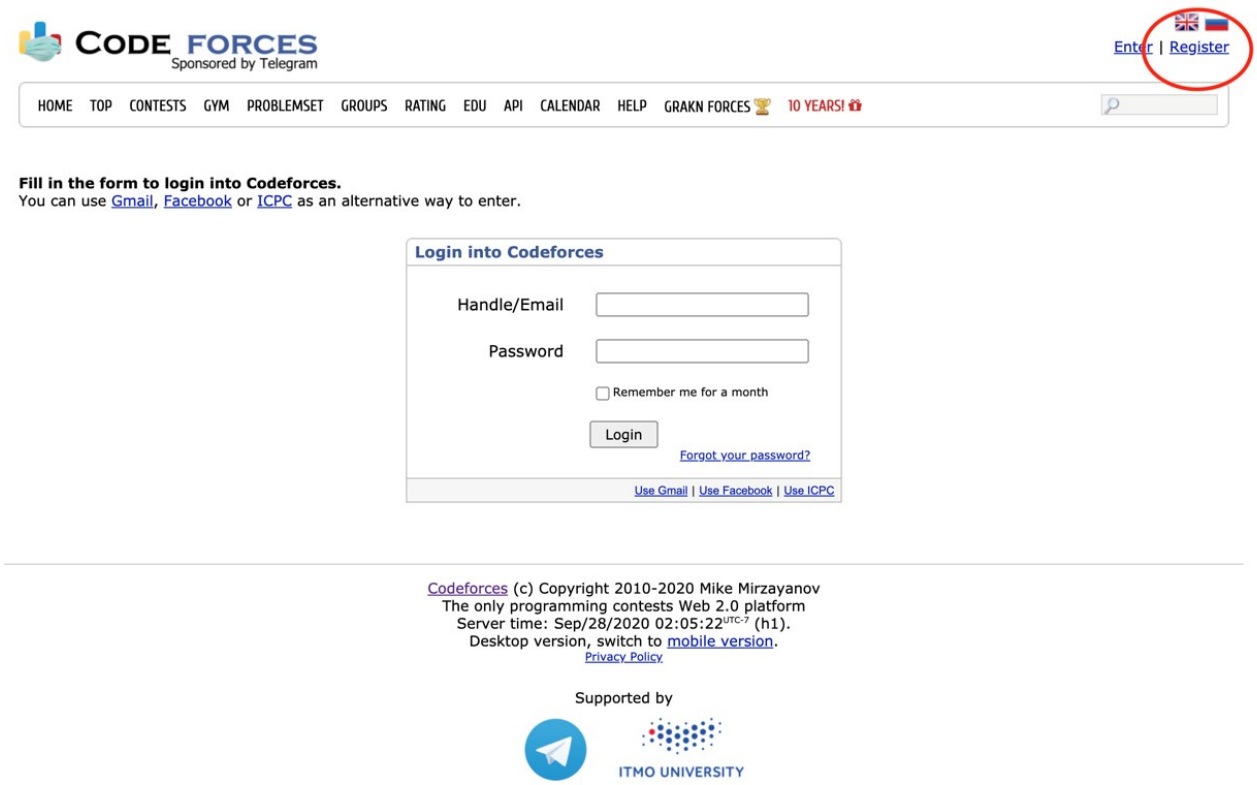
Guidelines of the Online Judge System

(Some of the screenshots are for a previous course, but the links are all up to date.)

0. Register Your Account in CodeForces

Access the CodeForces website with URL https://codeforces.com/enter.

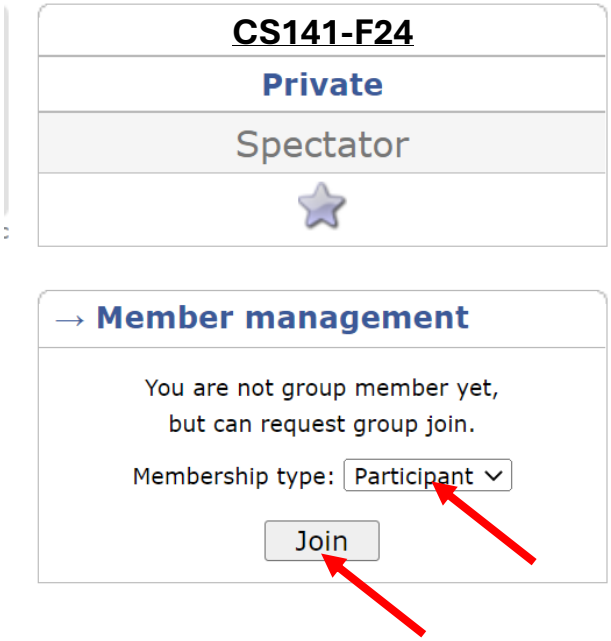
If you already have a CodeForces account, log in directly. If not, Click “Register” to register an account, then log in.



