Project Description

January 26, 2016

Important Information:

This is an **INDIVIDUAL** Project. All of the work you present must be your own. The following actions are violations of academic integrity rules:

- Copying solutions from any source including the Internet, solutions manuals, or other students.
- Copying the project from any source including other students.

The penalty for cheating includes:

1. An automatic zero on the assignment
2. Reduction of your final grade by one letter grade;
3. Notification of the incident to the UCF Office of Student Conduct.

More information about UCF rules and regulations at http://goldenrule.sdes.ucf.edu/

Project Objective:

Learn and explore the capabilities of the SimpleScalar simulator for MIPS architecture.

Project Description:

The SimpleScalar simulator lets you experiment with different aspects of architectural decisions in building a processor. For example, what should the cache structure be, what technique should branch prediction follow, will out of order execution be beneficial, and so on. SimpleScalar simulates each of these functionalities and lets you as a user run benchmarks to determine the usage of these functionalities.

The following are available to simulate and test in SimpleScalar. Let us call them the "Super Seven" for easy reference.

1. sim-bpred — simulates branch prediction
2. sim-cache — simulates cache behavior
3. sim-eio — simulates external I/O trace generator
4. sim-fast — sample fast functional simulator implementation
5. sim-outorder — simulates out of order execution
6. sim-profile — sample functional simulator implementation with profiling
7. sim-safe — sample functional simulator implementation

You will be assigned one benchmark to test the Super Seven.

Your task for this project is comprised of two sub-tasks.

1. Your first sub-task is to run your benchmark and obtain output on each of the Super Seven.
2. Your second sub-task is to perform analyses on the output and determine the nature of the benchmark provided to you. For example, if you notice a lot of memory accesses on running your benchmark on sim-cache, you may conclude that your benchmark is memory-intensive.

Project Deliverables:

You are required to submit two progress reports and one final report for successfully completing your project. The first report is related to your first sub-task and the second report is related to your second sub-task. The final report combines highlights of the first and the second progress reports to characterize the benchmark.

The **first report** should at a minimum contain the following:
1. Your name.
2. Identification of the command to run the benchmark.
3. Parameters used for invoking each of the Super Seven. Identify and write down the description of each of the commands for testing the Super Seven. For example, in sim-cache, if you want to test a 4-way associative cache, how would you specify that while invoking sim-cache. Note that this information is usually in the read-me files.
4. At least four out of seven outputs obtained for each of the Super Seven.

The **second report** should at a minimum contain the following:
1. Your name.
2. All seven outputs for each of the Super Seven.
3. One section for each of the outputs describing the trends seen for the benchmark.

The **final report** should at a minimum contain
1. Your name.
2. A discussion section that highlights the main findings from the first and the second report.
3. Characterization of the benchmark based on observed output behavior on running the Super Seven.

Due dates:

1. First report 10 am Feb 16, 2016
2. Second report 10 am Mar 1, 2016
3. Final report 10 am Mar 22, 2016

Report weights:

1. First report 25%
2. Second report 25%
3. Final report 50%

Accepted report document type: Only PDF submissions will be accepted.

Task Guidelines:

You have the choice to either install the SimpleScalar simulator on your personal computer or use the Eustis server. Here are the details for installing SimpleScalar and obtaining your benchmark.

Step 1. Preparing your personal computer for SimpleScalar simulator

Requirement: Linux-based operating system

If you do not have a Linux-based operating system on your personal computer, you may use Virtual Box to install one. Do let the TA know if you have any installation issues.

Next, follow steps described under SimpleScalar Installation below.

Step 2. Obtaining access to Eustis

1. Connectivity:
    1. If you are on campus, you have to be connected to the UCF_WPA2 or have a wired connection.
    2. If you are at home, you have to be on UCF's VPN. If you have any issues contact help@eecs.ucf.edu

2. You will need to ssh into Eustis from your personal computer.
    1. Non-Linux/Unix machines: You will need PuTTY or something similar.
    2. Linux/Unix machine: Use your machine's terminal.

    Type in

        ssh yourNID@eustis.eecs.ucf.edu

    Your default password is Pyymmdd

Step 3. SimpleScalar Installation

Once in your Eustis account or on your personal machine running Linux OS, do the following:

1. Create a directory for SimpleScalar

mkdir SimpleScalar

2. Download SimpleScalar
   cd SimpleScalar
   w3m http://www.simplescalar.com/agreement.php3?simplesim-3v0e.tgz

   Use the arrow keys to navigate to the bottom of the page to accept the terms. Move the cursor to 'I Agree' and hit enter.

You will see a message '(Download) Save file to: simplesim-3v0e.tgz' at the bottom of the page, hit enter to save file in the current directory. To get out of the browser, press q.

3. Compile SimpleScalar
   tar xvzf simplesim-3v0e.tgz
   cd simplesim-3.0
   make config-pisa
   make

4. Test the installation using
   ./sim-safe tests/bin.little/test-math

Step 4. Obtaining your benchmark

An email will be sent to you indicating your benchmark by Thursday, Jan 28, 2016. The email address on file will be used.

Step 5. Downloading your benchmark

Benchmarks are located on Eustis in /usr/local/class/benchmarks. Use the cp command to copy the zipped (.tar.gz) benchmark file to your simplesim-3.0 directory either in your Eustis account or your personal computer.