CS 101: Computer Programming and Utilization

July-Nov 2016

Prof. Bernard L Menezes
(cs101@cse.iitb.ac.in)

Lecture 1: Introduction to Prutor
(Online system for program preparation and submission)

About These Slides

- These slides describe Prutor which is a part of an *Intelligent Tutoring System (ITS)* developed by Amey Karkare at IIT Kanpur

- At IIT Bombay, it is available for cs101 at

   https://cs101.cse.iitb.ac.in

- Slides prepared by Nisha Biju and Anshuman Dhuliya

Login Page

- Login at

  https://cs101.cse.iitb.ac.in

- Accesible through the course page also

  http://www.cse.iitb.ac.in/~cs101/

- Enter your IITB ldap id and password to login

- If Prutor recognizes you as a registered student of cs101, it will authenticate your credentials
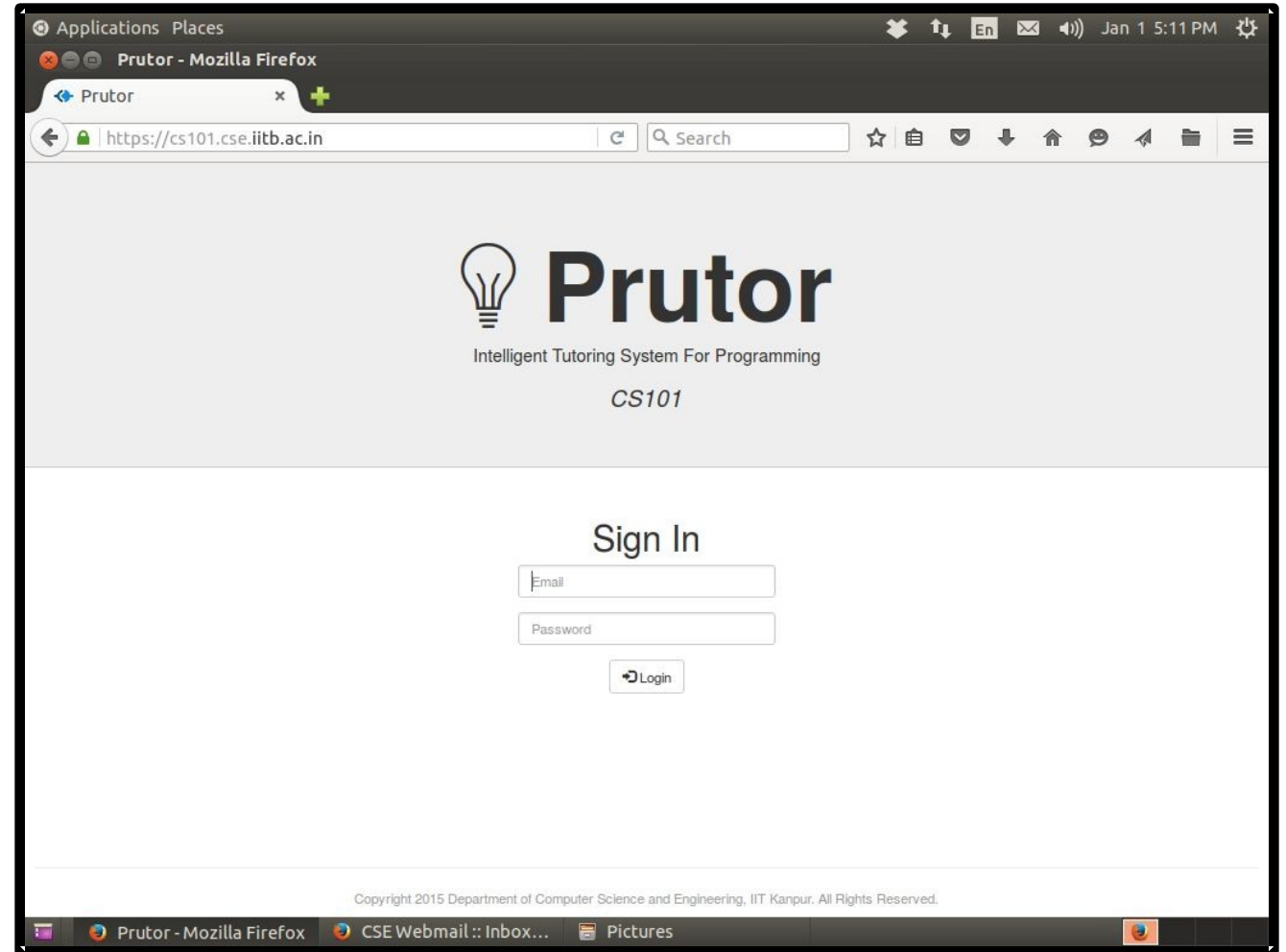
Your Homepage

Your homepage shows you the

- Events Arena: Questions (i.e. problems for which you need to write programs

  (may be ungraded questions)

- Course Statistics Arena: The status of you submissions

- Grade Card Arena

- CodeBook Arena: Your submitted programs
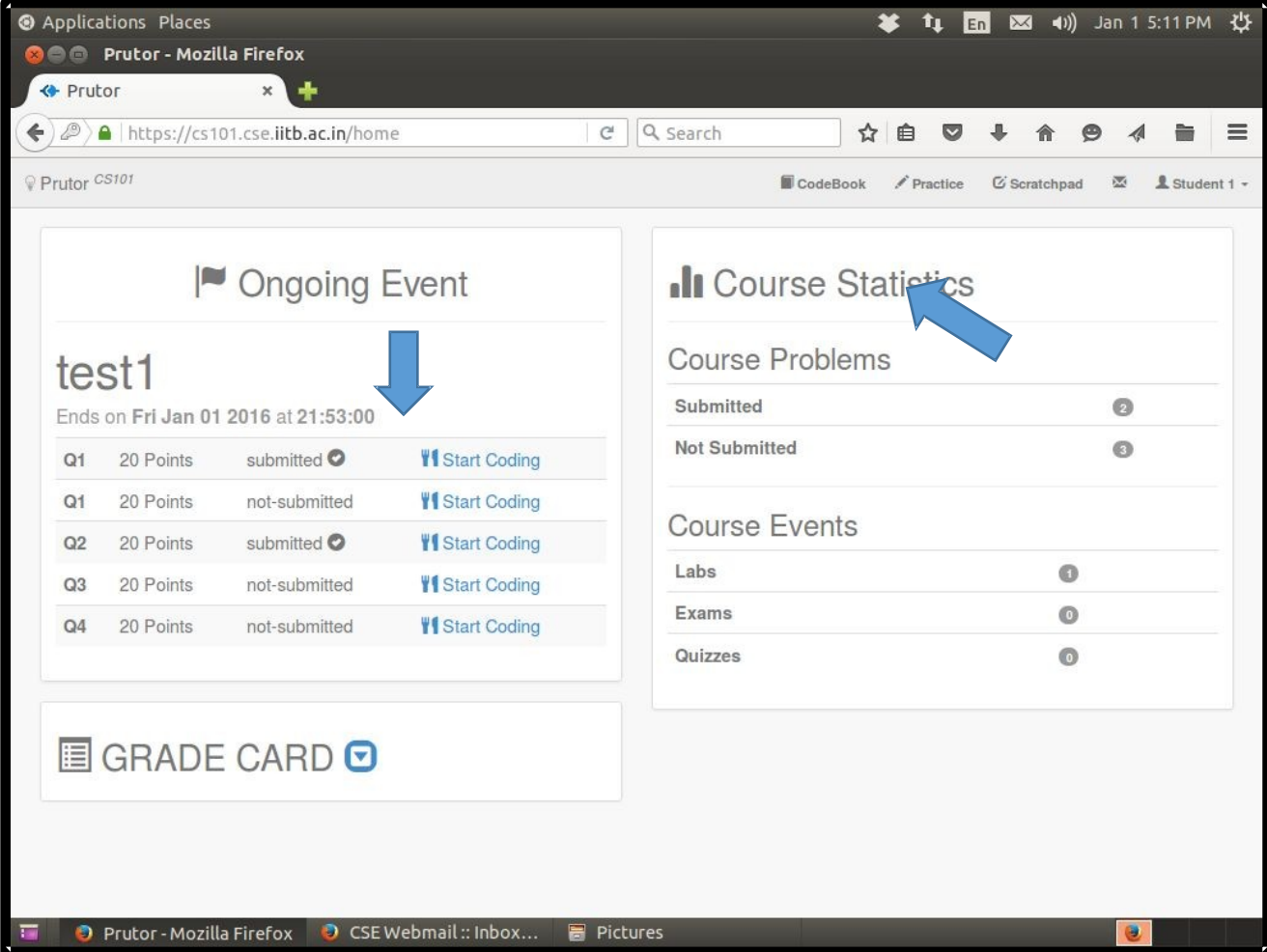
- Practice Arena

- Scratchpad Arena

CodeBook Arena

- Here you can see all the problems submitted by you.

- In the picture on the right, the CodeBook is blank as the student has not submitted any question.
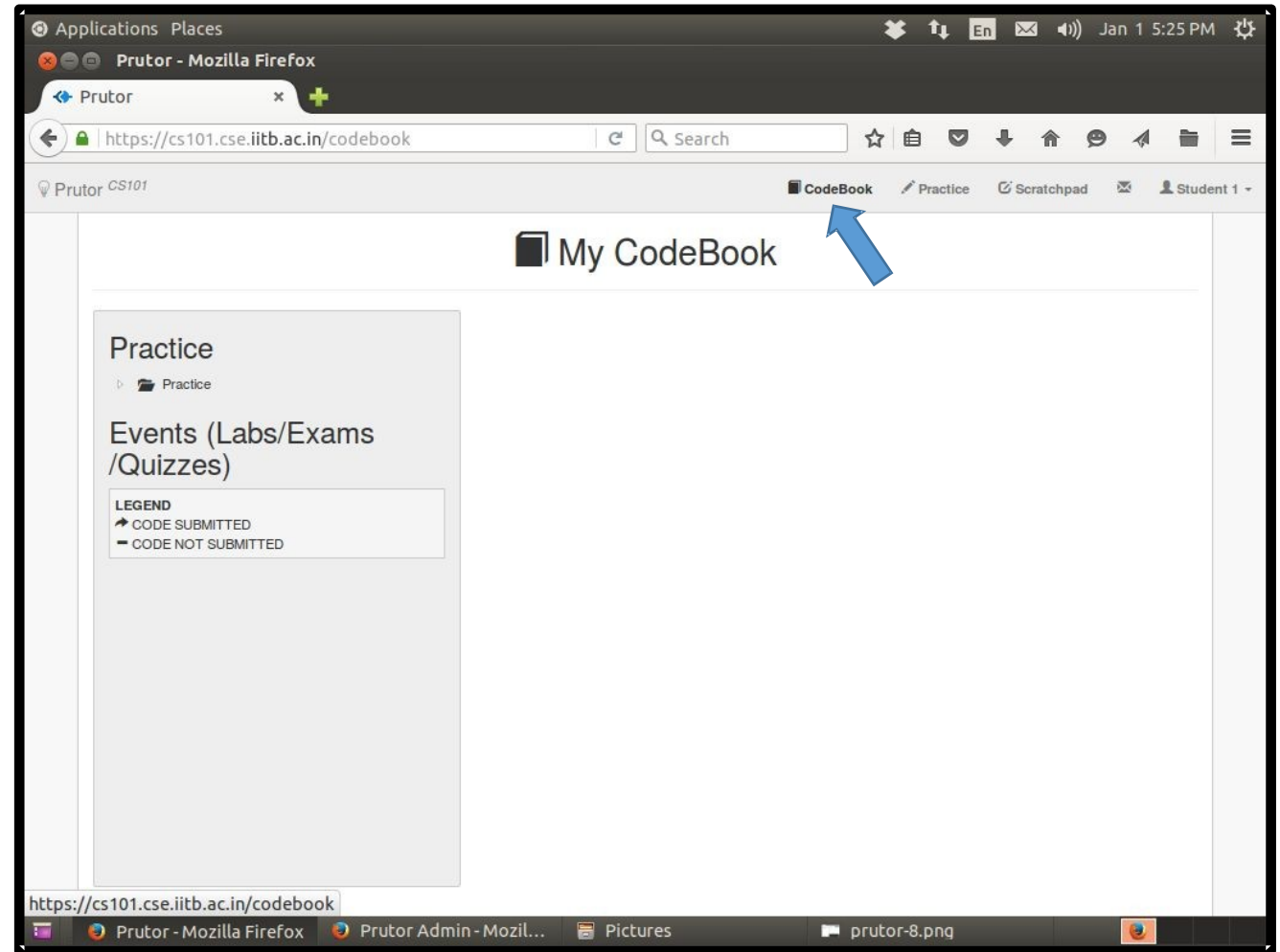
Events Arena

