Institut for Matematik og Datalogi
Syddansk Universitet

September 4, 2011
JFB

Introduction to Computer Science
E11 – Lecture 3

Lecture, September 1, 12:15–14, U26

There was be a brief introduction to LaTeX. We also covered up to and including page 64 in chapter 1 in the textbook.

Lecture, September 6, 8:15–10, U26

We will finish with chapter 1.

The textbook's interpretation of the mantissa in floating-point representations is not the same as the IEEE-standard and hence somewhat outdated: The book says that the mantissa 1010 means 0.1010 and that the first bit is always 1 in normalized numbers. IEEE-standard says that 1010 means 1.1010, meaning that the fixed normalization bit is a "hidden bit" or "implicit bit" before the radix point. In calculating the value represented by the mantissa, an extra 1 is added. This way the first bit in the mantissa may be 0. Notes about the IEEE standard can be found at http://steve.hollasch.net/cgindex/coding/ieeefloat.html. (For problems in this course, we will use the format described in the textbook, using the same number of bits, but the mantissa will have this IEEE-standard form, with the implicit bit.)

Lecture, September 8, 12:15–14, U26

We will cover chapter 2 of the textbook and begin on chapter 3.

Study group for first-year students
September 8, 10:15-12

- Consider the problems for the discussion section on September 14, along with the problems on pages 49 and 55 of the textbook for September 13. For these problems use the floating-point format discussed in class, which is the same as in the textbook except that it uses an implicit bit in the mantissa.

  Be prepared for someone in your study group to present problems in discussion section.

  Group 1 should prepare: Problem 4 on page 49, problems 1c amd 4c on pages 60–61, problem 2b on page 71, problems 1 and 6 on pages 80–81, problem 39 on page 84.

  Group 2 should prepare: Problems 1 on page 55, problems 2c and 6 on pages 60–61, problem 1c on page 71, problem 2 on pages 80–81, problem 51a on page 85, and problem 1 from the discussion section.

  Group 3 should prepare: Problems 2 on page 55, problem 3b on pages 60–61, problems 1b and 2d on page 71, problem 5 on pages 80–81, problem 54 on page 85, and problem 2 from the discussion section.

- Prepare a question about something from one of the first two lectures to ask in discussion section (or lecture).

Laboratory: September 13, IMADA's terminal room

Meet in IMADA's terminal room with your login information. Discussion in groups (only two, or possibly three, people per group, since you will sit at a computer for the first part): Two simulators are available from the course's homepage
http://www.imada.sdu.dk/∼joan/intro/index.html.
(there is some variation in quality and ease of use).

Download a simulator in Java from the homepage. Create a directory for this course and save the file as `Simulator.jar` or `BrookshearMachine.jar`. Run the simulators using `java -jar Simulator.jar` or `java -jar Brookshear.jar`. Clicking on `Help` will tell you how to set the contents of some memory cells.

1. Do problem 1 on pages 102–103 of the textbook. To input the data and program, type `[00] 14 02 34 17 C0 00` in the white field at the top of the window. Click on `Load Data`.

   In the BrookshearMachine you can just enter the data directly into the fields. Note that when in `Memory List view` you need to specify two bytes at a time, i.e., the first memory cell denoted 00 gets 14 02, the second memory cell denoted 02 gets 34 17, etc.

   Use Appendix C, starting on page 565 of the textbook, to figure out what should happen. Then, `Single Step` through the execution to see that it does happen.

2. Do problem 2 on page 103. To load the value `B0` into the program counter, type `[PC] B0` in the `Data Input Window` and click on `Load Data`. For the BrookshearMachine enter B0B0 at address 00 to jump to B0. Why does register 3 get the values it does when you step through the program?

3. Do problem 3 on page 103. Note that the operation `B` is usually referred to as a *conditional* branch, and there is usually also an *unconditional* branch instruction, which always causes the program counter to get the specified value (without checking the values of any registers). How is the conditional branch instruction used here to get the effect of an unconditional branch?

4. Do problem 4 on page 104. This is strange in that it is an example of how a program can modify itself when there is no distinction between program and data. Discuss the security implications of this.

Discuss the following problems from the textbook in groups of three or four:

Page 49: Problem 4.

Page 55: Problems 1, 2 (see the appendix on page 561).

Discussion section: September 14

Pages 60–61: Problems 1c, 2c, 3b, 4c, 6.

Page 71: Problems 1b, 1c, 2b, 2d (for these problems use the floating-point format discussed in class, which is the same as in the textbook except that it uses an implicit bit in the mantissa).

Page 80–81: Problems 1, 2, 5, 6.

Page 84: Problem 39 (again use the format discussed in class).

Page 85: Problems 51a, 54.

Pages 85–86: Problems 1, 5, 6.

1. Choose two floating point numbers and add them together. If you cannot express the result in the same format, try two other numbers.

2. Choose a number which cannot be expressed exactly in the floating point format we use, but could be expressed exactly if there were more bits. How many more bits do you need?