This homework assignment tests your ability to utilize modeling techniques to describe problems for which there are already solutions. You can create your models using any drawing package you like, only printed documents will be graded. The grade weight for each question is equally distributed.

1 CMake, again?

On the course webpage, you will find a lecture with examples from CMake (linked from the homeworks page). You will need to read and understand this code and the various configuration files, for the following questions.

1.1 Structural models

Create an object diagram for the project created in the first version of the project indicated in lec11/CMakeLists.txt. This object diagram should be with regard to the files in the project, not the objects in the code in those files. You can use whatever names of classes you like, but ensure that the object names correspond to features found in CMakeLists.txt.

1.2 Behavioral models

Describe a state model for the project generation and compilation process of the project in the first lec11/CMakeLists.txt. Here, you should investigate the various stages of the Makefile generation, make execution, and execution of the compiler. This will be a complicated state model, but not so complicated that it won’t fit on one page. You are modeling only the state of the build process, and not the state of the various compilation and preprocessing processes. Remember to have the events and actions, do activities, entry activities, etc., have names that correspond to actual processes and events in the process. For consistency, use bracton.ece for your experiments, or you can use any command line interface that can generate Makefiles (e.g., Cygwin).

1.3 Interaction models

The tutorial begins with running example01 and demonstrating leaked memory. Construct a sequence model that explains the creation and execution of example02, example03, and example04 (each sequence model should be separate, and may be further decomposed to save space or make the design easier to present).

As part of your sequence model, mention the files that must be created, edited, and various calls to the compilation and project synthesis process. Make sure you show calls to the memory check tool, as presented in the tutorial. Here, your events to/from objects should clearly be based on text from the tutorial.

Finally, please maintain causality with the tutorial: you may not edit CMakeLists.txt for example03 while demonstrating the sequence model for example02.

These interaction models will be complicated, and may require more than one sheet of paper, or may require you to decompose behaviors in order to make the sequence model easier to understand. Use your judgement here, and remember that someone who sees your interaction models should be able to perform those actions.