1. Join the Group as a Participant

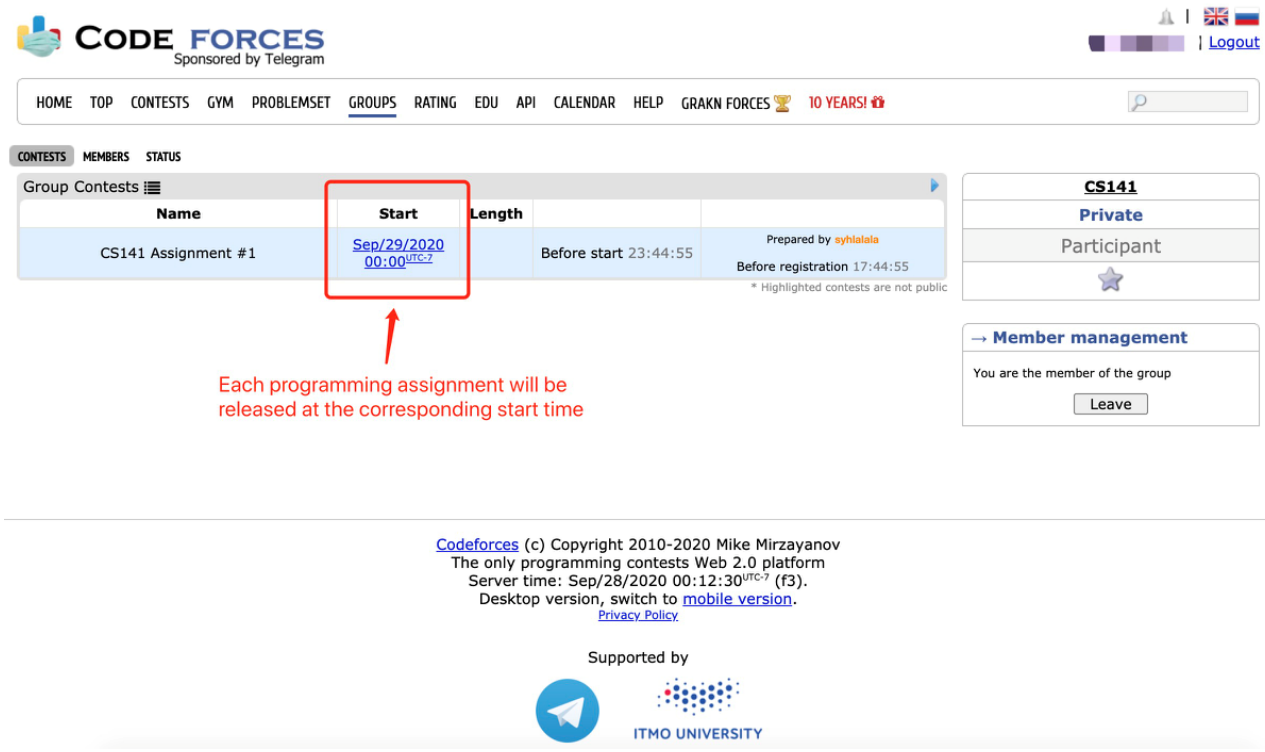
After login, click here to join our group: https://codeforces.com/group/imtYsVNAKN/contests

Remember to choose "Participant" as Membership type. (The default option is "Spectator", which will not allow you to submit your code.) Then click **"Join"**.

