Institut for Matematik og Datalogi
Syddansk Universitet

September 27, 2010
JFB

Introduction to Computer Science
E10 – Lecture 7

Lecture, September 20, 8:15–10, U37

We covered up through section 4.4 of chapter 4 (section 4.5 will be covered later in the course). We also began on chapter 5, covering up throughsection 5.1 and the subsection "The Art of Problem Solving" from section 5.3.

Lecture, September 23, 14:15–16, U71

Kim Skak Larsen will lecture on databases from chapter 9 in the textbook.

Lecture, September 27, 8:15–10, U37

We will cover chapter 5 in the textbook.

Discussion section: October 4, 14:15–16, U26A

Discuss the following in groups of three or four (the page numbers and problem numbers are from the textbook - though depending on your textbook, you may have to subtract 18 from the page numbers):

1. Page 239–240: Problems 1, 2, 3.
2. Page 250–251: Problem 5, 6, 7.
3. Page 260: Problem 2, 3, 4.
4. Pages 270: Problem 3.
5. Pages 271–276: Problems 50, 53 (what precondition and loop invariant should hold)?

6. Consider the following problem (mentioned in lecture): There are three politicians, A, B, and C. You know that one of them always tells the truth, one of them always lies, and one of them sometimes tells the truth and sometimes lies. You are allowed to ask these three politicians any three true/false questions you like, and you may choose which politician is asked which question. How would you determine how to order the politicians by how often they tell the truth? This problem is quite difficult. Try your problem solving abilities, but do not be disappointed if you fail.
7. Page 277 questions 1, 2, 3, 4, 6.

Assignment due 12:15, October 11

Late assignments will not be accepted. Working together is not allowed. (You may write this either in English or Danish.) Submit a single PDF file through the Blackboard system and include your name on the first page. Remember to explain your answers.

Do the following problems in the textbook:

1. Design an algorithm for finding the cubic root of a positive integer, rounded down to the nearest integer. Thus, given input N, a positive integer, you should find a positive integer m such $m^3 \leq N$, but $(m+1)^3 > N$. Use the binary search idea.
   (a) Express the algorithm in pseudocode.
   (b) Find a fundamental operation and use big theta notation to express how long your algorithm takes. Express this as a function of the positive integer N which is input, and also as a function of the length of N (the number of bits in N). State any assumption you make about how long it takes to multiply two integers, one of length s and one of length t.
2. Page 272: Problem 17. Express the algorithm in pseudocode. Your algorithm should be general enough to solve general problems like this, with arbitrary numbers of bits in the numbers (though the sum will always have one more bit than the other two numbers, which you may assume have the same number of bits). Your algorithm should also report correctly when there is no solution.