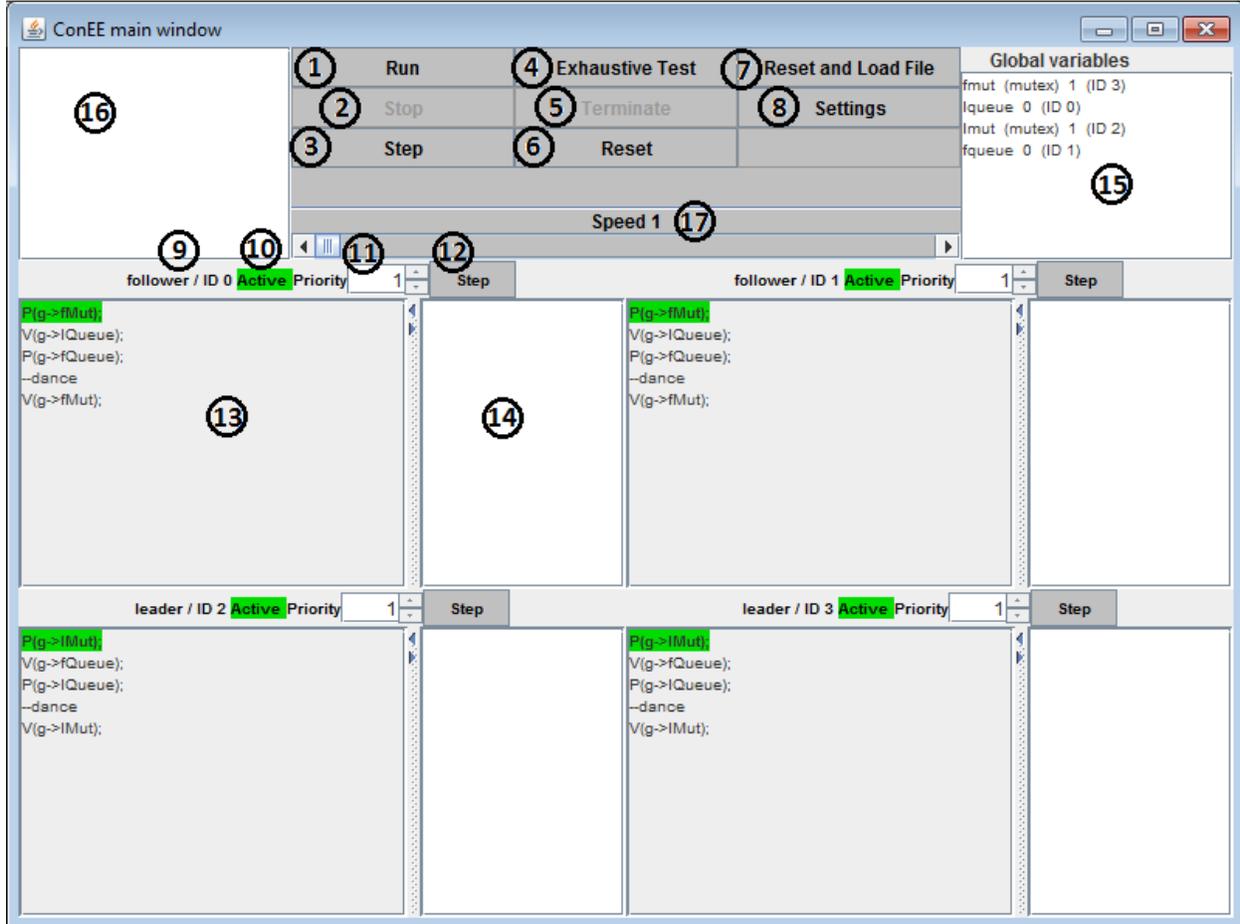


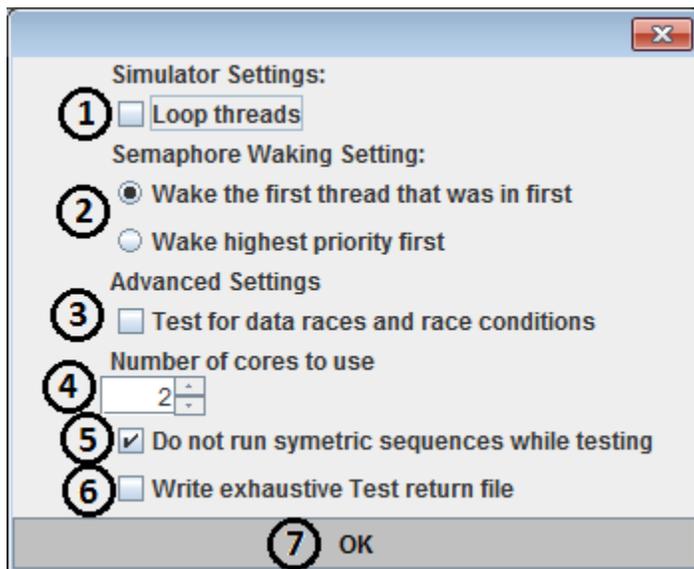
ConEE Main Window



- 1) Run
Systematically steps a random thread until it either is stopped or runs out of threads to step.
- 2) Stop
Becomes available when running the simulator. Stops the running and makes all the other buttons available again.
- 3) Step
Has one random thread execute a line.
- 4) Exhaustive Test
Begins enumerating and running every possible combination of the loaded program.
- 5) Terminate
Stops the Exhaustive test.
- 6) Reset
Resets all the threads program counters and variables.
- 7) Reset and Load File
Resets all settings and opens the navigation dialog to allow the user to open a text file which ConEE will attempt to parse.
- 8) Settings
Opens the settings window (see below)
- 9) Thread ID

- Displays the name and ID number of the thread
- 10) Active Label
Displays whether the thread is active, completed, or waiting.
- 11) Priority Spinner
Displays the priority of the thread. A higher priority thread will be executed more often when Run is active or when clicking the Step button, and will be woken first when the Semaphores are set to Waking Highest Priority first (see below)
- 12) Local Step Button
Steps the thread that it is attached to.
- 13) Code Display Window
Displays the code for the specific thread.
- 14) Local Variable Display Window
Displays the local variables for the thread and their values.
- 15) Global Variable Window
Displays the Global Variables and their values.
- 16) Output Console
Displays all output statements.
- 17) Speed Dial
Controls the speed at which the simulator steps processes when running.

Settings Window



- 1) Loop Threads
Checked: When using Run or any of the Step button the threads will loop when they reach the end of their execution;
Unchecked: When using Run or any of the Step buttons the thread will go inactive when it reaches the end of its execution.
- 2) Semaphore Waking Setting
Selects the waking method for all the Semaphores.
