

Discrete Mathematics in Computer Science

D1. Introduction to Recurrences

Malte Helmert, Gabriele Röger

University of Basel

Discrete Mathematics in Computer Science

— D1. Introduction to Recurrences

D1.1 Recurrences

D1.2 Examples of Recurrences

D1.3 Fibonacci Series – Mathematical Induction

D1.1 Recurrences

Recursion (1)

The concept of **recursion** is very common in computer science and discrete mathematics.

- ▶ When **designing algorithms**, recursion relates to the idea of solving a problem by solving smaller subproblems of the same kind.
- ▶ **Examples:**
 - ▶ For example, we can sort a sequence by sorting smaller subsequences and then combining the result \rightsquigarrow **mergesort**
 - ▶ We can find an element in a sorted sequence by identifying which half of the sequence the element must be located in, and then searching this half \rightsquigarrow **binary search**
 - ▶ We can insert elements into a search tree by identifying which child of the root node the element must be added to, then recursively inserting it there \rightsquigarrow **trees as data structures**

Recursion (2)

The concept of **recursion** is very common in computer science and discrete mathematics.

- ▶ When **designing data structures**, it is often helpful to think of a data structures as being composed of smaller data structures of the same kind.
- ▶ **Examples:**
 - ▶ A **rooted binary tree** is either a leaf or an inner node with two children, which are themselves rooted binary trees.
 - ▶ A **singly linked list** is either empty or a head element followed by a tail, which is itself a linked list.
 - ▶ A **logical formula** is either an atomic formula or a composite formula, which consists of one of two formulas connected by logical connectives (“and”, “or”, “not”).

Recursion (3)

The concept of **recursion** is very common in computer science and discrete mathematics.

- ▶ In **combinatorial counting problems**, counting things often involves solving smaller counting problems of the same type and combining the results.
- ▶ **Examples:**
 - ▶ counting the number of **subsets of size k** of a set of size n
 - ▶ counting the number of **permutations** of a set of size n
 - ▶ counting the number of **rooted binary trees** with n leaves

Recurrences

In this part of the lecture, we study **recurrences**, i.e., recursively defined functions $f : \mathbb{N}_0 \rightarrow \mathbb{R}$ where $f(n)$ is defined in terms of the values $f(m)$ for $m < n$.

- ▶ Such recurrences naturally arise in all mentioned applications.
- ▶ They are particularly useful for studying the runtime of algorithms, especially recursive algorithms.

Learning Objectives

- ▶ Recurrences are a wide topic, and in our brief coverage we will only scratch the surface.
- ▶ Our aim is to equip you with enough knowledge to
 - ▶ understand **what recurrences are**
 - ▶ understand **where they arise**
 - ▶ understand **why they are of interest**
 - ▶ get to know some **important examples** of recurrences, such as the **Fibonacci series**
 - ▶ get a feeling for some **mathematical techniques** used to solve recurrences, in particular:
 - ▶ **mathematical induction**
 - ▶ **generating functions**
 - ▶ the **master theorem** for divide-and-conquer recurrences
 - ▶ **apply the master theorem in practice**

D1.2 Examples of Recurrences

Examples of Recurrences

In this section, we look at three recurrences that arise in **combinatorics**, i.e., when **counting** things:

- ▶ **factorials**: counting permutations
- ▶ **binomial coefficients**: counting subsets of a certain size
- ▶ **Catalan numbers**: counting rooted binary trees

We also have a first look at the **Fibonacci series**, perhaps the most famous recurrence in mathematics.

Counting Permutations

Let S be a finite set, and let $n = |S|$.

Question: How many permutations of S exist?

We answer this question by answering the following slightly more general question:

Let X and Y be finite sets, and let $n = |X| = |Y|$.

Question: How many bijective functions from X to Y exist?

The permutation question is the special case where $S = X = Y$.

Counting Bijections – Derivation

How many bijective functions from X to Y exist, where $n = |X| = |Y|$?

Denote this number by $f(n)$.