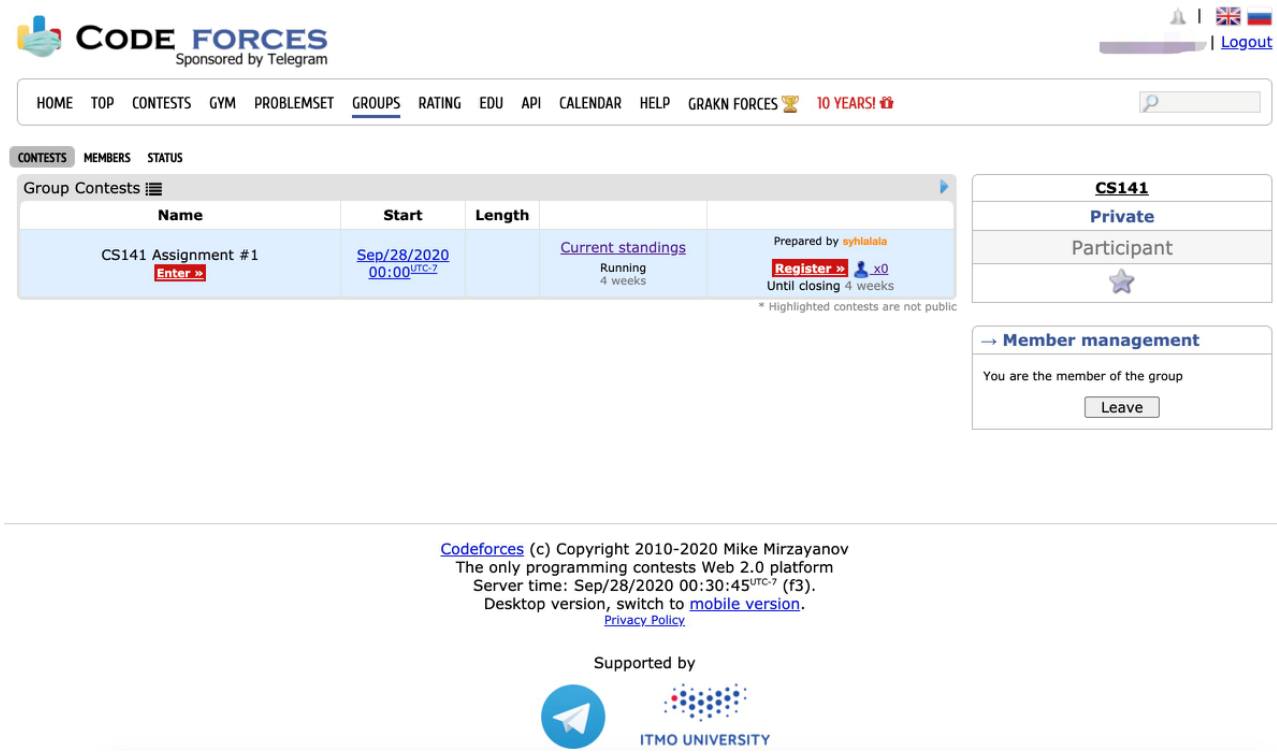


Then you can see the home page of our group, each programming assignment will be released at the corresponding start time. Usually, this is the time that the corresponding assignment is released.

The rest of the screenshots were taken from other courses/contests as examples. All the steps should be exactly the same.



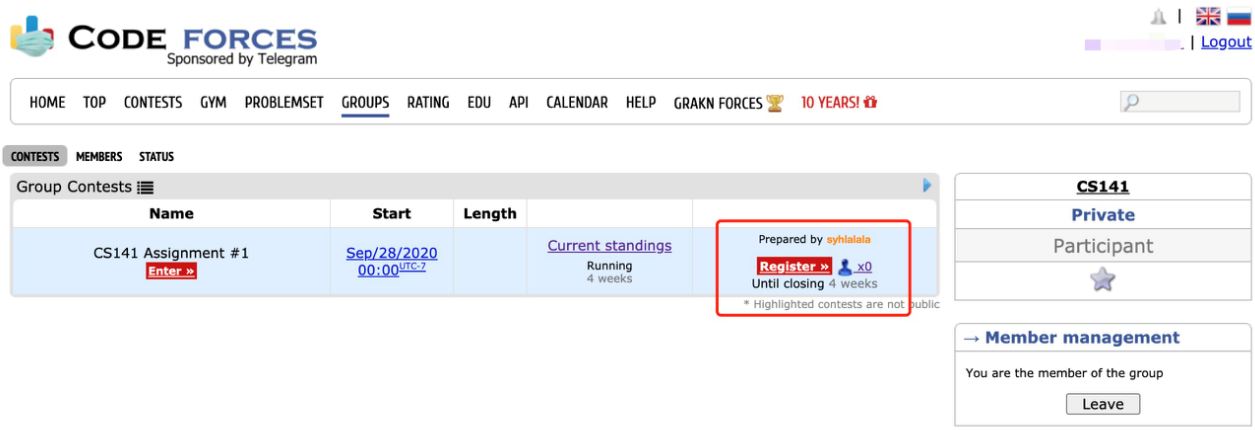
Once it's released, the page will look like this:



