Institut for Matematik og Datalogi
Syddansk Universitet

November 19, 2010
JFB

Introduction to Computer Science
E10 – Lecture 11

Lecture, October 7, 14:15–16, U71

Rolf Fagerberg lectured on section 9.5 in the textbook (merging and hashing).

Lecture, October 11, 8:15–10, U37

We will cover the subsection on "Software Verification" in section 5.6. Then, we will begin on chapter 12 in the textbook.

Lecture, November 8, 8:15–10, U37

We will continue with chapter 12 in the textbook, concentrating on security (also from sections 3.5 and 4.5) and cryptography.

Discussion section: November 10, 12:15–14, U28

Do problem 15 on page 608 before coming to discussion section, so that you can compare solutions. Discuss the following problems in groups of three or four.

1. Questions 2, 3, and 5 on page 580.
2. Compare your solutions to problem 15 from page 608.
3. Question 3 on page 587.
4. Do problems 2, 4b, 6, 9, 12, and 14 on pages 607–608.
5. Discuss the social issues 1 and 5 on pages 611–612.

Assignment due 8:15, November 24

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. If you submit this assignment more than once, use the same identification number both times. Remember to explain your answers.

1. Assume sets of numbers are represented by arrays sorted on element value. For example, the set $\{14, 27, 13, 9, 32\}$ is represented by an array of length 5 containing $[9, 13, 14, 27, 32]$. Write a program in Java or Maple for constructing $(A \cap B) \cup C$. Use an algorithm similar to that in Figure 9.15 (which goes through each list only once). As in problem 5 (from the discussion section described on the note for Lecture 10), process the three arrays simultaneously (you should not first calculate $A \cap B$ and then union with C). Test your algorithm. Turn in both the program code (commented) and your test results.

2. If a hash file is partitioned into 11 buckets, what is the probability of at least two of three arbitrary records hashing to the same sections? Assume that the hash function is such that a randomly chosen record is equally likely to hash to any of the sections. How many records must be stored in the file until it is more likely for collisions to occur than not? Assume again that there are 11 bins.