- ▶ We have $f(0) = 1$: there exists one possible function from $X = \emptyset$ to $Y = \emptyset$ (the empty function), and it is bijective.
- ▶ For $n \geq 1$, let $x \in X$ be any element of X .
 - ▶ Every bijection $g : X \rightarrow Y$ maps x to some element $g(x) = y \in Y$.
 - ▶ There are $n = |Y|$ possible choices for y .
- ▶ In order to be bijective, g must bijectively map all **other** elements in X to **other** elements of Y .
 - ▶ Hence, g restricted to $X \setminus \{x\}$ is a bijective function from $X \setminus \{x\}$ to $Y \setminus \{y\}$.
 - ▶ Because $|X \setminus \{x\}| = |Y \setminus \{y\}| = n - 1$, there are $f(n - 1)$ choices for these mappings.
- ▶ This gives us $f(n) = n \cdot f(n - 1)$ for all $n \geq 1$.

Counting Bijections – Result

Theorem

The number of bijections between finite sets of size n , or equivalently the number of permutations of a finite set of size n , is given by the recurrence:

$$f(0) = 1$$

$$f(n) = n \cdot f(n-1) \quad \text{for all } n \geq 1$$

Closed-form solution:

$$f(n) = n!$$

Counting k -Subsets

Let S be a finite set, let $n = |S|$, and let $k \in \{0, \dots, n\}$.

Question: How many subsets of S of size k exist?

Denote this number by $\binom{n}{k}$.

- ▶ We have $\binom{n}{0} = 1$: the only subset of size 0 is \emptyset .
- ▶ We have $\binom{n}{n} = 1$: the only subset of size n is S itself.
- ▶ For all other cases, we count proper, nontrivial subsets.
Let $x \in S$ be any element.
- ▶ There are two kinds of subsets of S of size k :
 - ▶ subsets that do not include x :
Such subsets include k elements of the set $S \setminus \{x\}$.
Because $|S \setminus \{x\}| = n - 1$, there are $\binom{n-1}{k}$ such subsets.
 - ▶ subsets that include x :
Such subsets include $k - 1$ elements of $S \setminus \{x\}$.
Because $|S \setminus \{x\}| = n - 1$, there are $\binom{n-1}{k-1}$ such subsets.
- ▶ In summary: $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$ for all $n \geq 1$ and $0 < k < n$.

Counting k -Subsets – Result

Theorem

Let S be a finite set with n elements, and let $k \in \{0, \dots, n\}$.

Then S has $\binom{n}{k}$ subsets of size k , where

$$\binom{n}{0} = 1$$

$$\binom{n}{n} = 1$$

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1} \quad \text{for all } n \geq 1, 0 < k < n$$

Closed-form solution:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Counting k -Subsets – Proof of Closed-Form Solution

To prove that the given closed-form solution is correct, it suffices to verify that it satisfies the recurrence:

- case $k = 0$: $\frac{n!}{k!(n-k)!} = \frac{n!}{0!(n-0)!} = \frac{n!}{1 \cdot n!} = 1 = \binom{n}{0}$.
- case $k = n$: $\frac{n!}{k!(n-k)!} = \frac{n!}{n!(n-n)!} = \frac{n!}{n! \cdot 0!} = \frac{n!}{n! \cdot 1} = 1 = \binom{n}{n}$.
- case $0 < k < n$:

$$\begin{aligned}
 & \frac{(n-1)!}{k!((n-1)-k)!} + \frac{(n-1)!}{(k-1)!((n-1)-(k-1))!} \\
 &= \frac{(n-1)!}{k!(n-k-1)!} + \frac{(n-1)!}{(k-1)!(n-k)!} \\
 &= \frac{(n-1)! \cdot (n-k)}{k!(n-k-1)! \cdot (n-k)} + \frac{(n-1)! \cdot k}{(k-1)! \cdot k \cdot (n-k)!} \\
 &= \frac{(n-1)! \cdot (n-k)}{k!(n-k)!} + \frac{(n-1)! \cdot k}{k! \cdot (n-k)!} \\
 &= \frac{(n-1)! \cdot ((n-k) + k)}{k!(n-k)!} = \frac{(n-1)! \cdot n}{k!(n-k)!} = \frac{n!}{k!(n-k)!}
 \end{aligned}$$

Binary Trees

Definition (binary tree)

A **binary tree** is inductively defined as a tuple of the following form:

- ▶ The **empty tree** $()$ is a binary tree.
Such a tree is called a **leaf**.
- ▶ If L and R are binary trees, then (L, R) is a binary tree.
Such a tree is called an **inner node**
with **left child** L and **right child** R .

