Institut for Matematik og Datalogi
Syddansk Universitet

February 21, 2002
JFB

Discrete Mathematics with Applications
F02 – Lecture 4

Lecture, February 18

We finished sections 3.1 and 3.2 of chapter 3, along with the subsection of section 2.3 having to do with division.

Lecture, February 25

The section of the DM11 notes on structural induction will be covered, along with section 3.3 of chapter 3. We will begin on sections 2.3 and 2.4 of chapter 2.

Lecture, March 4

We will finish section 2.4, along with Lamé's Theorem from chapter 3. We will also cover section 2.5 of chapter 2.

Problems to be discussed on March 5

- Use structural induction to show that if a well-formed formula of variables, numerals, and operators from $\{+, -, *, /, \uparrow\}$ has n operators, the sum of the number of variables and numerals is $n + 1$.
- Show that concatenation is an associative operation, i.e. $(x \odot y) \odot z = x \odot (y \odot z)$. Use the definitions given in the course notes.
- Give a recursive definition of binary trees with height h, where the height of the tree is defined to be the number of nodes on the longest path from the root to a leaf. Use structural induction to show that a binary tree of height h has at most $2^h - 1$ nodes.

- Lists can be defined recursively exactly as strings. In Maple a list is written as items, separated by commas, and surrounded by square brackets, such as: `A:=[1,2,3,4,5,6];` Write your own Maple function to compute the sum of the squares of the values in a list (for example, those in the list A above). If your list L has n values, then you should compute $\sum_{i=1}^{n}(L[i])^2$. Do it without using the Maple function **sum**. It is possible in Maple to define a function recursively. Thus, you could define two parameters for your function, the list, plus the length of the list. When the length is zero, your sum should be zero. Can you define your function for a list of length n in terms of a result of applying your function to a list of length $n-1$? You will need the **if** statement in Maple, which you can read about from **Help**.

- 18, 24, 28, (30), (32), 36, and 38 from section 3.3.
  You can also do 18 in Maple, but assume that all lists have length at least 1.
  Note that you should use structural induction for 24 and 28.
  Note that a *palindrome* is a string which is the same as its reverse.

- (5), (6), 13, 21, 22, 23, and 35 from section 2.3.