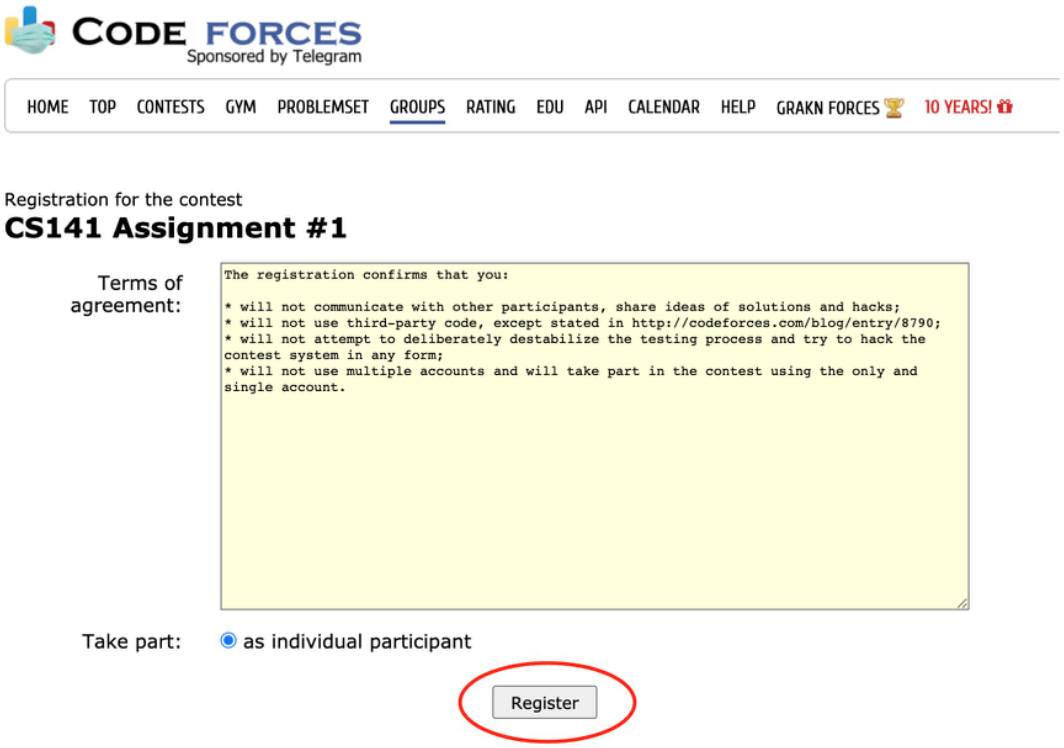
2. Registration for the Contests

After it's released, you can see the page above.

Click “Register” to register for the contest. You should be registered for the contest to be able to submit.



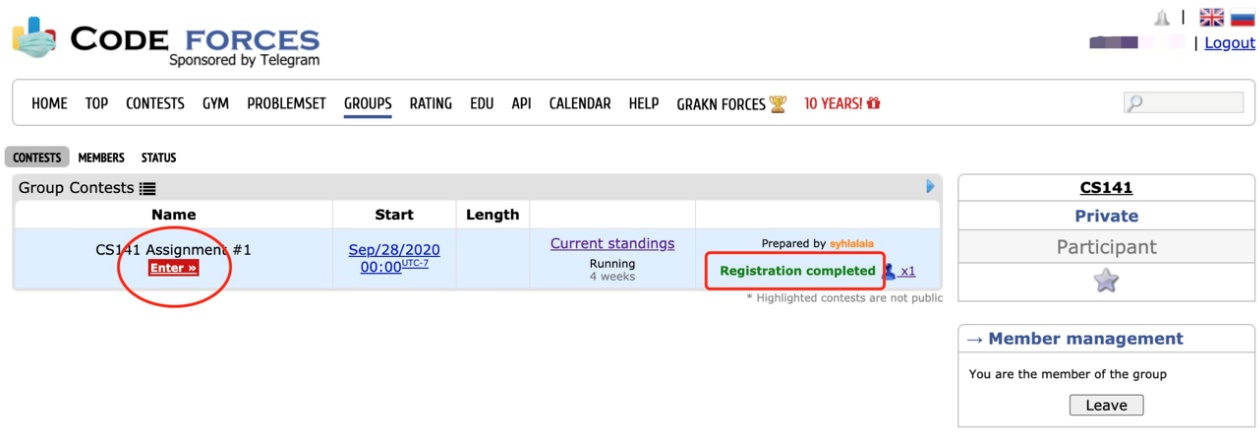
Continue clicking **“Register”**, then you will receive a message that “You have been successfully registered”.