'Wake the first thread that was in first' has the Semaphores wake the threads in the order that they waited on the Semaphore

'Wake highest priority first' wakes the highest priority thread that waited on the Semaphore first.

3) Test for data races and race conditions

When checked the exhaustive test will compare the results of the variables (including Semaphores) at the end of each combination and check to see that they always come out the same.

4) Number of Cores to Use

Used for Exhaustive testing. Determines the number of threads that the Exhaustive Test starts. Best results come from setting it to the number of cores the computer uses.

5) Symmetric Sequence Settings

An optimisation setting for the exhaustive test. Say you have two identical threads: A and B. They have the same code: line 1, line 2.

Possible combinations:

A line 1	A line 1	A line 1	B line 1	B line 1	B line 1
A line 2	B line 1	B line 1	A line 1	A line 1	B line 2
B line 1	A line 2	B line 2	A line 2	B line 2	A line 1
B line 2	B line 2	A line 2	B line 2	A line 2	A line 2

Now if we consider the fact that A line x and B line x are the exact same and look at the table again:

line 1					
line 2	line 1	line 1	line 1	line 1	line 2
line 1	line 2	line 2	line 2	line 2	line 1
line 2					

We can see that there are in fact only two combinations:

line 1	line 1
line 2	line 1
line 1	line 2
line 2	line 2

By checking 'Do not run symmetric sequences while testing' the Exhaustive Test skips all sequences which it knows are going to be identical to ones that it has already run. This option is not available if ID is used in the program because that means that the threads are not identical.

6) Write Exhaustive test return file

Creates a .CSV file in the folder where ConEE is run from. The file which contains all the information that the Exhaustive test returns upon completion, with the inclusion of the begin and end times, as well as the elapsed time. If an error is thrown, it contains the execution sequence that gave the error.

7) OK

Closes the settings window.