Welcome to the Lab

CMPUT 229

University of Alberta

Fall 2012

Outline

The Lab

- The lab sessions are mostly time for you to work. Attendance is not mandatory.

The Lab

- The lab sessions are mostly time for you to work. Attendance is not mandatory.
- Your TAs are available during the lab sessions. Ask us questions if you have them!

The Lab

- The lab sessions are mostly time for you to work. Attendance is not mandatory.
- Your TAs are available during the lab sessions. Ask us questions if you have them!
- There will sometimes be presentations (like this). We will let you know in advance when these are.

The Lab

- The lab sessions are mostly time for you to work. Attendance is not mandatory.
- Your TAs are available during the lab sessions. Ask us questions if you have them!
- There will sometimes be presentations (like this). We will let you know in advance when these are.
- You can attend any lab section, but registered students always have priority.

What Will We Be Doing?

- In short: assembly language programming for the MIPS architecture.
- But writing assembly for real hardware is hard:
  - You have to interface with the operating system, or write lots of low-level code to manage the hardware yourself.
  - Testing and debugging requires special equipment and is difficult.
- So we use a simulator: SPIM.

What is SPIM?

- SPIM is MIPS backwards!
- It's a MIPS simulator.
- It gives you an easy environment in which to run and debug your MIPS code.
- XSPIM is a graphical interface to SPIM.
- There's more information in appendix B of the textbook.

What is SPIM?

- SPIM is MIPS backwards!
- It's a MIPS simulator.
- It gives you an easy environment in which to run and debug your MIPS code.
- XSPIM is a graphical interface to SPIM.
- There's more information in appendix B of the textbook.

Some things to remember:

- SPIM uses the endianness of the host machine.
- SPIM is not a cycle-accurate simulator: it doesn't accurately simulate cache, memory latency, floating-point operations, etc. You can't tell how efficient your MIPS code is by running it in SPIM.

Pseudo-Instructions

- SPIM implements some pseudo-instructions that expand into real instructions. When you run code, you might see instructions you didn't write. For example:

What you wrote	What you see in SPIM
`li $t0, 10`	`ori $8, $0, 10`
`move $v0, $t0`	`addu $2, $0, $8`

- Pseudo-instructions will do what you intend, so they're nothing to worry about. Just be aware that they may expand to multiple instructions, and don't be surprised when you see them.

Basic XSPIM Usage

- Run `xspim` from a terminal.
- Press the “load” button, type in a filename, and click “assembly file” to load a file.
- Press the “run” button and then click “ok” to run your program.
- If your program needs input or prints output, XSPIM will pop up a terminal. **Closing this terminal will kill XSPIM.**
- Press the “quit” button to quit.

Basic SPIM Usage

- Start `spim` in a terminal.
- Type `load ‘‘file.s’’` to load your program.
- Type `run` to run your program.
- If your program needs input or prints output, it will appear in the terminal.
- Type `exit` to quit.
- You can always type `?` to get help!

Assignment Tips for CMPUT 229

- Read specifications very carefully. Pay special attention to what you are to include - sometimes we don't want a `main` method.
- Even if you don't finish an assignment, make sure you submit something that loads and runs.
- Test your assignments on the lab machines before you submit. That's where we'll be marking them.
- For most assignments, the marksheet will be posted in advance. Look at it to get an idea of what's important.
- Style marks are easy marks. Format your code like the `example.s` file we provide, and write good comments.
- With moodle (eClass) you can submit as many times as you like. Submit early and often (e.g. every time you get part of the assignment working). We don't accept any late submissions!

Questions?