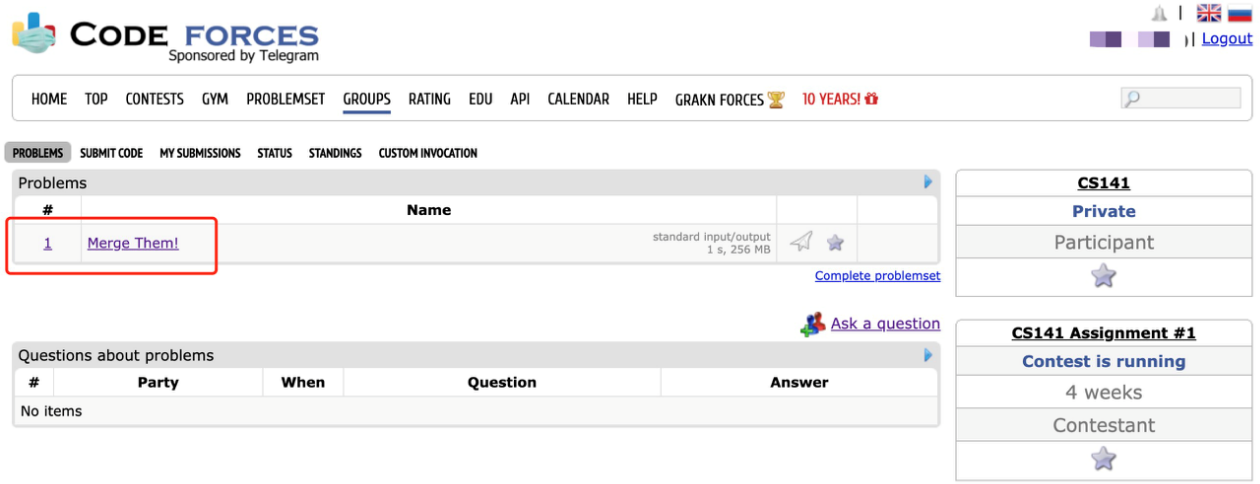
3. Start Programming

Before starting programming, make sure you can see "Registration Completed" displayed here, which means you have successfully registered. If not, try to repeat the registration step.

