THE UNIVERSITY OF AUCKLAND
Department of Computer Science
Private Bag 92019,
Auckland, New Zealand

415.320 Algorithmics

Assignment No 2

August 2000

Due date: 11 September 2000, time 4.00 pm, 320 box in the Computer Science Department main office.

1. **[15 points]** Prove the relation

$$\sum_{i=1}^{n} i^k \log_2 i \in O(n^{k+1} \log_2 n).$$

Hint: you may use the inequality $\sum_{i=1}^{n} f(i) \leq \int_{x=1}^{x=n+1} f(x)\, dx$.

2. **[5 points]** Let $c > 0$ and consider the recurrence relation

$$t(n) = \begin{cases} c, & \text{for } n = 1, \\ t(n/2) + 1, & \text{for } n > 1, \end{cases}$$

when n is a power of 2. Prove that $t(n) \in \Theta(\log_2 n)$.

3. **[10 points]** Solve the following recurrence relation:

$$t_n = \begin{cases} n, & \text{for } n = 0, 1, \\ 5t_{n-1} - 6t_{n-2}, & \text{for } n > 1. \end{cases}$$

4. **[20 points]** Find an explicit solution for the following recurrence relation:

$$T(n) = \begin{cases} a, & \text{for } n = 0, 1, \\ T(n-1) + T(n-2) + c, & \text{for } n > 1, \end{cases}$$

where a, c are real constants. Let f_n denote the nth Fibonacci number. For which values for a, c, the relation $f_{n+1} = T(n)$ is true for all n?

5. **[30 points]** Consider the following recurrence relation:

$$T(n) = \begin{cases} T(n/2) + 1, & \text{for even } n, \\ 2T((n-1)/2), & \text{for odd } n. \end{cases}$$

- Prove that the solution of the recurrence relation for powers of 2 is $T(n) \in \Theta(\log_2 n)$.
- Show that for an infinity of integers n, $T(n) \in \Omega(n)$.
- Comment the adequacy of solving recurrence relations only for powers of 2.

6. **[15 points]** Solve the recurrence relation:

$$T(n) = \begin{cases} 1, & \text{for } n = 2, \\ 2T(\sqrt{n}) + \log_2 n, & \text{for } n > 2, \end{cases}$$

for powers of powers of 2.

7. [**5 points**] Is the solution of the recurrence relation:

$$T(n+1) = \begin{cases} -1, & \text{for } n = 0, \\ 2T(n), & \text{for } n > 0, \end{cases}$$

the time complexity of some algorithm?