Tutorial 1
Algorithms and Data Structures 1 (1DL210)
2023

Sarbojit Das, Stephan Spengler

1 Induction

1. Using induction, show that for every $n \geq 12$ it holds:

$$\sum_{i=0}^{n}(i-3) \geq \frac{n^2}{4} \tag{1}$$

2. Using induction, prove the following formula for every $n \in \mathbb{N}$ and $a \in \mathbb{R} \setminus \{0,1\}$:

$$\sum_{i=0}^{n} a^i = \frac{1-a^{n+1}}{1-a} \tag{2}$$

Remark. The series $\sum_{i=0}^{\infty} a^i$ is called the *geometric series* of base a.

3. Consider the following recursively defined function:

$$T(n) = \begin{cases} 0 & \text{if } n = 0 \\ 2^n + T(n-1) & \text{if } n > 0 \end{cases} \tag{3}$$

Using induction, show that there are two constants c_1 and c_2 and a natural number n_0 such that $c_1 \cdot 2^n \leq T(n) \leq c_2 \cdot 2^n$ for every $n \geq n_0$.

Remark. We say that $T(n)$ is *tightly bounded* by 2^n.

Bonus question. Find a closed-form expression for $T(n)$ and prove it using induction.

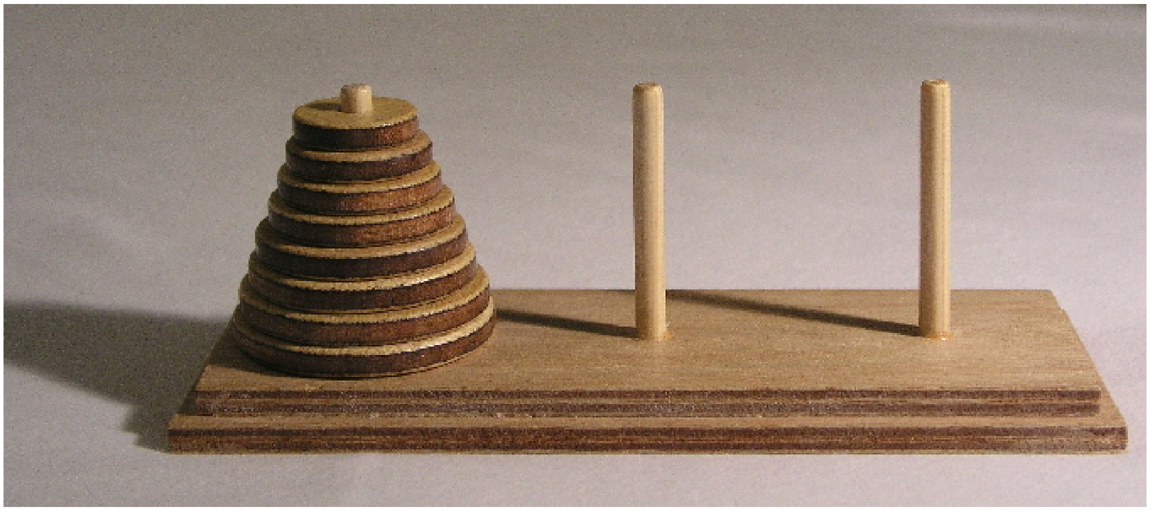

Figure 1: The game *Tower of Hanoi*. *Source:* Wikimedia Commons
(`https://commons.wikimedia.org/wiki/File:Tower_of_Hanoi.jpeg`)

2 Tower of Hanoi

Description. The picture in Figure 1 shows the game *Tower of Hanoi*. The game is composed of three sticks (the *towers*) and $n \in \mathbb{N}$ disks of increasing radius. In the picture, $n = 8$.
At the beginning of the game, all disks are positioned on the first stick, with decreasing radius from bottom to top. The second and third sticks are initially empty. The goal of this game is to move all disks from the first stick to the third one. The rules of the game are:

- Only one disk can be moved at a time.
- Only the topmost disk of a tower can be moved.
- A disk can only be moved onto another tower if the moved disk has a smaller radius than the topmost disk of the tower, or if the target stick is empty.

Questions.

1. Describe a general recursive algorithm to solve the *Tower of Hanoi* game with n disks.

   Hint. Assume that you already know a solution for $n-1$ disks and from that derive a solution for n disks.

2. Find a recursive formula $T(n)$ for the number of moves needed to solve *Tower of Hanoi* with your algorithm.

3. Find a closed-form expression for $T(n)$ and prove it using induction.