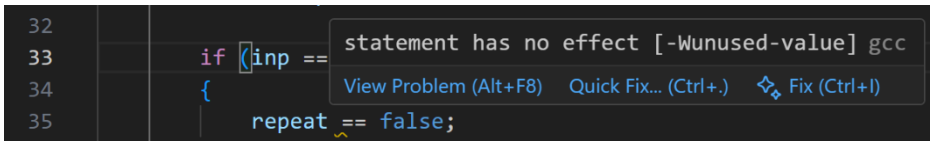
q2.cpp

The code compiles, but there are compiler warnings. Use this information and figure out why the code doesn't work as expected. Suggest fixes. If you know how to compile with a debugger, you can use that as well. Note that some of them refer to the same line. Refer to each warning by line when answering. Refer to this list of warnings.

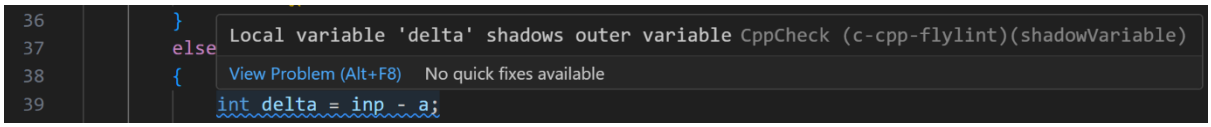


1. Line 29, `while (repeat = true)`

2. Line 35, `repeat == false;`



3. Line 39, `int delta = inp - a;`



Explain the issue with each line, as suggested by cppcheck and the gcc compiler. Give a fix to the issue to make the program work as expected.

Hints

- Compile and test your code to check if it matches the expected results. We gave you a .cpp file because we want you to actually run the code! We are not CSE, you are not a human compiler, you should try code out!
- Read the instructions carefully, they are meant to help you, not distract you.
- Avoid using AI tools directly, since it is eliminating thinking from your working process. You can use it to help you think, but don't ask it to solve the questions for you directly.
- The questions should be so straightforward that copilot *autocomplete* should be able to finish the code for you. If you want to learn well, I suggest disabling copilot autocomplete for homework questions.