- An Event is visible only when it is Ongoing

- In the picture on the right, the ongoing event is test1, which has five questions with 20 marks each

- Click on the Start Coding link to start solving the questions

- The events in graded labs will contribute to your grades, and the events in regular labs will be for practice only

    (We will have six graded labs)

Plain C++ and Simplecpp

- The events arena shows the list of problems

- Some of these are plain C++ problems and some are simplecpp problems

- All the problems are edited, compiled, and executed in the same manner

- The output for plain C++ programs appears directly

- Viewing the output of simplecpp needs an additional step
  (Explained in a later slide)

Code Editor

- You reach here by clicking the Start Coding link of a question

- The Code Editor saves your code every 5 seconds

- You can also save your code by pressing Ctrl-s

- Or you can save using the Save option of the File tab

We can play back your key strokes and retrace your steps to figure out whether you wrote the code yourself or copied it


Problem Description

Your Editor
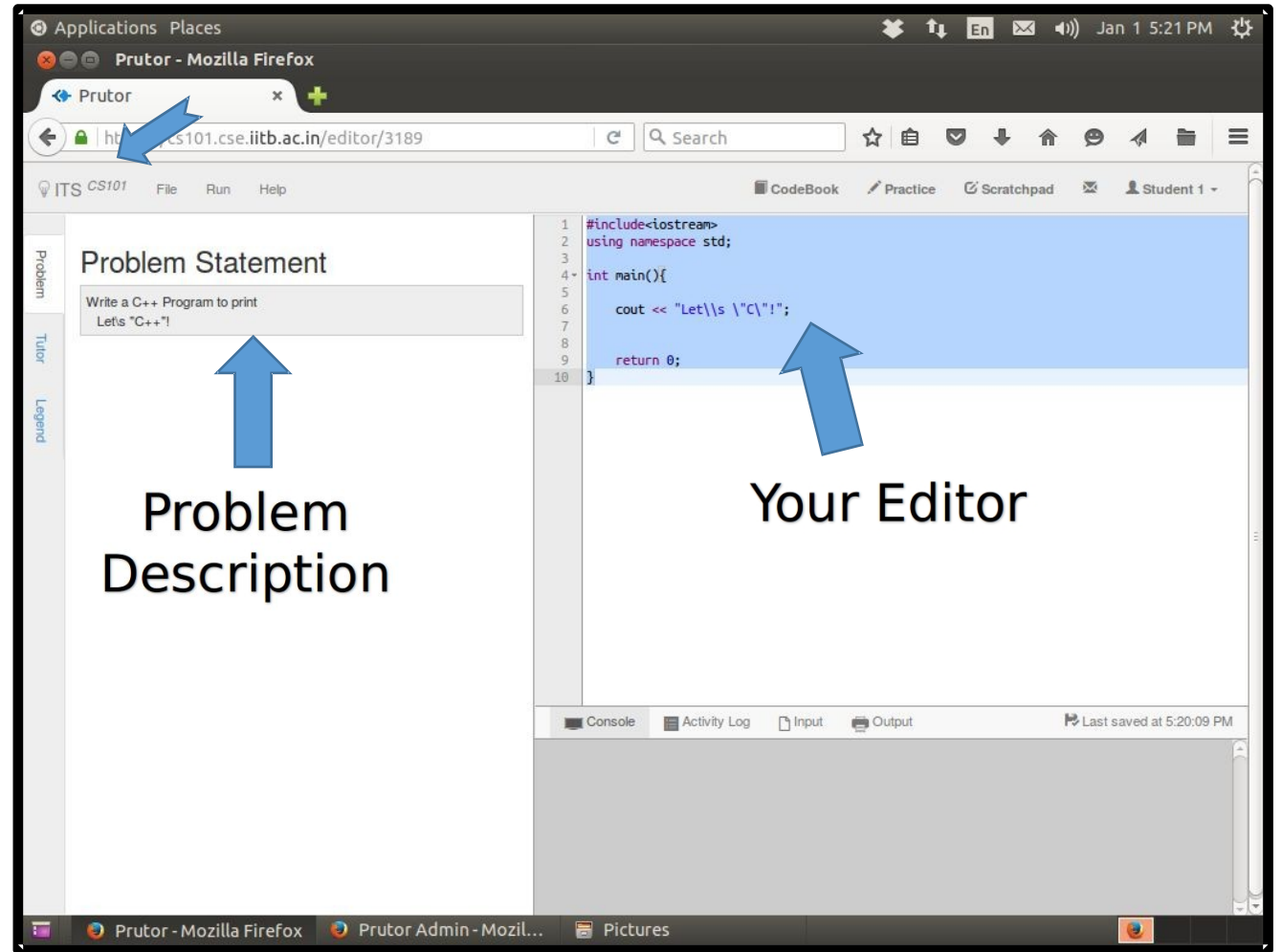
Compiling Code

- After writing the code, choose the Compile option in the Run tab
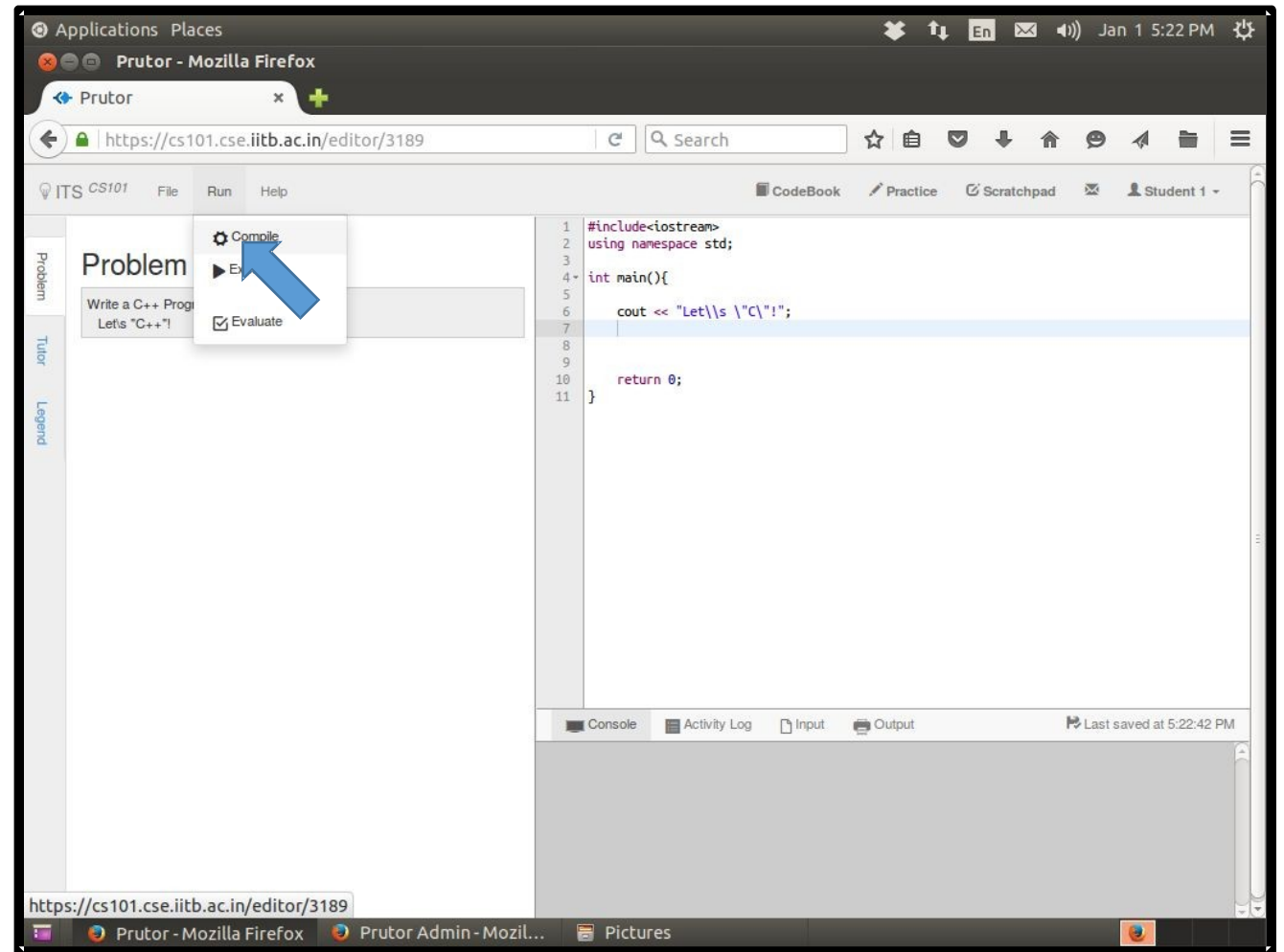
Compiler Messages

- You can view the compilation results below the Code Editor in the gray area, which is your virtual console

- Also a floating notification is shown on top-right corner for few seconds

- In the picture on the right, the compilation has failed as can be easily inferred from the message in red blob of the floating notification
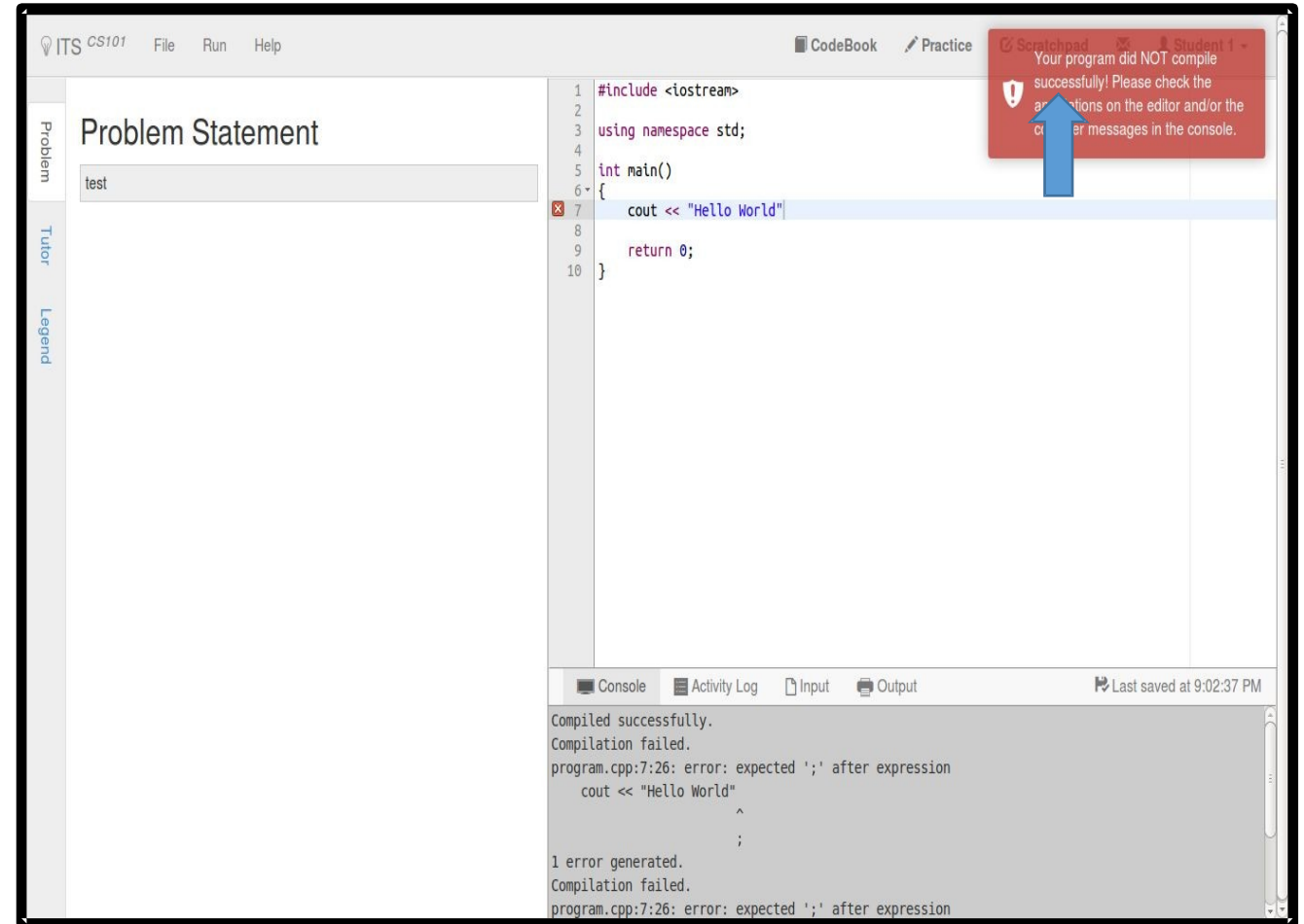
Error Messages