German: Binärbaum

Note: With these kinds of trees, the order of children matters, i.e., (L, R) and (R, L) are different trees (unless $L = R$).

Counting Binary Trees

Question: How many binary trees with $n + 1$ leaves exist?
(Why $n + 1$?)

Denote this number by $C(n)$.

- ▶ We have $C(0) = 1$: $()$ is the only tree with one leaf.
- ▶ For $n \geq 1$, the tree must be an inner node.
Each child must have between 1 and n leaves.
The number of leaves of the children must sum to $n + 1$.
- ▶ Hence, if the left child has $k + 1$ leaves, the right child has $(n + 1) - (k + 1) = n - k = (n - k - 1) + 1$ leaves.
- ▶ We obtain: $C(n) = \sum_{k=0}^{n-1} C(k)C(n - k - 1)$.

Counting Binary Trees – Result

Theorem

There are $C(n)$ binary trees with $n + 1$ leaves, where

$$C(0) = 1$$

$$C(n) = \sum_{k=0}^{n-1} C(k)C(n-k-1) \quad \text{for all } n \geq 1$$

Closed-form solution (without proof):

$$C(n) = \frac{1}{n+1} \binom{2n}{n}$$

Catalan Numbers

The numbers $C(n)$ are called **Catalan numbers** after 19th century Belgian mathematician Eugène Charles Catalan.

First terms of the Catalan sequence:

1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786, 208012, ...

Fibonacci Series

- ▶ The last recurrence we consider in this section is the famous **Fibonacci series** (or Fibonacci sequence).
- ▶ We directly introduce it with its definition as a recurrence rather than via an application.

Fibonacci Series – Definition

Definition (Fibonacci series)

The **Fibonacci series** F is defined as follows:

$$F(0) = 0$$

$$F(1) = 1$$

$$F(n) = F(n - 1) + F(n - 2) \quad \text{for all } n \geq 2$$

German: Fibonacci-Folge

First terms of the Fibonacci series:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, ...

Closed-form solution: \rightsquigarrow next section

Fibonacci Series – Trivia

- ▶ The Fibonacci series is named after Leonardo of Pisa a.k.a. **Fibonacci** (son of Bonacci), who introduced it to Western Europe in the 13th century.
- ▶ It has been known outside Europe much earlier, dating back to the Indian mathematician Pingala (3rd century BCE).
- ▶ The series has many, many applications.
- ▶ There exist mathematical journals solely dedicated to it, the most famous one being “Fibonacci Quarterly”.

D1.3 Fibonacci Series – Mathematical Induction

Overview

- ▶ In this section, we prove a closed-form expression for the Fibonacci series.
- ▶ We do this because the result itself is interesting (because of the many applications of the Fibonacci series), but also to practice proving closed-form expressions for recurrences by mathematical induction.
- ▶ In the next section, we describe a more advanced technique with which we cannot just **prove** the given expression but also **derive** it ourselves.

Golden Ratio

Definition (golden ratio)

The number

$$\varphi = \frac{1 + \sqrt{5}}{2}$$

is called the **golden ratio**.

German: **goldener Schnitt**

- ▶ Numerically, $\varphi = 1.618034$ (approximately).
- ▶ The golden ratio is a famous mathematical constant because it naturally occurs in many contexts and because of its aesthetical properties.

Negative Inverse of the Golden Ratio

Definition (negative inverse of the golden ratio)

The

$$\psi = \frac{1 - \sqrt{5}}{2}$$

is called the **negative inverse of the golden ratio**.

- ▶ Numerically, $\psi = -0.618034$ (approximately).
- ▶ The name for ψ derives from the fact that $\psi = -\frac{1}{\varphi}$.
However, we do not need this property here,
and therefore we do not prove it.

Fibonacci Series – Closed-Form Expression

Theorem

$$\begin{aligned} F(n) &= \frac{1}{\sqrt{5}} \left(\left(\frac{1 + \sqrt{5}}{2} \right)^n - \left(\frac{1 - \sqrt{5}}{2} \right)^n \right) \\ &= \frac{1}{\sqrt{5}} (\varphi^n - \psi^n) \end{aligned} \quad \text{for all } n \geq 0$$

Before we prove the theorem, we prove a number of lemmas.

