Homework #3

Pull the initial files for this homework into your private git repository by changing into that directory and executing the command "git pull". Then complete the problems below, add, commit, and push the results to your GitHub repository.

1. (10 points) Draw a UML class diagram to illustrate the relationship between the classes *MEMORY_pvt* and *tlm_target_socket*. You may ignore attributes and methods and focus only on the high-level structure. Be sure to include enough additional classes so that the two classes are connected. You will find all necessary code in the *hw03/p1* directory, except for the *tlm_target_socket.h* file, which is in the following directory:

   ```
   /afs/eos/dist/ds5-2013.06/FastModelsTools_8.2/OSCI/TLM/include/tlm/
   tlm_h/tlm_sockets
   ```

   Save your diagram as a .gif, .jpg, .png, or .pdf file in the *hw03/p1* directory. Then add and commit this file to the repository.

2. (10 points) List all of the common dependencies for the attributes (*i.e.* member variable types) for the classes *SimpleBusAT* and *traffic_generator*, which are defined in the files found in the *hw03/p2* directory. You may ignore methods (*i.e.* member functions) and base classes. For template instances, list the template and its type-parameters separately. Update the file *common_dependencies.txt* in the *hw03/p2* directory with your answer. Then commit this file to the repository.

3. (30 points) In the *hw03/p3* directory, you will find a file called *LMS_pipe.hier*. This file contains a simplified Verilog hierarchy, including modules and the instances that they contain, in the following format:

   module [module name]
   [module name] [instance name]
   [module name] [instance name]
   . . .

   Some modules contain other modules, but other modules do not, such as the module FAX1. The latter are called "leaf-cells", because they are the leaves of the tree structure. Note that the root of the tree structure is the last module defined in the file. Note also that the leaf-cells do not have a "module" line in the *LMS_pipe.hier* file.

   The *module.h* and *module.cpp* files define a class to store each module and the list of instances it contains. The *instcount.cpp* file contains code to read the *LMS_pipe.hier* file into a map data structure and print it out again. Note that the order of printing is different, and that instance names are discarded (because we won't be needing them).

   Modify this code so that it counts and prints the total number of leaf-cells in the expanded hierarchy. For your reference, a simpler example is provided in the file *example.hier*.

Your code should count 8 total leaf-cells in the expanded hierarchy of *example.hier*. Remove the code to print out the hierarchy. Ensure that you update the *Makefile* if needed so that the code builds and executes properly with the *make* and *make sim* commands.

WARNING: Please do not commit files that are not needed to build your design, such as object and executable files. If such files are committed to your repository, a few points will be deducted.

4. (30 points) The file *README.txt* in the *hw03/p4* directory contains a sentence with 10 words. Create a class called *reader* with a constructor that reads a file and stores the individual words using the *string* type and one STL container type. The *reader* class should also contain a method called *reversePrint()* to print out the sentence in reverse order (one word per line). Ensure that your code compiles and executes correctly with the readfile.cpp file provided. Ensure also that you update the *Makefile* if needed so that the code builds and executes properly with the *make* and *make sim* commands. As before, please do not commit files that are not needed to build your design, such as object and executable files.