- The compiler also issues messages about errors in specific lines

- We can see the error in the box if we take the mouse pointer towards the left side of the editor

- Read the compiler message at the bottom carefully and correct your error
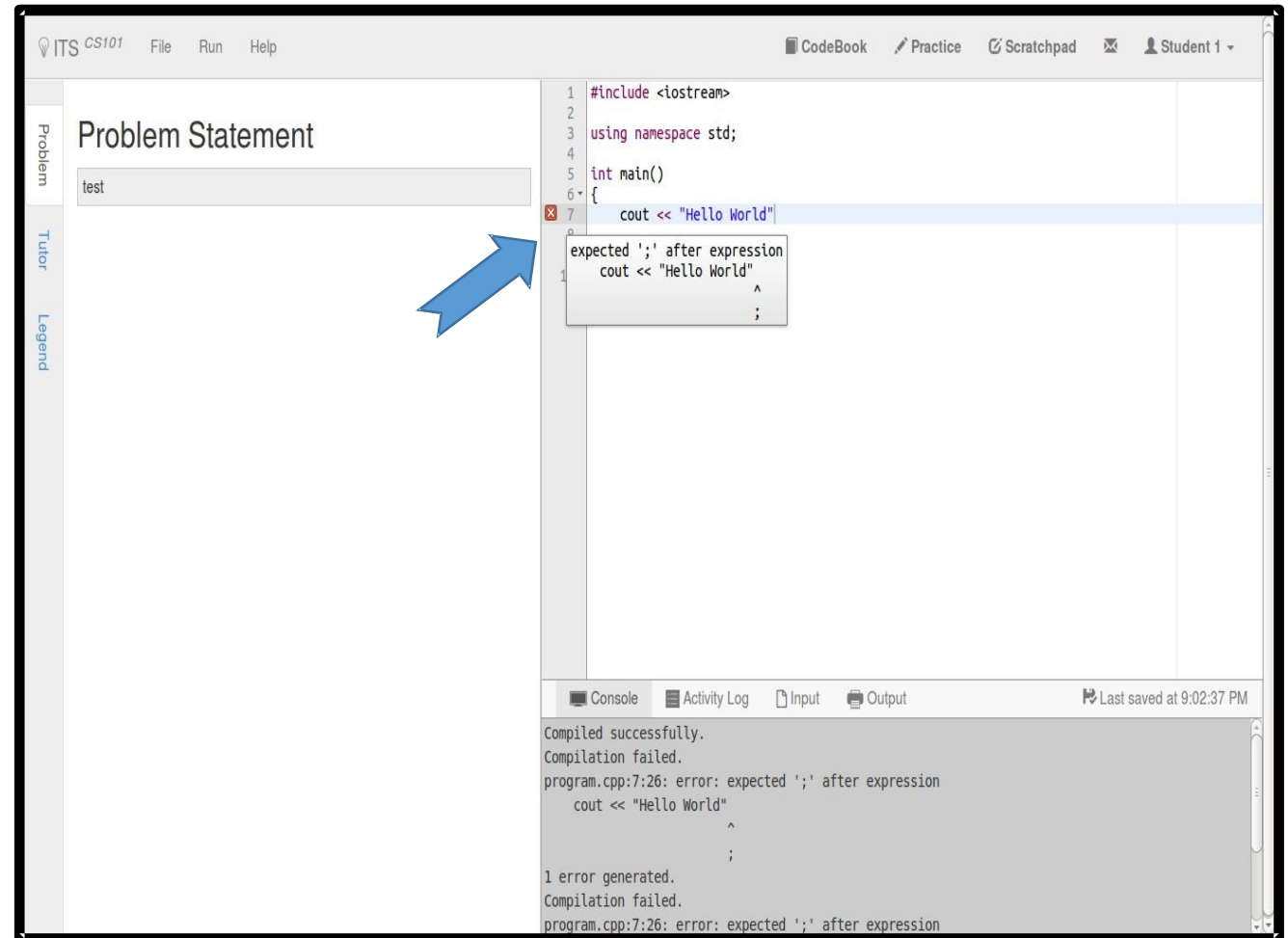
Successful Compilation

- The floating notification on top-right corner will notify you when your compilation is successful

- The color of the floating notification would be green
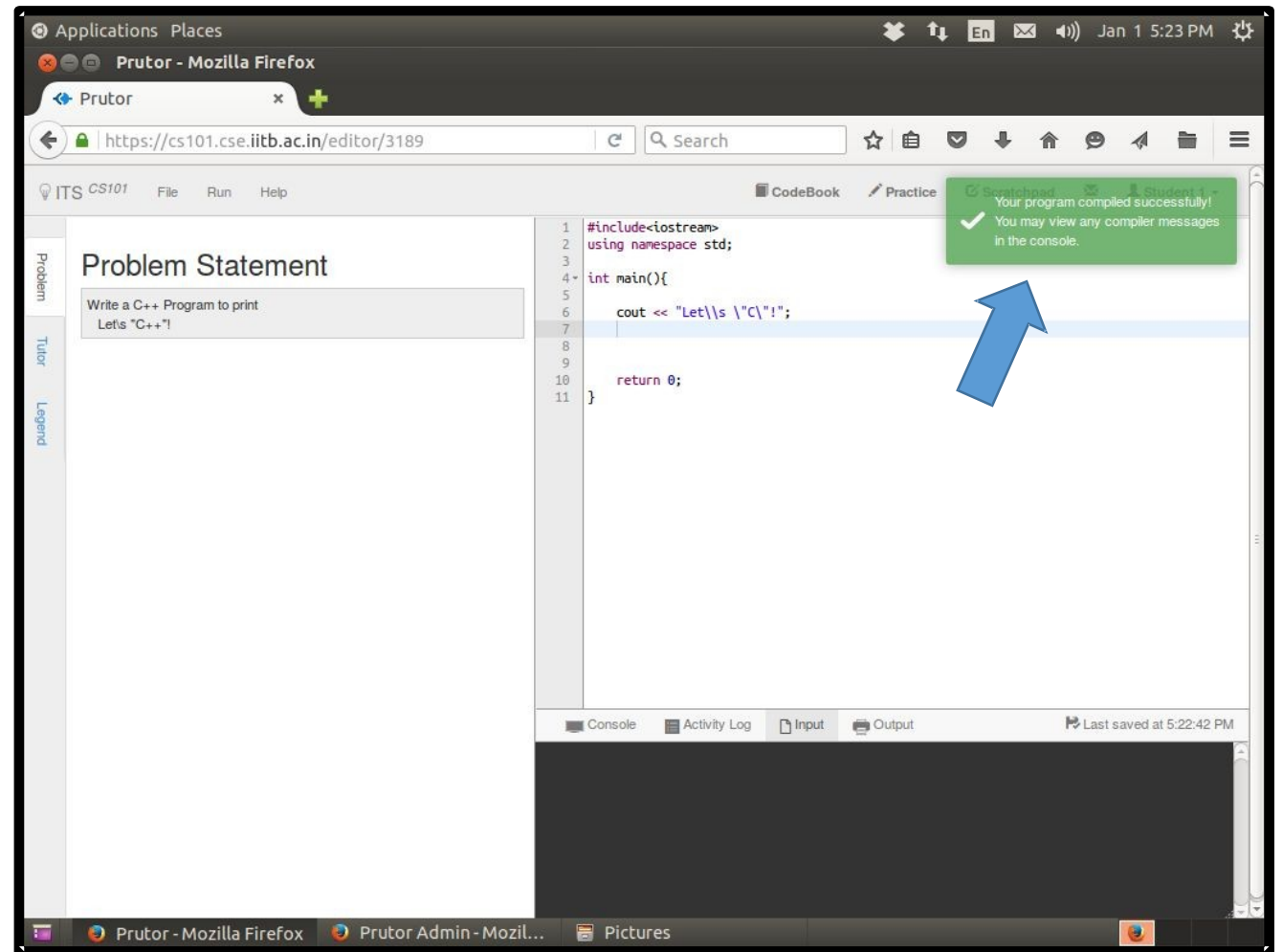
Executing Your Program

- If your program requires input data, then you need to provide it before executing the program.

- For providing the input, click on the Input tab below the Code Editor and enter the input as you expect it to be

- Click on Execute option in the Run tab to execute the program

- Note: Before executing make sure that your program has compiled successfully
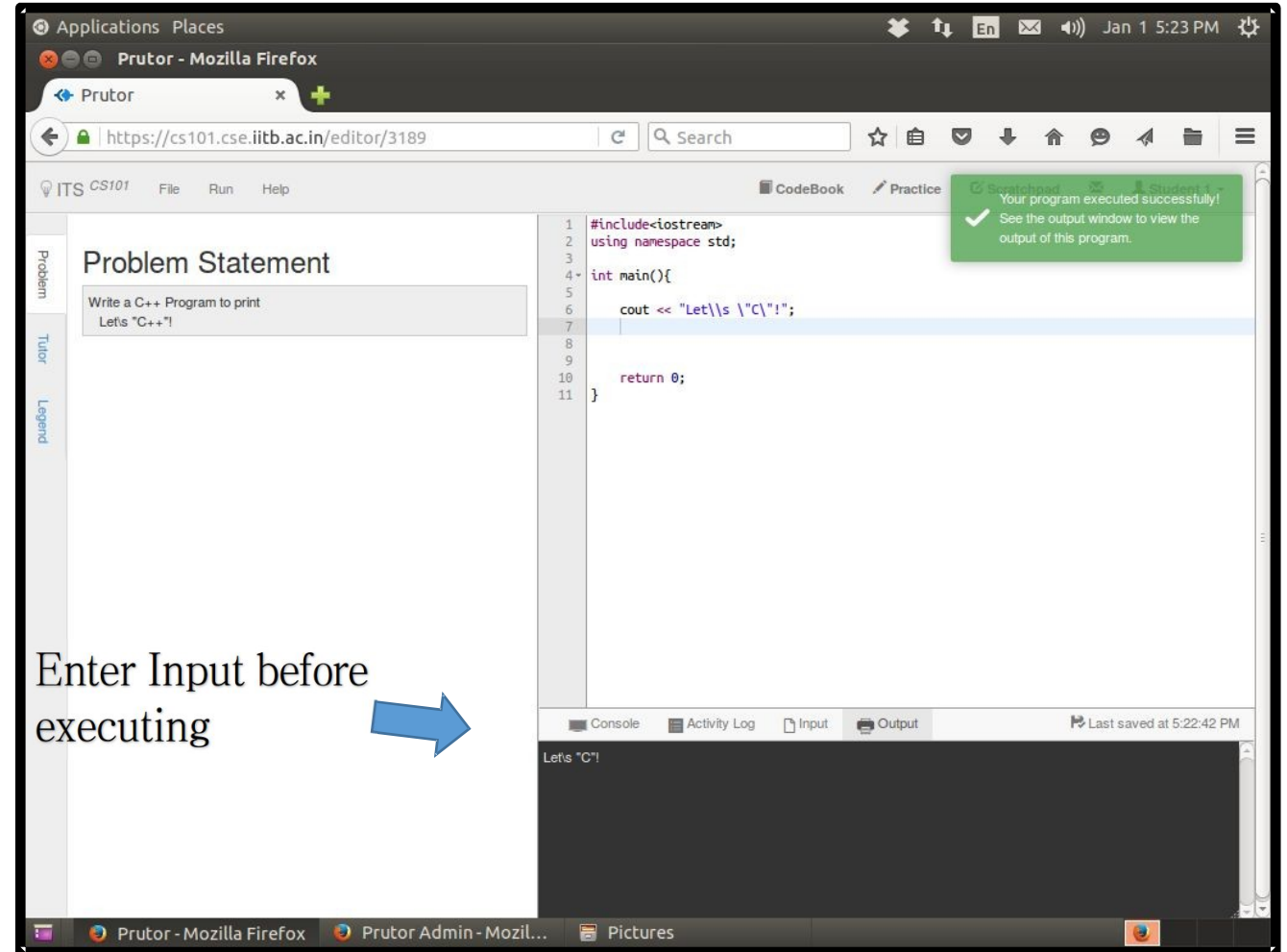
Viewing the Output

- After successful execution, a green floating notification will appear on the top-right corner and the output will be shown at the bottom
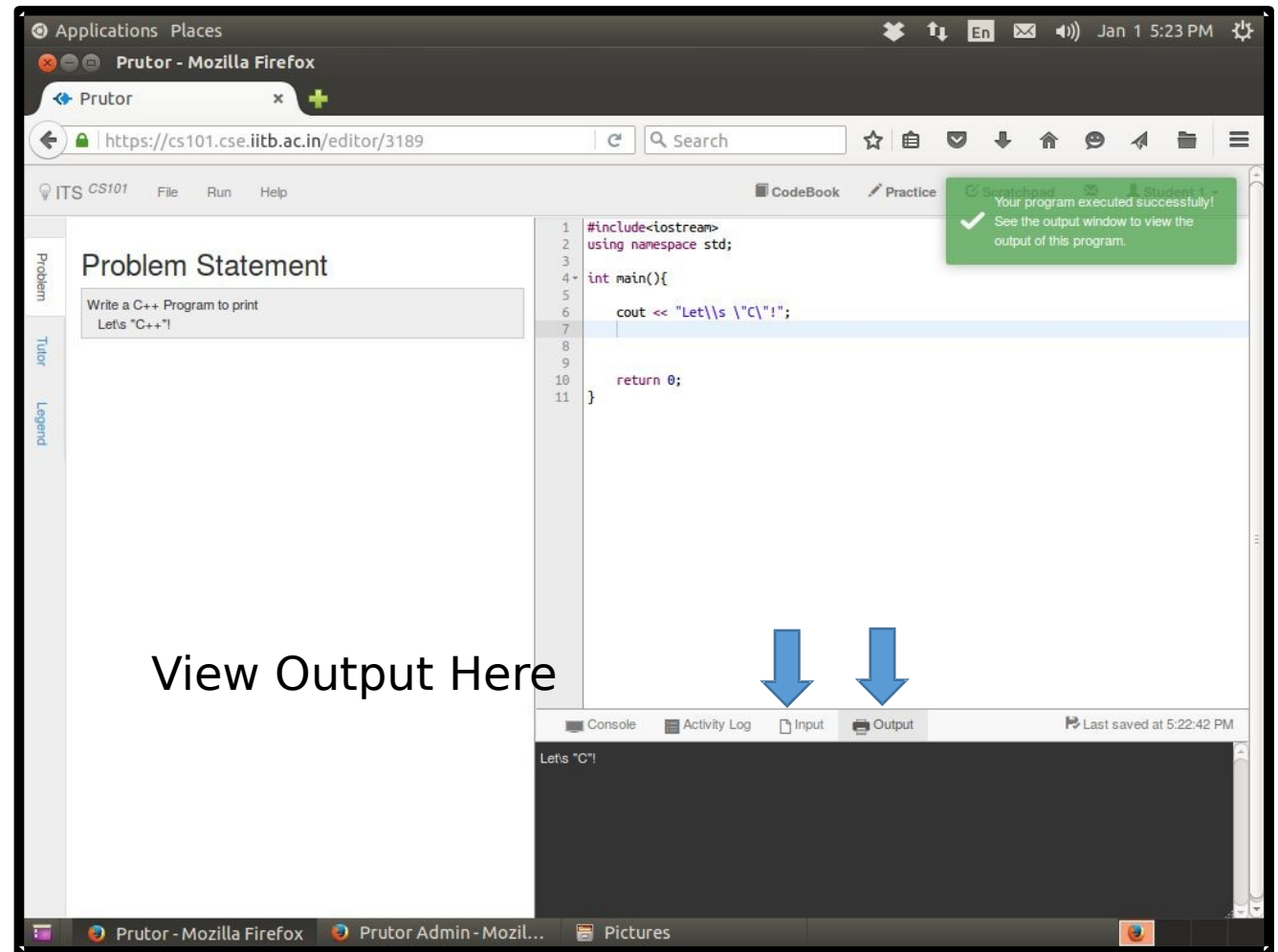
Evaluating Your Program

- Evaluation refers to running the program on the given test cases

- You can see some test case but there may be some hidden test cases

- Click on the Evaluate option in the Run tab

- We may evaluate your program on additional test cases created after submission

- Evaluation Results are displayed on the left column heading Tutor

- Note: You can submit your code even if it does not pass all the test cases.

Repeated Evaluation

- You can evaluate your program as many times as your like before submission

- This allows you to keep correcting your program

- If your program passes all test cases, a green floating notification on the top-right corner will appear with an appropriate message

Submit Your Code

- Never forget to submit your code, even if it is incomplete, before the event ends

- However, it is not necessary to submit a code to start attempting another question. But do save it before leaving

- Code that is not submitted can not be evaluated

- To submit, click on Submit option in the File tab

Homepage After Submission

- After submitting your code, the homepage will mark your question as submitted
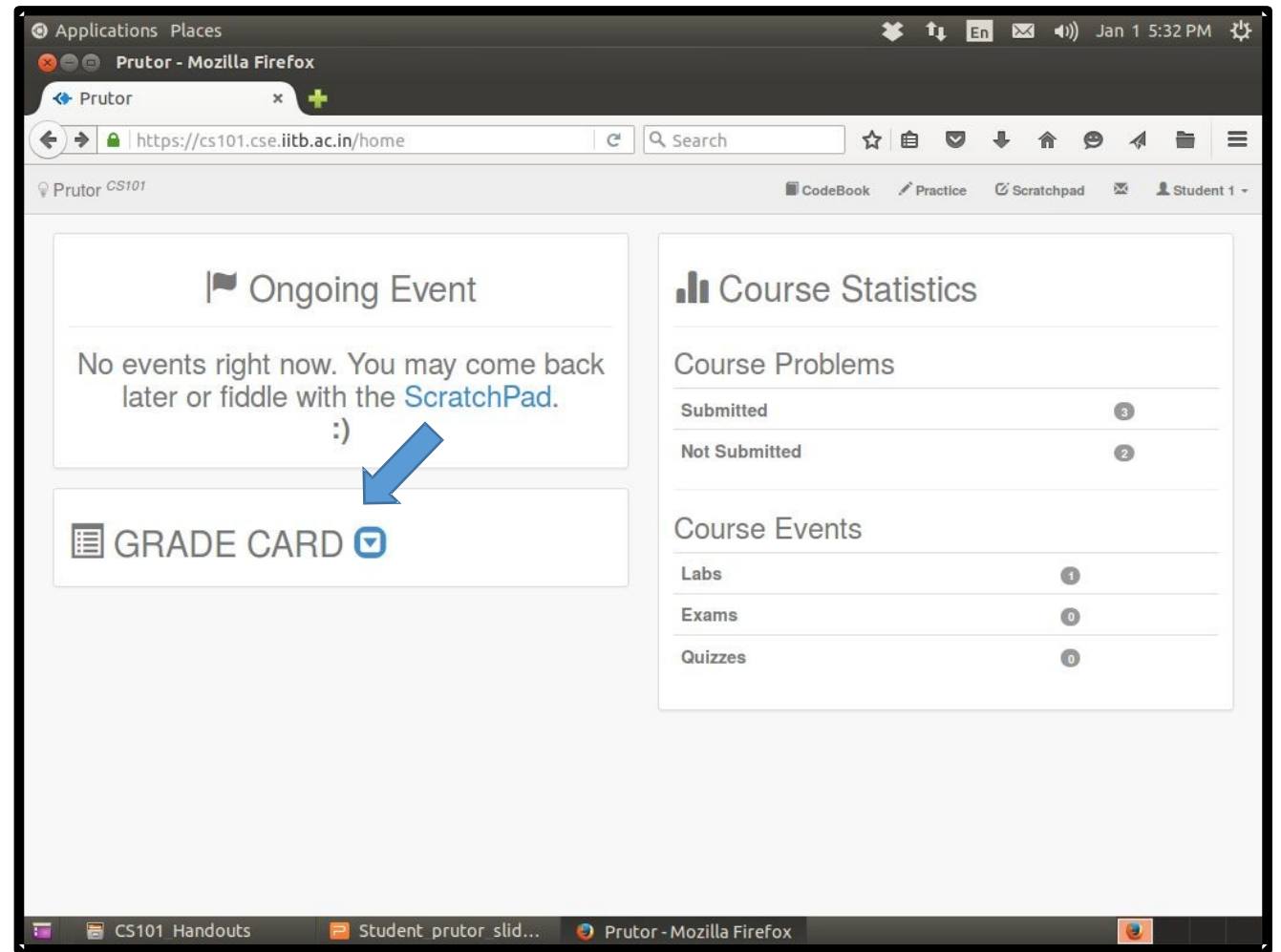
End of Event (1)

- You have limited time to write the code for all questions.

- You will be intimated through a floating notification when you have only 30 minutes left

- When the time is up, you will get a notification and you will not be allowed to edit the code any more

- You will be redirected to your homepage
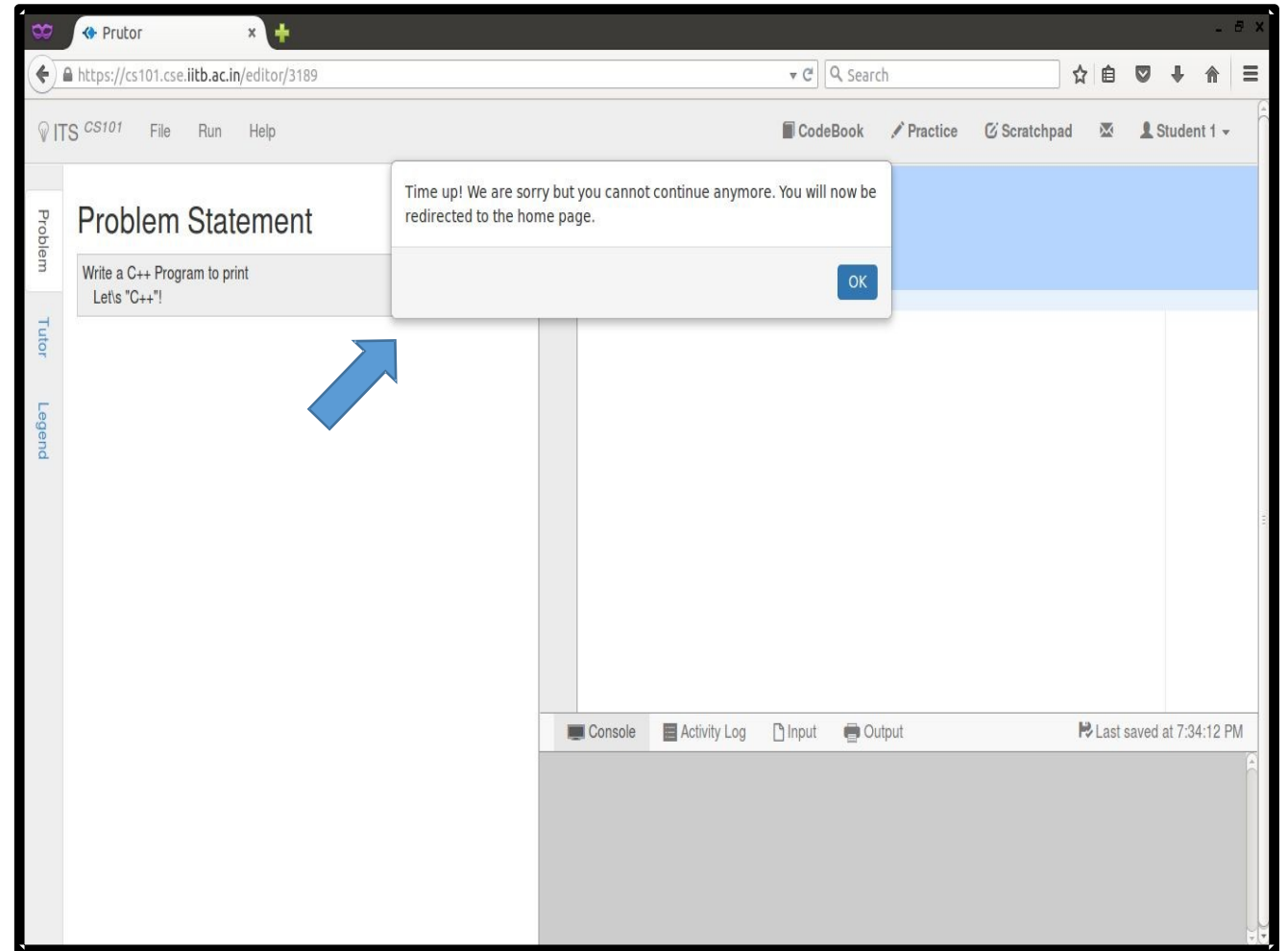
End of Event (2)

- After an event finishes, it stops showing on your homepage (i.e. it is not accessible to you)

- You can view your submitted code after the lab week is over (i.e. all batches have finished the same lab) by visiting your Codebook

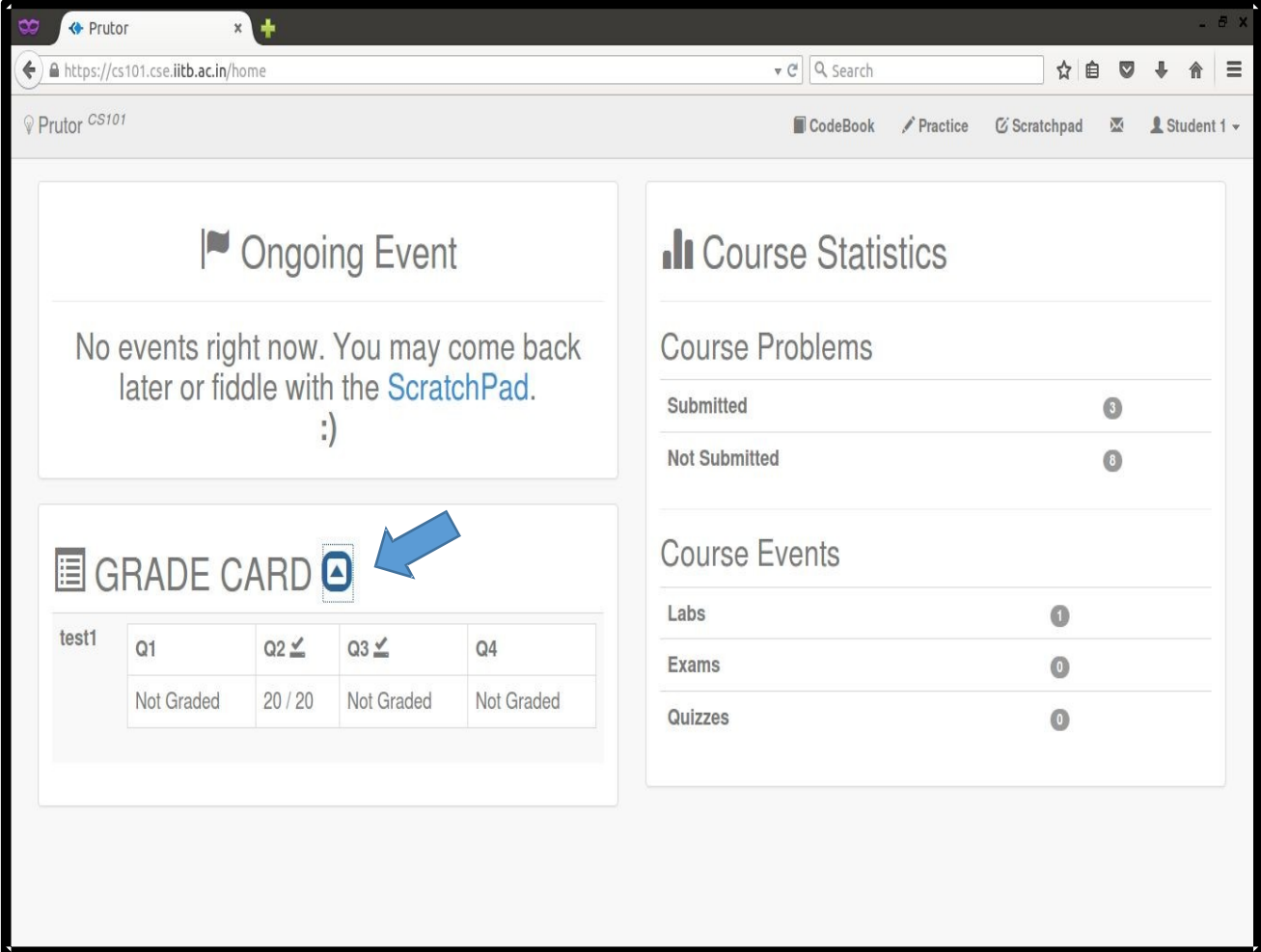- Once the grading is done, you can view your marks by clicking on the arrow button next the Grade Card

Graded Questions

- When your question is graded the marks are shown in the GRADE CARD box

- Only the Q2 is evaluated in this screenshot. The other questions are yet to be evaluated (or the student did not submit the code)

CodeBook (1)

- Here you can see the questions you have attempted in the test1

- Click on the question to see your submitted code and the evaluator's remarks, if any
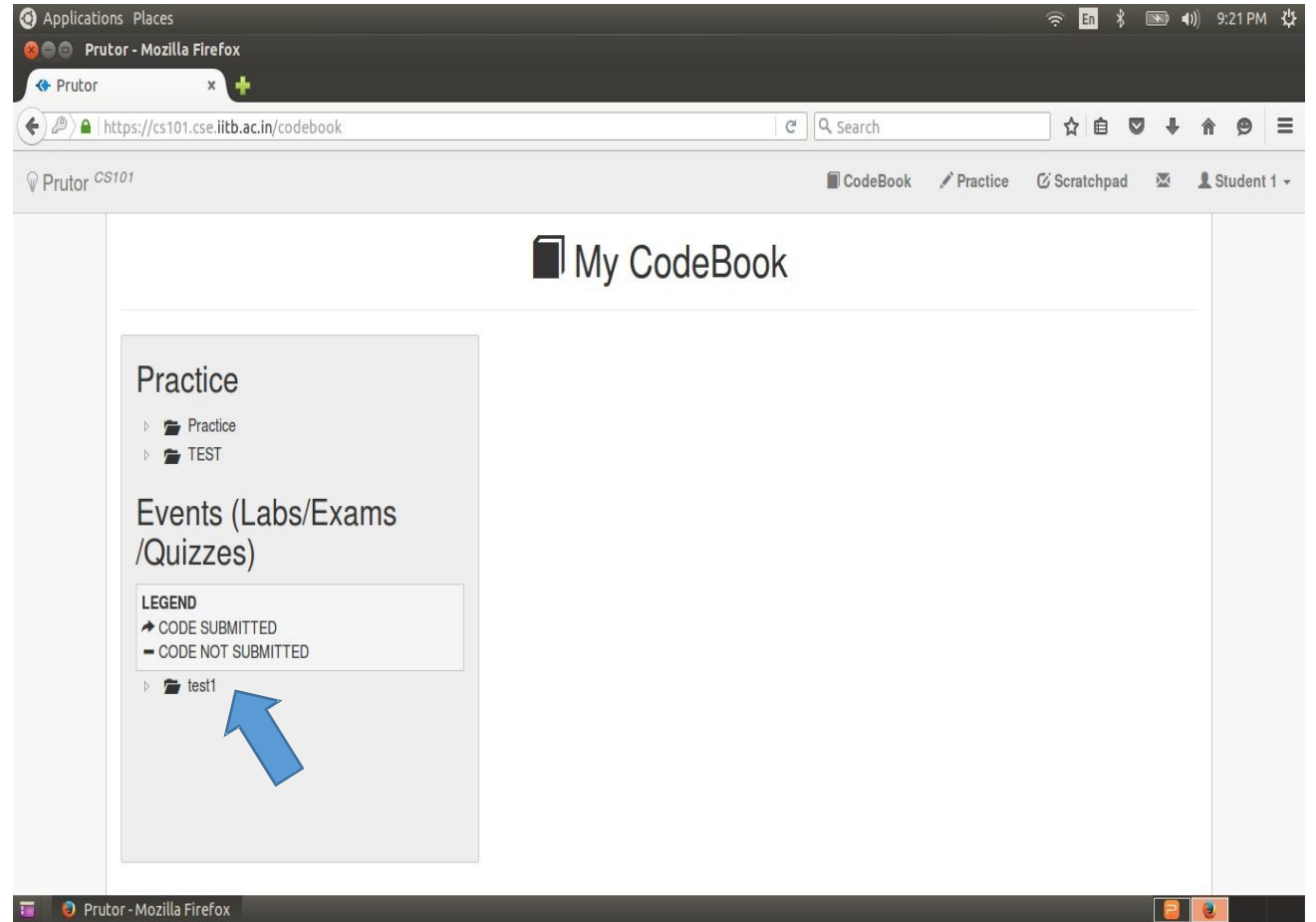
CodeBook (2)

- This page gives the details of the problems attempted and grading status, if the problem is part of a Graded Lab

- You can view the evaluator's remark at the bottom

- In the last text box you can send a Request for re-evaluation of your code

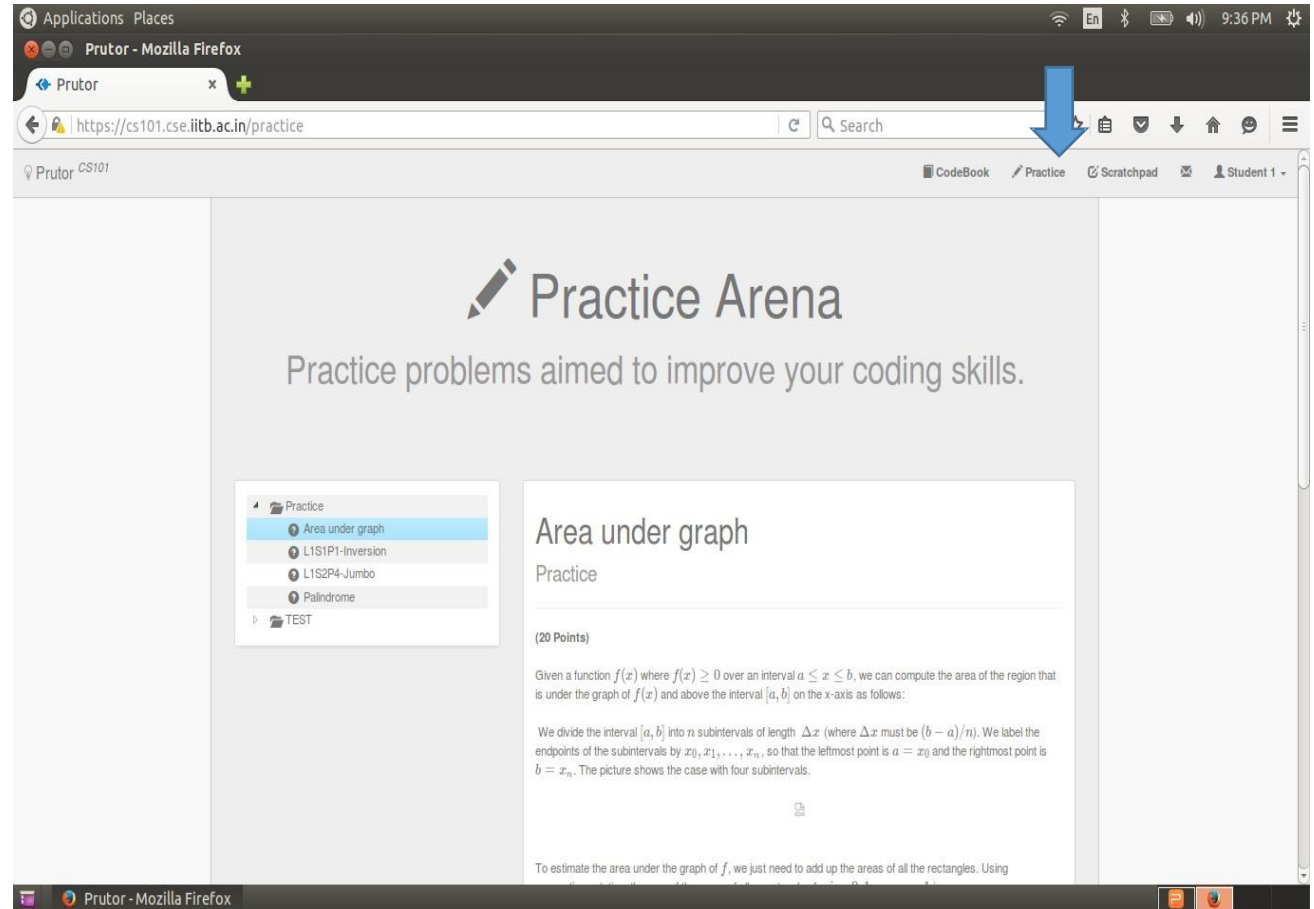- Re-evaluation request can be made only once for a question

Practice

- You can use the Practice Arena to practice programming

- Here you get the same Code Editor where you can Compile, and Execute your code.

- This section will contain practice problems (along with test cases, if any)

- The code in the Practice Arena is not for submission

Scratchpad (1)

- You can use the Scratchpad to practise programming on your own (no questions are provided)

- You need to create a file before you can start editing

- Here you get the same Code Editor where you can Compile, and Execute your code
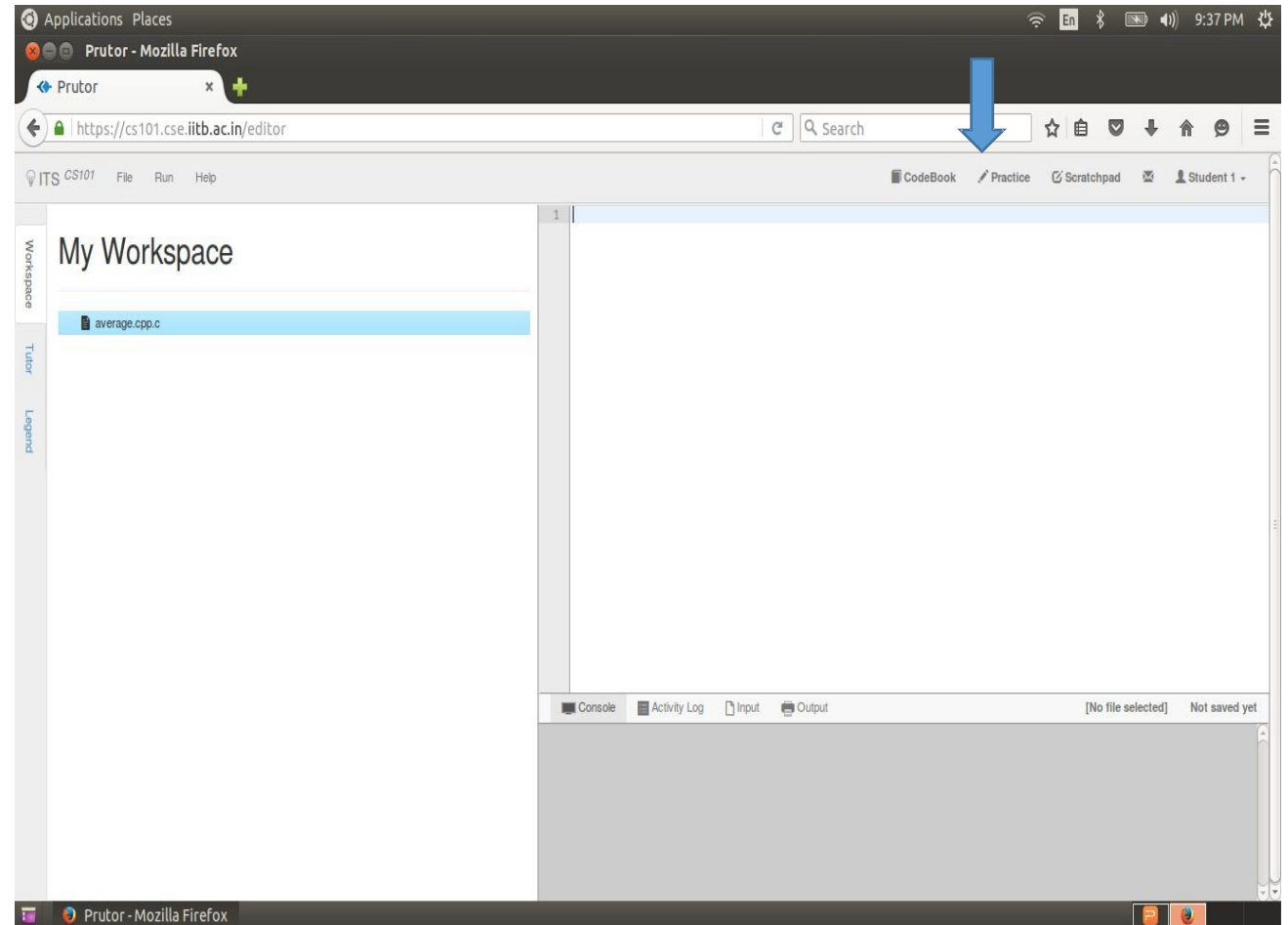
Scratchpad (2)

- You can add files and directories in the Workspace using the File tab

- If you don't know what files and directories are, don't worry right now. You will learn towards the end of the course. They are not needed right now
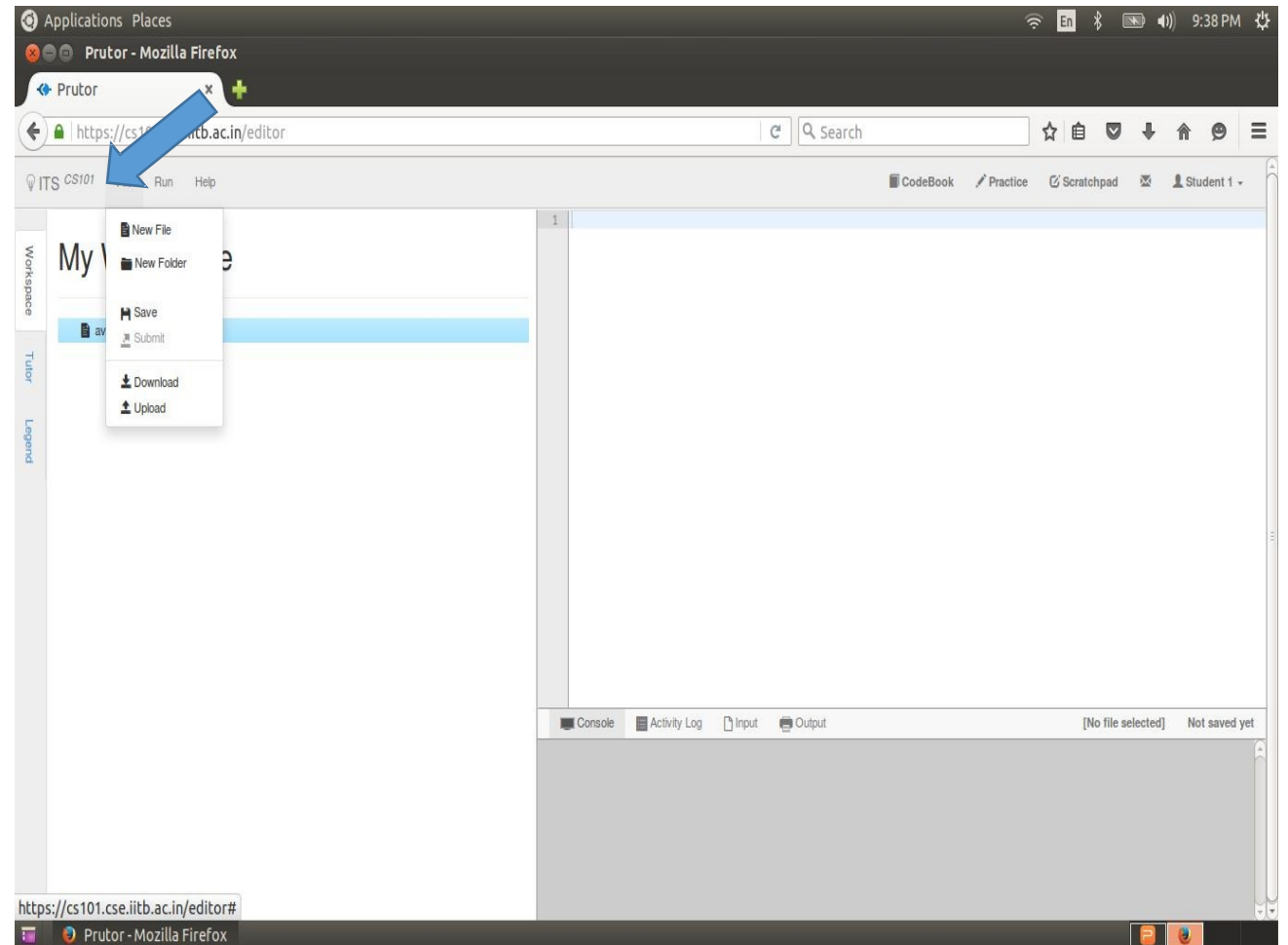
Scratchpad (3)