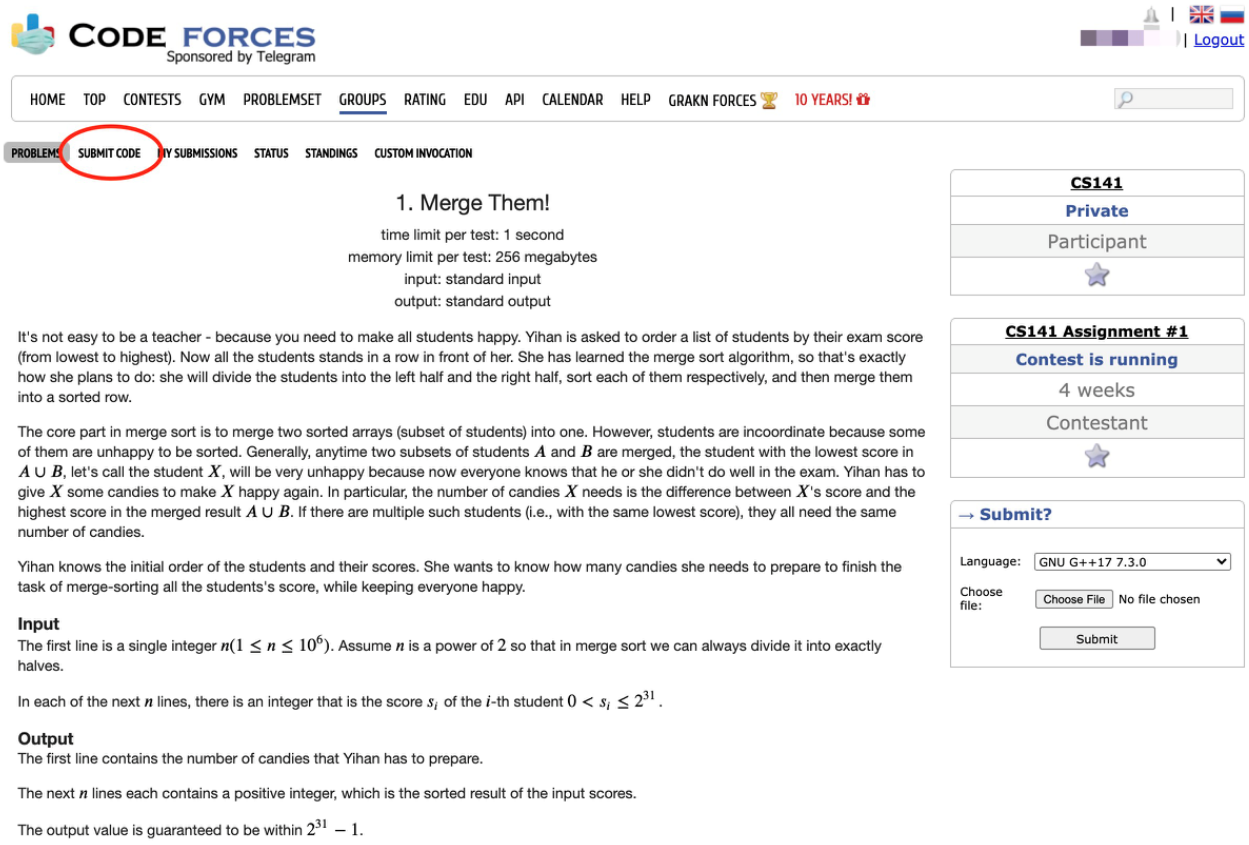
Then, **click “Enter”.**



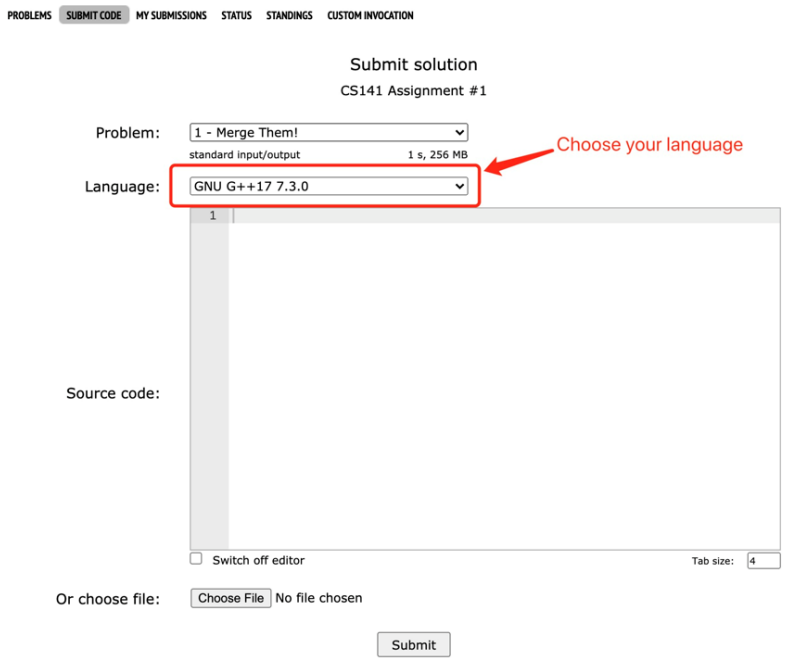
You will see the programming problems for this assignment.



Click on the name to enter. Then you can see the whole problem.



To submit your code, **click "Submit Code". Don't forget to choose your language.**



After finishing your code, click the **"Submit"** button to submit your answer.

