Programming Assignment: `watch`

Description

https://pixabay.com/images/id-2564955/

Sometimes you are waiting for one or more users to login to the system. There are lots of reasons why you might want to know when someone logs in. You might want to start a chat session with that user. You might be waiting for that person to login and download a file you need.

The `who` command might be useful; it tells you which users are logged in right now. You could type 'who' every few minutes and see if the logname appeared on the list. That would be pretty tedious. The shell called tcsh has a built-in feature that watches for logins and logouts for a specified list of users. The tcsh manual page has all the details.

Instructions

You will write a program called `watch` that takes as command line arguments a list of users you want the program to watch for. Every five minutes the program wakes up and checks the utmp file. It compares the list of active users it finds there to the list of active users it found last time. If it finds a user is not logged on but had been logged on before, it should tell you. If it finds a user is logged on now and had not been logged on before, it should tell you. The interval of checking defaults to five minutes. If you prefer a different interval, you should be able to specify the interval on the command line. Under Unix, a user may login to several terminals at the same time, so the program should only report when a user goes from no logins to one or more logins or when a user goes from one or more logins to no logins. If you do not make this restriction, each time a user opens a new terminal window on a graphics workstation, you will be notified.

Specifications

The watch program you write must meet the following specifications:

[a] It takes one or more lognames as command line arguments. It watches for the comings and goings of the lognames specified on the command line. It does not report on users you do not specify.

[b] When the program starts, it prints the lognames of users on the given list who are currently logged in.

[c] It checks the utmp file every 300 seconds to see if anyone on the list of lognames has logged in or logged out. If the first command line argument is an integer, it uses that number as the number of seconds to sleep between checking the utmp file.

[d] The program reports when a user on the list logs in. A user is considered to have logged in if, when the utmp file was last checked, that user was not logged in anywhere, but now the user is logged in at one or more terminals.

[e] The program reports when a user on the list logs out. A user is considered to have logged out if, when the utmp file was last checked, that user was logged in, but now the user is not logged in at any terminals.

The program should produce output of the form (the ... indicates five-minute pauses):

```
% watch Alice happy Bob fido king susie
Alice happy are currently logged in
 ...
happy logged out
susie logged in
...
happy king logged in
... susie logged out
fido Alice logged in
...
```

If you invoke the program with

```
% watch 120 Alice happy Bob fido king Susie
```

then it checks for changes every 120 seconds. In practice, one would run the program in the background by typing a command with a trailing ampersand:

% watch 120 betsy happy maya fido king susie &

[f] The program buffers disk access to the utmp file by using the functions in the file utmplib.c.

[g] The program exits when the person who launched the program is no longer logged in.
(There are several ways of doing this part.)

Submission

The submission structure is as follows:

1. Complete your draft of the programming assignment.
2. Use feedback received from your instructor to improve your code.
3. Complete your final version of the programming assignment.
4. Submit your final version.

Deliverables

Use the following naming convention for your files:

- **Draft:** LastnameFirstname-Coursenumber-Assignment title.txt
  (e.g.: SmithJane-CS43203-watch-Draft.txt).

- **Final**: LastnameFirstname-Coursenumber-Assignment title.txt
  (e.g.: SmithJane-CS43203-watch-Final.txt).
    - ***Output****: Include a screenshot of the sample output to show the program running.*