Tutorial Problem Set #8

Due: Wednesday, November 13, 2024, 11:59 PM

Policy

- Piazza questions on tutorial problems will be ignored or deleted. Questions will only be answered in your assigned tutorial section.
- Sample executables can be found in your 1249 git repository directory (run `git pull`).
- Completing the problem set will reduce the weight of the final exam by 0.5%. To complete a problem set, you must pass at least 50% of the public and secret tests.
- You may assume all input is valid. Tutorial problems **NEVER** require checking for invalid inputs.

Question 1

In this exercise, you will explore a modification of the Observer pattern, in which an observer may observe multiple subjects.

In this problem, the subjects are sources of news, each of which will get its headlines from a separate file (this code is provided, along with two sample files; you may wish to add headlines to these files for testing purposes). We have provided headline files for two news sources: TV and social media. For simplicity of searching, headlines are in all lower case, one headline per line of the file. There will be no blank lines.

The observer is the class `InformedCitizen` that watches both news sources and is looking for headlines that match their interest. In our provided `main`, the interest is `science`, though in a more general program, it could be anything.

Each time a source of news publishes a headline, the observer must check to see whether the headline contains their interest, and if so, the observer announces where they got the news from and what the headline was (see repo directory for sample output).

The key challenge, when dealing with observers of multiple subjects, is the observer will need to know which subject a notification came from. For this reason the `notify` method, which normally takes no parameters, now takes a parameter `whoFrom`, which is a pointer to the subject that called `notify`. This way, the observer will know which subject to request the state from.

You can check whether the string `s1` contains the string `s2` with the expression `s1.find(s2) != string::npos`.

We have provided the entire program except the class `InformedCitizen`; it will be your job to provide this class.

Submission

Submit the file `news.cc`.