Sveučilište u Dubrovniku
Prof. dr. sc. Hrvoje Kraljević
Dr. sc. Daniel Kressner

2005/2006

Logika, skupovi i diskretna matematika

Exercise 7

The assignments are due to 23.01.2006.

Tutorial 7.1

1. The population of Utopia increases 5% per year. In 2000 the population was 10000. What was the population in 1970?
2. Solve the recurrence relation $S_n = 6S_{n-1} - 8S_{n-2}$ with $S_1 = 1$ and $S_2 = 0$.
3. Solve the recurrence relation $S_n = 7S_{n-1} - 10S_{n-2}$ with $S_1 = 5$ and $S_2 = 16$.
4. Consider the sequence $S_1, S_2, \ldots$ where S_n denotes the number of n-bit strings that do not contain the pattern 00.
   (a) Find a recurrence relation and initial conditions for the sequence $\{S_n\}$.
   (b) Show that $S_n = f_{n+2}$, $n = 1, 2, \ldots$, where f denotes the Fibonacci sequence.
   (c) By considering the number of n-bit strings with exactly i 0's, show that

$$f_{n+2} = \sum_{i=0}^{\lfloor (n+1)/2 \rfloor} \binom{n+1-i}{i}, \quad n = 1, 2, \ldots.$$

Tutorial 7.2

1. Construct sequences and patterns for which the number of comparisons needed by Algorithm 7.4 in the lecture notes grows proportionally with $(n - m + 1)m$.
2. How many iterations does the Euclidean algorithm require for two consecutive Fibonacci numbers? It can be proven that this is the worst-case input for the Euclidean algorithm. Derive an asymptotic upper bound for the time needed by the Euclidean algorithm.
3. Show that $n! = \Theta(n^n)$ and $1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} = \Theta(\log n)$.

Homework assignment 7.1 **7 points**

1. Assume that a person invests \$3000 at 12 percent annual interest compounded *quarterly*. How long will it take to double the initial investment? How long will it take to double the investment if the interest is compounded *yearly*?

2. Solve the recurrence relation $S_n = 2S_{n-1} + 8S_{n-2}$ with $S_1 = 4$ and $S_2 = 10$.

3. Solve the recurrence relation $2S_n = 7S_{n-1} - 3S_{n-2}$ with $S_1 = 1$ and $S_2 = 1$.

4. Consider the sequence $S_1, S_2, \ldots$ where S_n denotes the number of n-bit strings that do not contain the pattern 010.

   (a) Compute S_1, S_2, S_3, and S_4.

   (b) By considering the number of n-bit strings that do not contain the pattern 010 *and*

   i. have no leading 0's (i.e., begin with 1);
   ii. have one leading 0 (i.e., begin with 01);
   iii. have two leading 0's; and so on,

   derive the recurrence relation

   $$S_n = S_{n-1} + S_{n-3} + S_{n-4} + S_{n-5} + \cdots + S_1 + 3. \tag{1}$$

   (c) By replacing n by $n-1$ in (1), write a formula for S_{n-1}. Subtract the formula for S_{n-1} from the formula for S_n and use the result to derive a short recurrence relation.

Homework assignment 7.2 **6 points**

1. Write an algorithm that returns the largest and the second-largest values in a sequence $s_1, \ldots, s_n$. Assume that $n > 1$ and that the values in the sequence are distinct.

2. Find a function $g(n)$ of the form A^B such that $\binom{n}{3} = O(g(n))$.

3. Consider the following algorithm:

```
Input:   m, u (both integers)
Output:  power

q = m
bot = 1
top = u
while q > 0 {
   if (q mod 2 == 1)
      bot = top*bot
   top = top*top
   q = q / 2         // integer division
}
power = bot
```

   (a) What does this algorithm compute?

   (b) What is the complexity of the algorithm with respect to the number of decimal digits of m?

Homework assignment 7.3 **7 points**

Get familiar with trees and sorting by reading the material handed out in the lectures (pages 193–199). Solve Exercise 16.2, parts 1–4, on page 199.