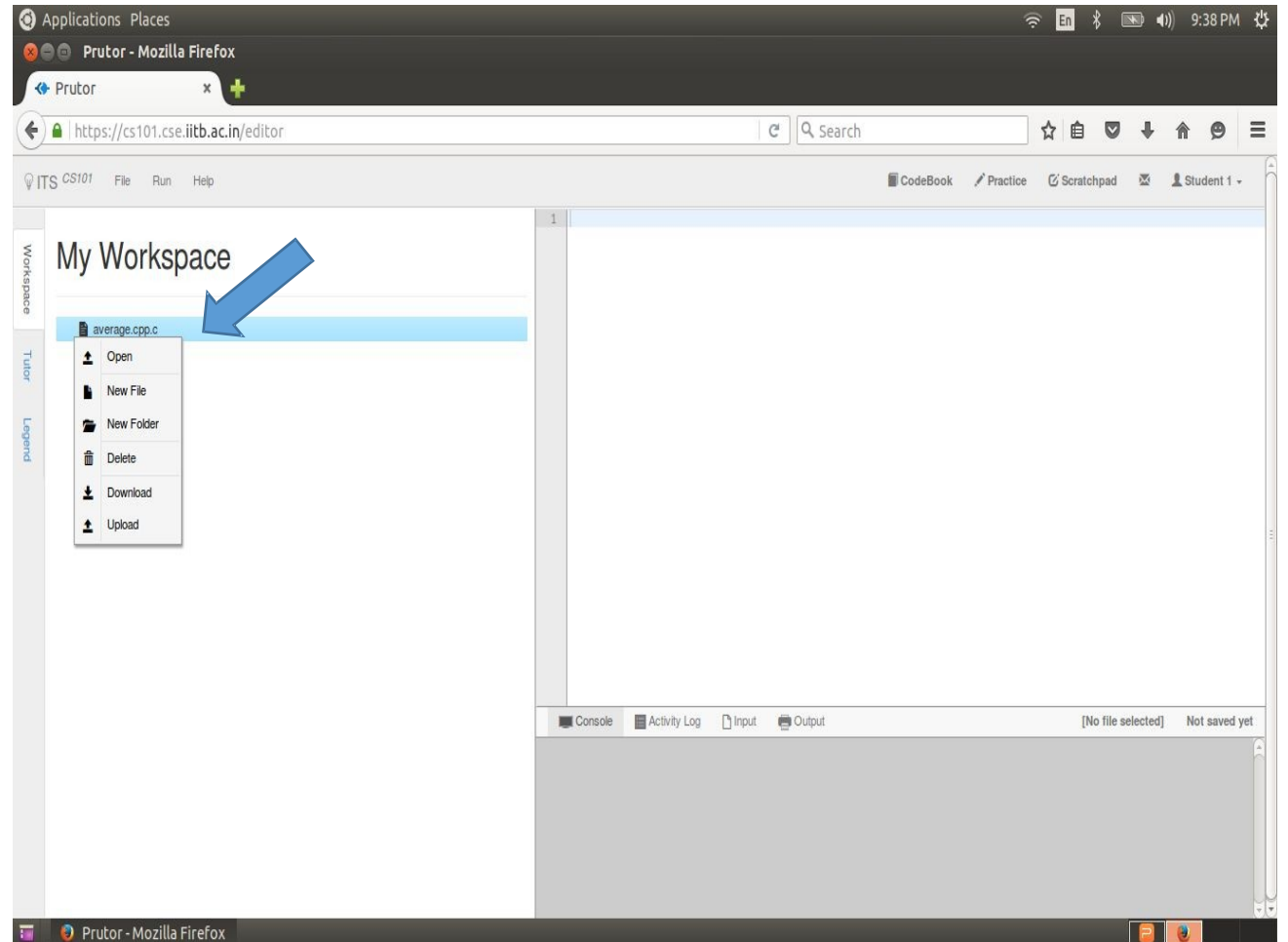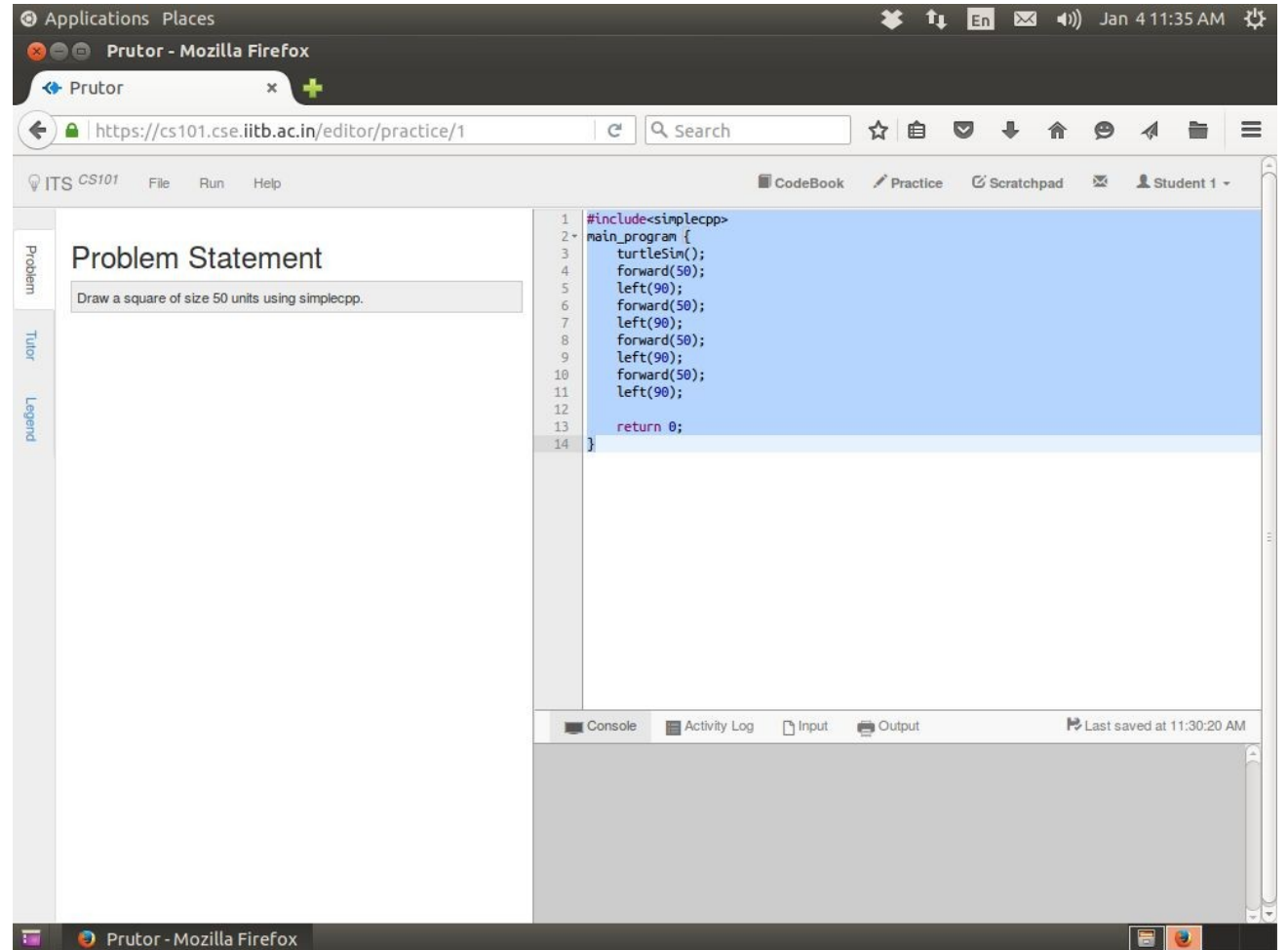- You can add files and directories to a directory created earlier by using a right click on the names in the Workspace

Creating Simplecpp Programs

- You reach here by clicking the Start Coding link of a question
- The Code Editor saves your code every 5 seconds
- You can also save your code by pressing Ctrl-s
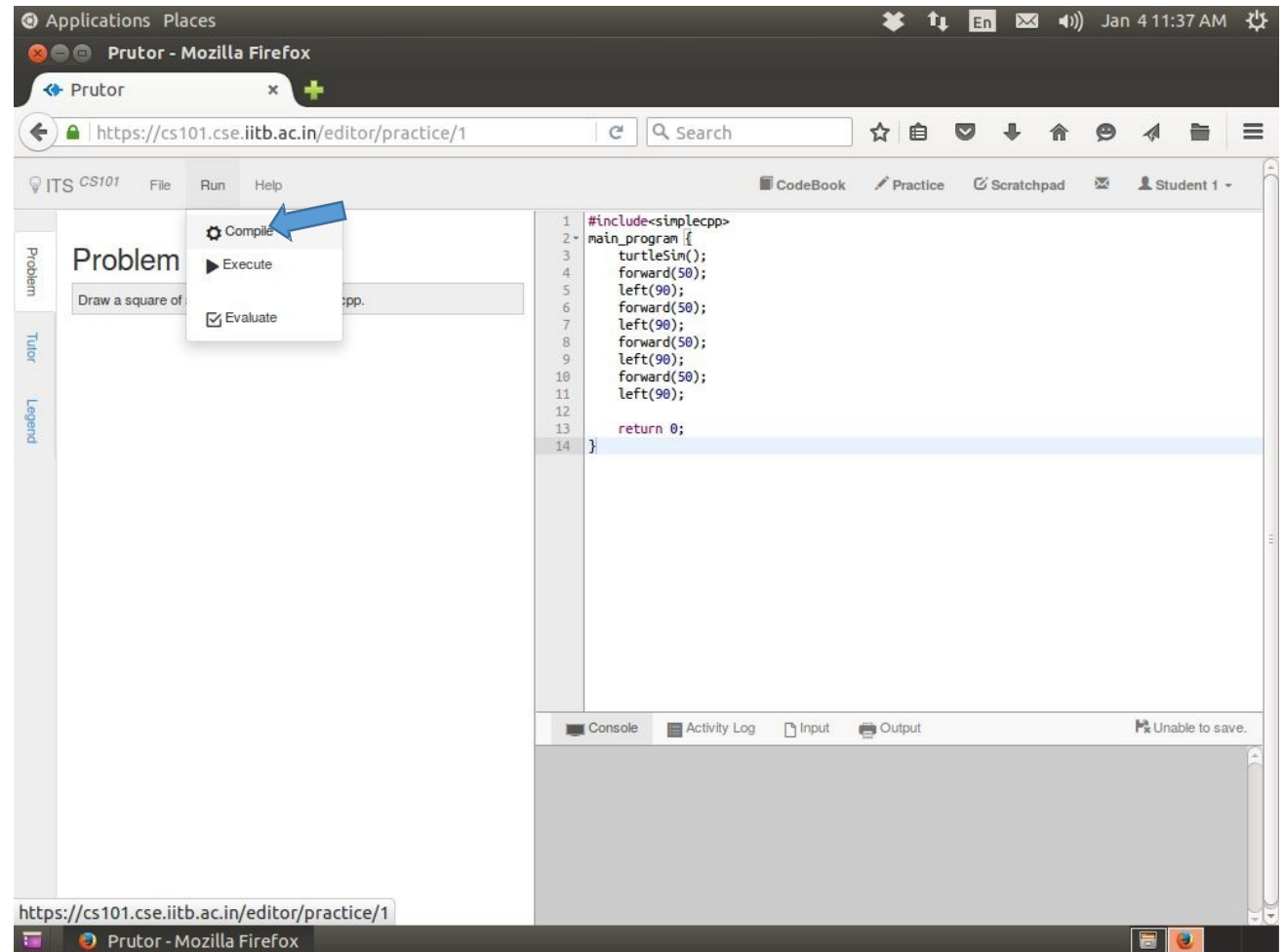- Or you can save using the Save option of the File tab

Compiling Simplecpp Programs

After writing the code, choose the Compile option in the Run tab

Successful Compilation

- The floating notification on top-right corner will notify you when your compilation is successful

