Exercise 1.

Get familiar with VCC by going through the following modules of the VCC tutorial handed out during class.
- Module 1: Investigation of the Voicemail Pager System
- Module 2: Implementation of Blackbox C++ Behavioral Virtual Components Using Embedded Wait API
- Module 3: Creation and Evaluation of Target Architectures for the Voicemail Pager

At the end on task 16 and 27 you will get an info0.txt in the results directory of the cell you use to run you analysis session. You will need to turn them in via e-mail. The tutorial is not difficult, but some attention to details is necessary to avoid getting huge amount of error messages.

Exercise 2.

Consider the following block diagram:

\[ a \to C \to F \to b \]

Signals a and b consist of a sequence of events each carrying an integer value. The events are generated randomly, both in time and value. Block C awaits a pair of input events (one from each port) and compares their values. After each comparison, it outputs an event carrying the value of the larger between the two inputs. (You will have to implement a well-defined behavior with respect to the possibility of receiving several consecutive events on the same port, e.g. you might discard all but the last one, and use it in the functionality. You have complete freedom on what policy to implement, as long as it is consistent.)

Block F counts the number of input events and outputs one event (with value equal to the last input event) every three input events received.

1. Describe each block in a different way (Blackbox C++, Whitebox C or STD) and turn in a printed version, together with some comments on what you did.
2. Check that your specification is correct, compiling the code and running Functional Simulation (as in Task 16 of Module 2).
3. Send the info0.txt files to the TAs via e-mail.