

# Data Structures and Algorithms COMP-251 A

## Problem Assignment 3

### 1. Algorithms for the Fibonacci Function

The following matrix relationship between a matrix power and the Fibonacci sequence leads to an  $O(\log n)$  time algorithm for computing  $f(n)$ .

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^n = \begin{bmatrix} f(n+1) & f(n) \\ f(n) & f(n-1) \end{bmatrix}$$

Prove by *induction* that this relation holds for all  $n$  greater than one.

### 2. Binary Search

The recurrence relations for the total number of *comparisons* done in binary search in the worst case are given by  $T(n) = 1$  for  $n=1$ , and for all  $n>1$  by:

$$T(n) = 1 + T\left(\left\lfloor \frac{n}{2} \right\rfloor\right)$$

Prove by *induction* on  $n$  that the **exact formula** for the solution to these recurrence equations is given by:

$$T(n) = 1 + \lfloor \log n \rfloor$$

### 3. The Skyline Problem in Computer Graphics

Problem 5.11 in the Udi Manber text.

### 4. Analysis of Heap Sorting

(a) Prove that *bottom-up* heap construction with  $n$  keys takes  $O(n)$  time.

(b) Show that

$$\sum_{i=1}^n \log i = \Omega(n \log n)$$