Then you can see whether your answer is correct. If all test cases are passed, it will be judged as **"Perfect result – x points"** (x is the total number of points of that problem). If not, means you failed in some test cases, and it will show your score based on the number of tests you passed.

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS STANDINGS ADM. EDIT CUSTOM INVOCATION

My Submissions

#	When	Who	Problem	Lang	Verdict
130314428	Sep/29/2021 20:44 UTC-7	syhlalala	E - Feeding Friendsy	GNU C++17	**Perfect result: 80 points**
130314158	Sep/29/2021 20:38 UTC-7	syhlalala	E - Feeding Friendsy	GNU C++17	**Perfect result: 80 points**
130313967	Sep/29/2021 20:33 UTC-7	syhlalala	E - Feeding Friendsy	GNU C++17	Partial result: 75 points
129912417	Sep/26/2021 02:40 UTC-7	syhlalala	I - Sphere Mongers	GNU C++17	Partial result: 96 points
129911661	Sep/26/2021 02:29 UTC-7	syhlalala	I - Sphere Mongers	GNU C++17	Partial result: 96 points
129911303	Sep/26/2021 02:24 UTC-7	syhlalala	I - Sphere Mongers	GNU C++17	Partial result: 18 points

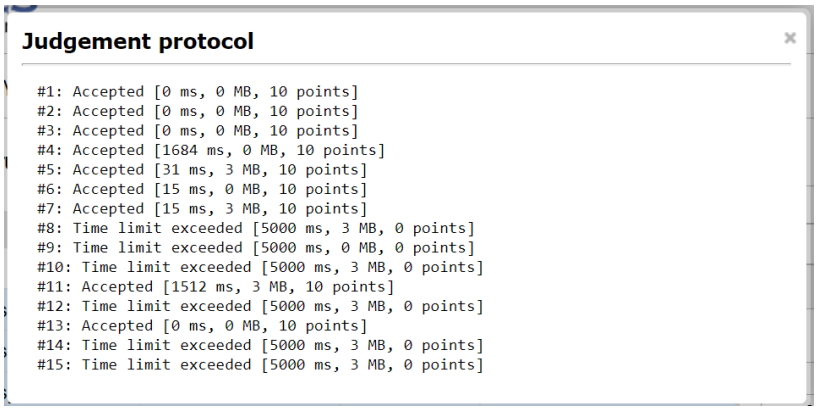
You can submit an arbitrary number of times before the deadline. The highest score for each problem will be counted. If you want to re-submit after debugging, you could just go back to the "Submit Code" page and submit your new version.

Note that you have to pass the first test case (which is usually just the example) to get any points. Otherwise you will see something like:

129737615	Sep/24/2021 00:54 UTC-7	syhlalala	H - Fetch Quest	GNU C++17	Wrong answer on test 1

This is, however, a feature of the system. So just make sure you at least pass the example.

If you want to know more about your results, click the "verdict" column for each problem in the table above. You will see a list of your results for each test case. For example:



This helps you understand the problems in your code.

With some of the core operations and functions covered above, you can try clicking on other buttons to explore other features that have not been introduced!

4. Other Notes

- You won't be able to see the test data. This is also the situation when you are in an interview, or, when you are actually at work in a company. One important skill as a programmer is to design test cases to find bugs in your own code. You need to consider a lot of corner cases to make sure your code is correct and robust.
- Each problem has 10-20 test cases, each with some points, based on the problem. This means that even your algorithm cannot solve the entire problem, if you can solve part of the problem (e.g., only for small inputs or simple inputs), you can get some points. Don't give up!
- For all the problems, we guarantee that a reasonable algorithm implemented in C++ will pass all test cases within the time limit. Other languages are allowed, but not guaranteed to always satisfy time limit. We will use time scaling policy to help accommodate this: The system automatically adjusts time limits by the following multipliers for some languages. You can see more details here. However, still, C++ is highly recommended because it is the most efficient in running time.
- Sometimes the input time may be long and it can dominate your running time. To simply make your input time faster, here are some useful material C+, Java and Python (again, even so, it's not guaranteed that python can always pass the largest test case):
    - C++: https://www.geeksforgeeks.org/fast-io-for-competitive-programming/
    - Java: https://www.geeksforgeeks.org/fast-io-in-java-in-competitive-programming/
    - Python: https://www.geeksforgeeks.org/python-input-methods-competitive-programming/

Happy Coding!