Institut for Matematik og Datalogi
Syddansk Universitet

August 19, 2008
JFB

Introduction to Computer Science
E08 – Week 1

Textbook

Computer Science: An Overview, 10th Edition, by J. Glenn Brookshear, 2009.
The textbook will be supplemented with notes.

Format

Lectures will usually (after the first) be on Mondays and (some) Wednesdays in U9. The Monday lectures will be 12:15–14 and the Wednesday lectures will be 10:15–12. There will be two discussion sections, S71 and S72, each of which will meet once or twice every week (after the first). The "instruktor" for S71 will be Arun Vadiveal and the "instruktor" for S72 will be Niels Kjeldsen. Some of the discussion sections will take place as labs in IMADA's terminal room. The first lab will be instead of your first discussion section in the third week, week 37.

The course will be graded on a Pass/Fail basis, and satisfactory completion of all 10 assignments is required to pass. "Satisfactory completion" means that the answers are correct, with only very minor errors, and that they have been turned in on time. (Note that if you send them to me by e-mail, that you are responsible for them actually getting to me on time. Sending from IMADA's computers will help ensure that your e-mail is not delayed at an intermediate computer for many hours.) You are allowed to retry (at most once) on at most 3 assignments which were not approved. These 10 assignments count as the exam in the course, so cheating on these assignments is viewed as cheating on an exam. You are allowed to talk about course material with your fellow students, but working together on assignments is cheating. If you have questions about the assignment, come to me or your "instruktor".

The weekly notes and other information about the course are available through the Worldwide Web. Use the URL:

http://www.imada.sdu.dk/∼joan/intro/

You are responsible for finding all weekly notes there (or in Blackboard) yourself. Please read the appropriate sections in the textbook or notes before coming to class and bring your textbook with you. Preparing for discussion sections (and labs) is important.

I have office hours on Mondays and Thursdays from 9:00 to 9:45 (starting September 1).

Lecture, August 26

Rolf Fagerberg will lecture. He will begin with an introduction to the course, covering chapter 0 in the textbook, but skipping section 0.2. He will also begin on chapter 1.

Lecture, August 27

Kim Skak Larsen will lecture. He will cover more of chapter 1 in the textbook.

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.)

Lectures, September 1 and 3

We will finish chapters 1 and 2 and begin chapter 3.

Discussion section: week 36

Discussion in groups:

1. Divide into groups of three people (or four). One person will choose five cards to give to the first "performer", the first performer will give four of them to the second "performer", one at a time, and the second performer will announce what the fifth card is. Each person should practice each "performer" part at least three times.

2. Discuss (in your groups) various methods for improving the magic trick. For example, if the second card determines the suit, after seeing the trick repeated several times, the audience might find it easier to guess how it is done. Define an algorithm for a modification of the trick which makes this harder to see.

3. Discuss how to do a magic trick, where one performer is thinking of a number between 1 and 24, tells the audience the number, gets some cards from the audience, and passes some cards to the second performer, who announces the number.

4. Discuss how to extend this to larger numbers than 24. When might it be better to only use the color on the card (whether it is red or black, but also consider using the four different suits), rather than some permutation (ordering) of the cards?

5. Find a "bad" pair of integers for the greatest common divisor algorithm, where a pair is bad if the algorithms must perform a lot of steps relative to how large the numbers are. (One expects more steps for larger numbers.)

6. Do problem 3 on page 91 of the textbook.

7. Design a circuit, using only AND, OR and NOT gates which takes three bits as input and outputs a 1 if the input has at least two zeros, and a 0 otherwise.

8. Design a circuit containing only AND and XOR gates (with at most two inputs) which takes three bits as input and outputs a 1 if the input has at least two ones, and a 0 otherwise. (In the student resources

for the course textbook, under the Activities for Chapter 1, there is a simulator for logic circuits which you could use to check your circuit. It is time consuming to use, though.) As an extra challenge, try to do it so that there is only one AND gate, though more XOR gates. (Minimizing the number of AND gates can be useful in some cryptographic applications.)

9. Discuss questions 2 and 4 on pages 34–35 of the textbook.

Assignment due 12:15, September 8

Late assignments will not be accepted. Working together is not allowed. (You may write this either in English or Danish, but write clearly if you do it by hand.) Next to your name, write which sections you are in, S71 or S72.

1. Write down the algorithm for the "magic trick" described in class, where one performer passes four of five playing cards to the other, so that the other can tell what the fifth card is. Write this as an algorithm, so that all steps are clearly specified, without ambiguity.

2. Either do the first two problems below or the third one. The third one is somewhat more challenging. As with all problems in this course, explain your results.

   (a) Design and draw a circuit containing only AND, OR and NOT gates (with at most two inputs) which takes three bits as input and outputs a 1 if the input is 011, 111 or 110, and a 0 otherwise. (In the student resources for the course textbook, under the Activities for Chapter 1, there is a simulator for logic circuits which you could use to check your circuit. It is time consuming to use, though.)

   (b) Design and draw a circuit containing only AND, OR and NOT gates (with at most two inputs) which takes four bits as input and outputs a 1 if the input is 0101, 0111 or 1110, and a 0 otherwise.

   (c) Design and draw a circuit containing only AND and XOR gates (with at most two inputs) which takes six bits as input and outputs a 1 if the input has at least four ones, and a 0 otherwise. Use only four AND gates. How many XOR gates do you need?