CO545 Spring Term 2019–20 – Assessment 3 - Test A

This class consists of an in-class test, of duration 40 minutes (scaled to reflect any "extra time" allowances in your ILP). The test is based on the first six lectures on concurrency.

Assessment conditions

You will not be permitted to leave before the end of the session. If you anticipate finishing early it would be wise to bring a book to read to fill in the remaining time. **Even once you are finished you must abide by exam conditions.**

The assessment will be conducted under exam-like conditions and violations will be treated under the University's disciplinary procedures. The primary condition is that you must not seek the help of anyone else in completing this assessment. Please make sure you strictly follow these rules:

- Switch off your mobile phone and music players. You may use a programming environment, such as NotePad++, emacs, VS Code, or similar.
- There are restrictions on what you may or may not access during the exam:
  - You may use a web browser but you must not access Web sites outside `kent.ac.uk` with the sole exception of the documentation at:

    `http://erlang.org/doc/apps/stdlib/index.html`
  - You may access the CO545 Moodle page.
  - You may access your local file store and past assessment work.
  - You may use and access any of the course resources, such as slides, class work, and assignments.
  - You may also use any notes that you have made in lectures.
  - You must **not** search the Web for answers or to look up compiler error messages, for instance.
  - You must **not** access files via dropbox.com, google.com, or any other file-sharing web site.
  - You must **not** use any form of email or messaging during the session.
- You may use your own laptop provided that you allow class supervisors to view the screen at all times during the sessions. It is your responsibility to ensure that your laptop is adequately charged for the duration of the session.

If you are in doubt about what might be permitted, ask the session supervisor.

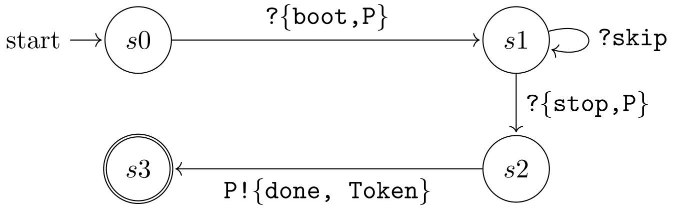
Once you have completed the assessment, which you should complete in a single Erlang (.erl) file, you should upload it to the Assessment area on the CO545 Moodle page.

Any attempt to submit after the end of the session will result in a mark of zero being awarded and might result in disciplinary action being taken.

Please leave your test script on your desk at the end.

Question 1 (15 marks)

(a). Implement the following CFSM as an Erlang function called `server` (which takes a single parameter `Token`):



Hint: you will need to define two functions because of the loop at state 1.

(8 marks)

(b). Define a client that interacts with the server. Assume at the start of the interaction that the server is in the start state. Your client should then exercise all paths in the CFSM of the server, leaving the server in the accept state.

The client should print (using `io:fwrite`) any messages it receives from the server, e.g. `io:fwrite("Got ~w~n", [Data])`.

(5 marks)

(c). Define a function which spawns the server, with any value for its first parameter, and a client interacting with the server.

(2 marks)

Question 2 (10 marks)

The following is a sequential function in Erlang:

```
testA(X, {F, G}) ->
    A = F(X),
    B = G(X),
    {A, B}.
```

The code has independent subcomputations bound to intermediate variables `A` and `B`.

(a). Implement a data parallel version of the above function which calculates the same results but performs the independent subcomputations (for `A` and `B`) concurrently, defining a single notion of a worker process which is used to create both independent subcomputations.

(9 marks)

(b). Define a function `test` which tests your solution with suitable inputs so that you can compare against the sequential version given in the question.

(1 marks)