CSC236 *Intro. to* the Theory of Computation

Lecture 4: Recurrences

Amir H. Chinaei, Fall 2016

Office Hours: W 4-4 BA4222

ahchinaei@cs.toronto.edu
http://www.cs.toronto.edu/~ahchinaei/

Course page:
http://www.cdf.toronto.edu/~csc236h/fall/index.html

Section page:
http://www.cdf.toronto.edu/~csc236h/fall/amir_lectures.html

review

- So far
  - Simple Induction, Strong Induction, WOP, and Structural Induction
- over 48 examples

- This week
  - Recurrence relations, closed forms, and
  - proof of their properties

Example 50: rabbits

- A rabbit couple lives in an island. They are newborn and do not breed until they are 2 months old. Since age 2-month, each couple produces another couple per month. Find a recurrence relation for the number of couples after *n* months, assuming they never die.

Example 50:

Example 51: “00” free strings

- Find a recurrence relation for the number of binary strings of length *n* that do not contain substring “00”. (Revisit of Example 27.)

Example 52: finding closed form

- Assume $f_n = c_1 f_{n-1} + c_2 f_{n-2}$
  - Find roots of $r^2 - c_1 r - c_2 = 0$, r_1 and r_2.
  - Then solve $f_n = \alpha_1 r_1^n + \alpha_2 r_2^n$, using the initial values of f.

Example 52:

Example 53:

$$f(n) = \begin{cases} 2 & n = 0 \\ 7 & n = 1 \\ 2f(n-2) + f(n-1) & n > 1 \end{cases}$$

Find the closed form of *f*.

Example 53:

Example 54:

Prove $f(n) < 2^{n+2}$. $f(n)$ defined in Examples 53.

Example 54:

notes: