More Autograder Management

Scenario

Once again, you are performing a much needed update to a particular autograder for a certain computer science course. This time, however, you are trying to make the reporting mechanism easier to interact with the specific student submissions. In particular, you would like to generate two types of reports for each submission--one report simply involves counting the total number of correctly passed test cases, while the other report determines the number of time-out errors in a submission. Begin wary of how the autograder has evolved in the past, you expect extra reports to be created.

Problem

Your task is to create a pair of Report classes which maintain a specific type of report for a Submission object. The Submission class has been provided, but you must create the Report classes. After every test case, the submission should inform the Report objects of the results (i.e., pass or fail). If the test case failed, the error report should ask the Submission object if the most recent test failed due to a time-out (using the *wasTimeoutError* method). You may add methods to the submission class as needed. Because the Submission object may be running on a separate thread from the Report objects, this is not simply a matter of running a test case, and then asking for the results.

For simplicity, reports can simply involve printing to the console.

Deliverables

1. Identify the design pattern you used to solve this problem, and the participants (i.e., the roles each class takes).
2. An implementation in a language of your choice.
3. A class diagram of your solution (including existing classes), so future developers can easily see how to work with your solution.