- ▶ Note that $|\psi| < 1$ and hence $\psi^n \rightarrow 0$ as $n \rightarrow \infty$.
- ▶ With some calculation, we see that we can alternatively compute $F(n)$ by rounding $\frac{1}{\sqrt{5}}\varphi^n$ to the nearest integer, ignoring the ψ^n term.

First Lemma

Lemma

$$\psi = 1 - \varphi$$

Proof.

$$\begin{aligned}\psi &= \frac{1 - \sqrt{5}}{2} \\ &= \frac{1 + 1 - 1 - \sqrt{5}}{2} \\ &= \frac{2 - (1 + \sqrt{5})}{2} \\ &= \frac{2}{2} - \frac{1 + \sqrt{5}}{2} \\ &= 1 - \varphi\end{aligned}$$



Second Lemma

Lemma

$$\varphi^2 = \varphi + 1$$

Proof.

$$\begin{aligned}\varphi^2 &= \left(\frac{1 + \sqrt{5}}{2}\right)^2 = \frac{1}{4}(1 + \sqrt{5})^2 \\ &= \frac{1}{4}(1 + 2\sqrt{5} + 5) \\ &= \frac{1}{4}(2 + 2\sqrt{5} + 4) = \frac{1}{4}(2 + 2\sqrt{5}) + \frac{4}{4} \\ &= \frac{1}{2}(1 + \sqrt{5}) + 1 \\ &= \varphi + 1\end{aligned}$$



Third Lemma

Lemma

$$\psi^2 = \psi + 1$$

Proof.

$$\begin{aligned}\psi^2 &= (1 - \varphi)^2 \\ &= 1 - 2\varphi + \varphi^2 \\ &= 1 - 2\varphi + \varphi + 1 \\ &= 1 - \varphi + 1 \\ &= (1 - \varphi) + 1 \\ &= \psi + 1\end{aligned}$$



Main Proof (1)

Reminders:

$$\begin{array}{lll} F(0) = 0 & F(1) = 1 & F(n) = F(n-1) + F(n-2) \text{ for all } n \geq 2 \\ \varphi^2 = \varphi + 1 & \psi^2 = \psi + 1 & \text{Claim: } F(n) = \frac{1}{\sqrt{5}}(\varphi^n - \psi^n) \end{array}$$

Proof.

Proof by (strong) induction over n .

First base case $n = 0$:

$$\frac{1}{\sqrt{5}}(\varphi^0 - \psi^0) = \frac{1}{\sqrt{5}}(1 - 1) = 0 = F(0)$$

Second base case $n = 1$:

$$\begin{aligned} \frac{1}{\sqrt{5}}(\varphi^1 - \psi^1) &= \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2} - \frac{1-\sqrt{5}}{2}\right) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}-1+\sqrt{5}}{2}\right) \\ &= \frac{1}{\sqrt{5}}\left(\frac{2\sqrt{5}}{2}\right) = 1 = F(1) \end{aligned}$$

...

Main Proof (2)

Reminders:

$$\begin{array}{lll} F(0) = 0 & F(1) = 1 & F(n) = F(n-1) + F(n-2) \text{ for all } n \geq 2 \\ \varphi^2 = \varphi + 1 & \psi^2 = \psi + 1 & \text{Claim: } F(n) = \frac{1}{\sqrt{5}}(\varphi^n - \psi^n) \end{array}$$

Proof (continued).

Induction step (n building on $n-1$ and $n-2$):

$$\begin{aligned} F(n) &= F(n-1) + F(n-2) \\ &= \frac{1}{\sqrt{5}}(\varphi^{n-1} - \psi^{n-1}) + \frac{1}{\sqrt{5}}(\varphi^{n-2} - \psi^{n-2}) \\ &= \frac{1}{\sqrt{5}}(\varphi^{n-1} + \varphi^{n-2} - (\psi^{n-1} + \psi^{n-2})) \\ &= \frac{1}{\sqrt{5}}(\varphi^{n-2}(\varphi + 1) - \psi^{n-2}(\psi + 1)) \\ &= \frac{1}{\sqrt{5}}(\varphi^{n-2} \cdot \varphi^2 - \psi^{n-2} \cdot \psi^2) = \frac{1}{\sqrt{5}}(\varphi^n - \psi^n) \end{aligned}$$

