Computer Design & Organization
Assignment 1
Due: Wednesday, October 4

The purpose of this assignment is to acquaint you with the sim-outorder simulator that is part of the SimpleScalar tool set and the environment in which it executes. Sim-outorder is a sophisticated instruction-level simulator that implements most of the architectural features we will study this quarter. But for this assignment, we will use it as though it looks very much like the R3000 you studied in CSE378. Sim-outorder implements the SimpleScalar instruction set architecture, which is very similar to the MIPS architecture.

For this assignment you are expected to work in teams of 3 people. (By the way most assignments will involve 3-person teams; for each assignment you should work with a different set of partners, so plan ahead for your partners.)

For this assignment, you should:

1. Pick an application from the application directory to instrument. All these programs are taken from the SPEC95 benchmark suite, which was the standard workload for architecture research two suite-generations ago (we are now on SPEC2000). They have already been precompiled for sim-outorder and you can use them as input to the simulator. Deepak or Danny will send you mail if it turns our that any of the applications are not appropriate for this assignment.
2. Set sim-outorder configuration parameters to reflect a computer that has the following configuration:
   - a pipeline that fetches, decodes, issues, executes and commits one instruction/cycle, no matter what the instruction type
   - only one of each type of functional unit
   - an 8KB, two-way set-associative L1 instruction and data caches with 32 byte blocks
   - a 256KB, direct-mapped L2 unified cache with 32 byte blocks
   - an 8-way, 128-entry data TLB
   - a 4-way, 64-entry instruction TLB
   - All caches and the TLB have an LRU block replacement policy.
   - The page size is 4KB.

   This means that the rest of the parameters are either left with their default values or must be set to: fetch:speed <1> and issue:inorder.

   The configurations are set by command line arguments or the config file.
3. Report the values for the following metrics:
   - total number of instructions committed -- this is all the instructions that have completed all the phases of instruction execution, including writing their results to the register file.

- the total number of block replacements for the L1 data cache
- the hit ratio for the L2 cache

4. Answer the following questions:
   - What is the rationale for the value of the number of cycles per instruction?
   - Why is there a difference between the number of blocks replaced and the number of blocks written back in the L1 data cache?
   - Justify the value of the total number of write-backs for the L1 instruction cache.
5. Hand in paper copies of the output generated by sim-outorder and your answers to the questions.

Deepak will hand out a document on using the sim-outorder simulator in section on Thursday.

Computer Design & Organization
Assignment 1
Due: Wednesday, October 4

The purpose of this assignment is to acquaint you with the sim-outorder simulator that is part of the SimpleScalar tool set and the environment in which it executes. Sim-outorder is a sophisticated instruction-level simulator that implements most of the architectural features we will study this quarter. But for this assignment, we will use it as though it looks very much like the R3000 you studied in CSE378. Sim-outorder implements the SimpleScalar instruction set architecture, which is very similar to the MIPS architecture.

For this assignment you are expected to work in teams of 3 people. (By the way most assignments will involve 3-person teams; for each assignment you should work with a different set of partners, so plan ahead for your partners.)

For this assignment, you should:

1. Pick an application from the application directory to instrument. All these programs are taken from the SPEC95 benchmark suite, which was the standard workload for architecture research two suite-generations ago (we are now on SPEC2000). They have already been precompiled for sim-outorder and you can use them as input to the simulator. Deepak or Danny will send you mail if it turns our that any of the applications are not appropriate for this assignment.
2. Set sim-outorder configuration parameters to reflect a computer that has the following configuration:
   - a pipeline that fetches, decodes, issues, executes and commits one instruction/cycle, no matter what the instruction type
   - only one of each type of functional unit
   - an 8KB, two-way set-associative L1 instruction and data caches with 32 byte blocks
   - a 256KB, direct-mapped L2 unified cache with 32 byte blocks
   - an 8-way, 128-entry data TLB
   - a 4-way, 64-entry instruction TLB
   - All caches and the TLB have an LRU block replacement policy.
   - The page size is 4KB.

   This means that the rest of the parameters are either left with their default values or must be set to: fetch:speed <1> and issue:inorder.

   The configurations are set by command line arguments or the config file.
3. Report the values for the following metrics:
   - total number of instructions committed -- this is all the instructions that have completed all the phases of instruction execution, including writing their results to the register file.

- the total number of block replacements for the L1 data cache
- the hit ratio for the L2 cache

4. Answer the following questions:
   - What is the rationale for the value of the number of cycles per instruction?
   - Why is there a difference between the number of blocks replaced and the number of blocks written back in the L1 data cache?
   - Justify the value of the total number of write-backs for the L1 instruction cache.
5. Hand in paper copies of the output generated by sim-outorder and your answers to the questions.

Deepak will hand out a document on using the sim-outorder simulator in section on Thursday.