- The color of the floating notification would be green
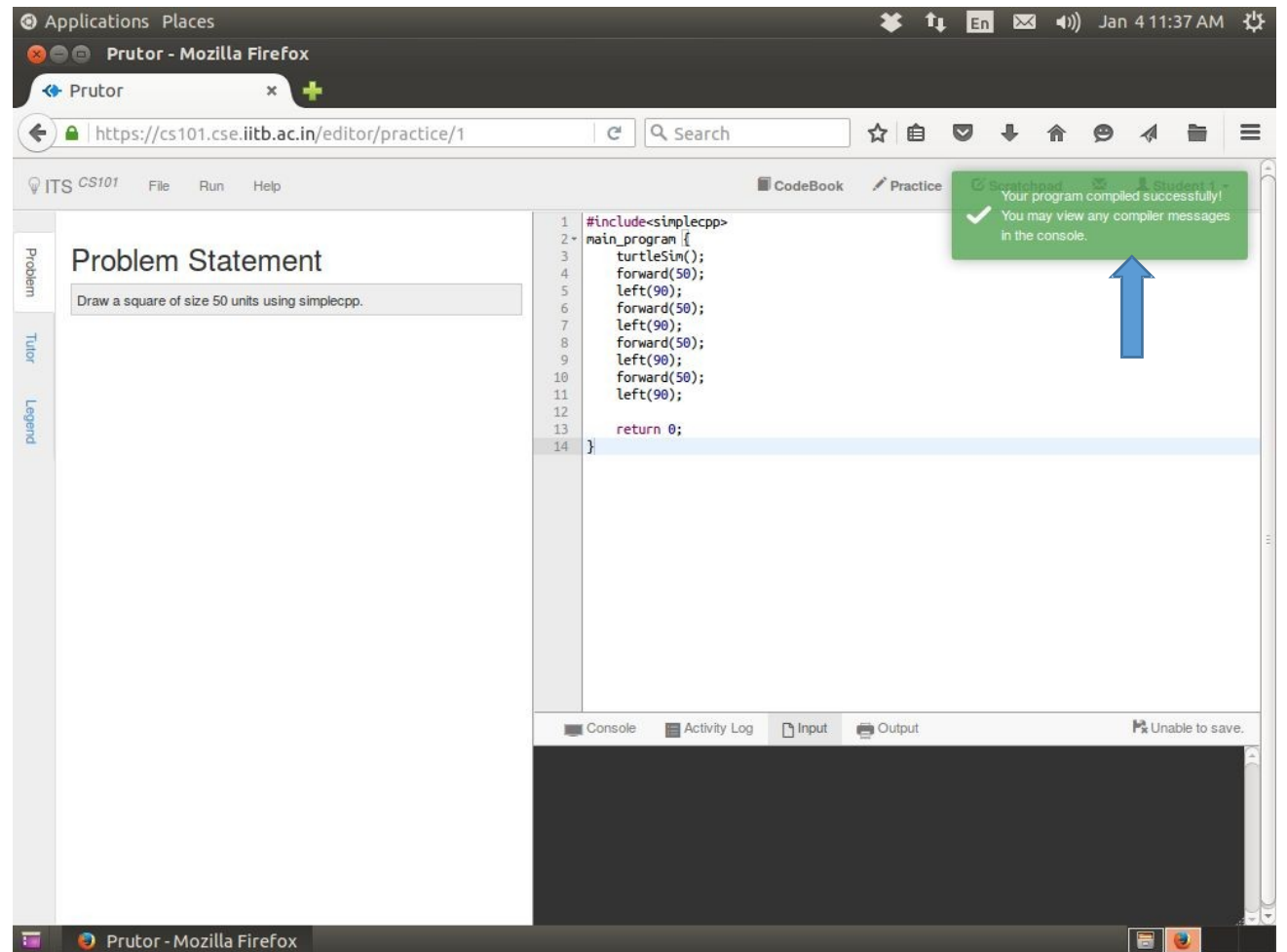
Executing Simplecpp Programs

- Click on Execute option in the Run tab to execute the program

- After successful execution, a green floating notification will appear on the top-right corner and a link will be shown at the bottom

- Click on the link in the output window to view the ouptut
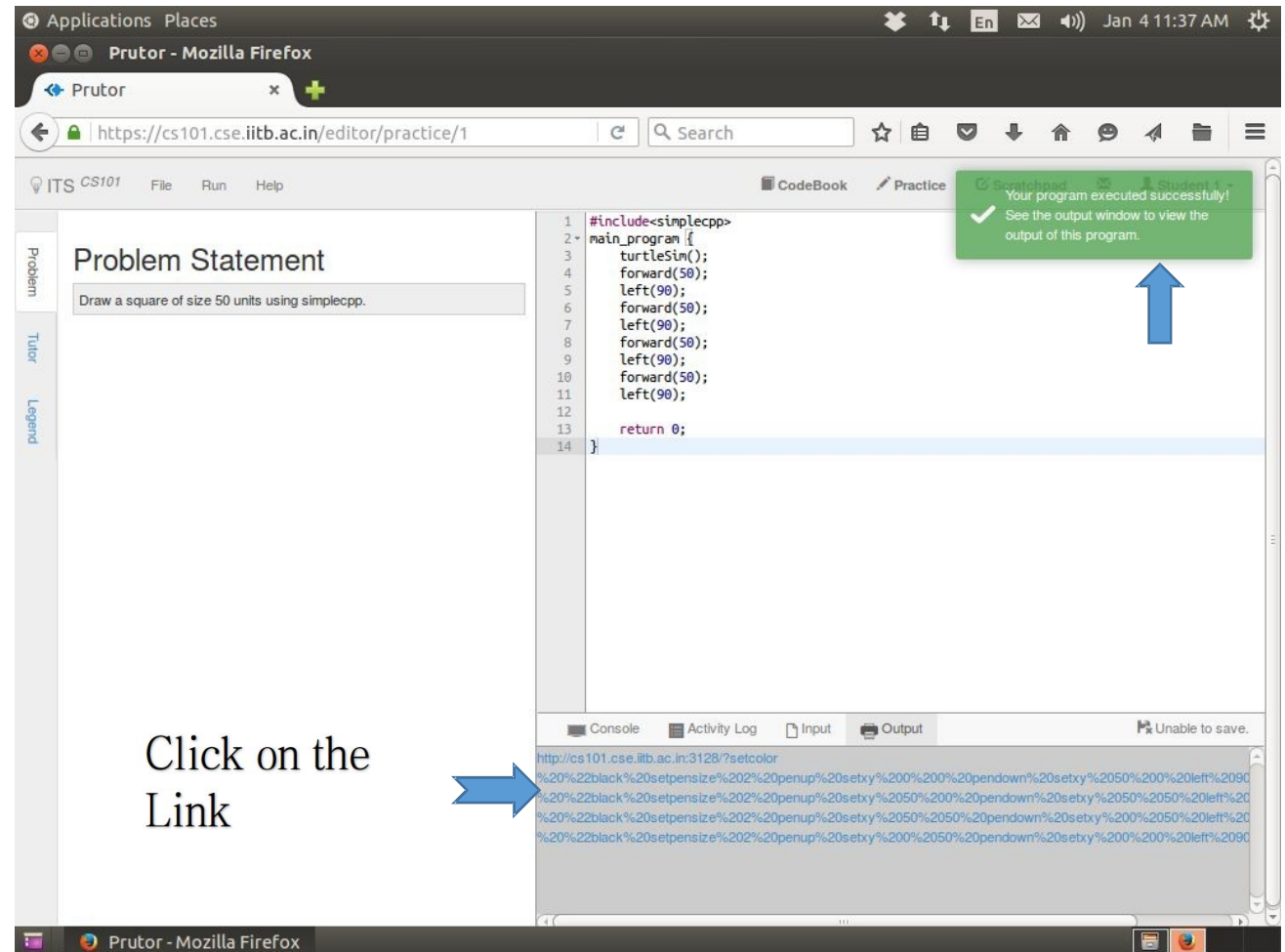
Click on the Link

Viewing the Output of Simplecpp Programs

- You reach here by clicking the link in the output window

Prutor Does Not Support Interaction
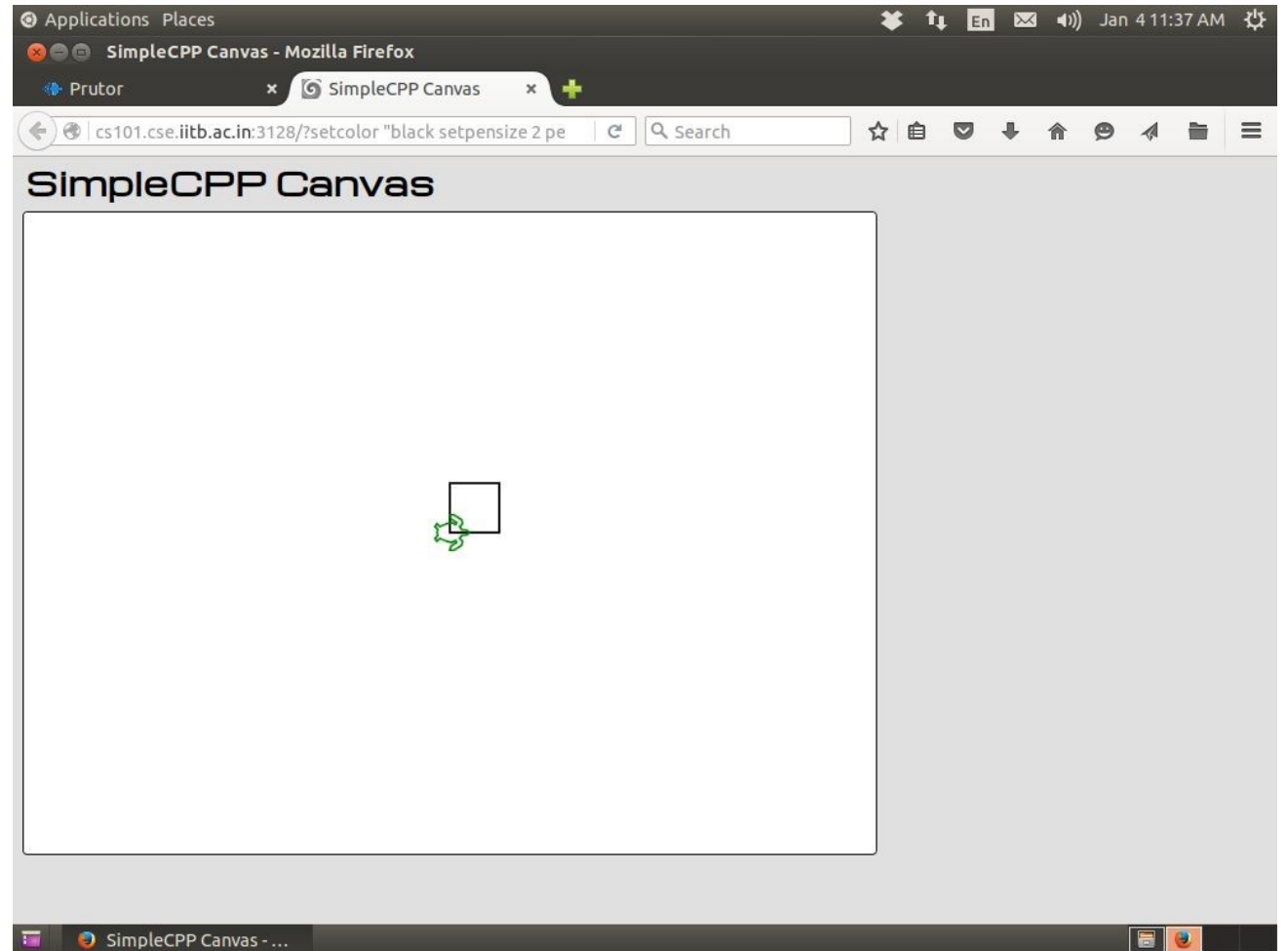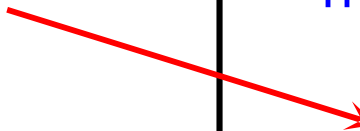
- We cannot interleave input and output

- We cannot expect to see the output
  No. of sides?

  and give the input after it

- The console window is first used as an input window and then the same window is used as an output window

- Do not print any thing extra in the input window or console other than the input needed by the program

- Such an interaction is supported in the command line execution of programs

```
#include <simplecpp>
main_program{
    turtleSim();
    cout << "No. of sides?";
    int noofsides;
    cin >> noofsides;
    repeat(noofsides){
        forward(10);
        right(360.0/noofsides);
    }
}
```

Thank You

Happy Prutoring!