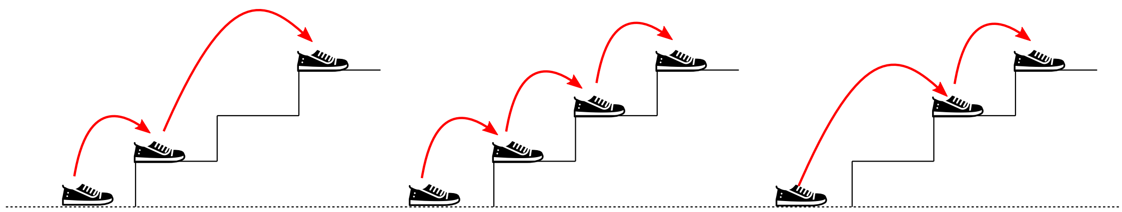
Stockholm University
Department of Mathematics
Lecturer: Marc Hellmuth
Tutor: Anna Lindeberg and Liam Cuclair

3. Exercise "Algorithm and Complexity (DA 4005)"

This exercise counts as a part of Individual Assignment 2

Exercise 1: Dynamic Programming 2+4+1=7

There are n stairs to climb and you stand at the bottom and want to reach the top. You can climb either 1 stair or 2 stairs at a time. The following figure shows all three different ways to climb $n = 3$ stairs.



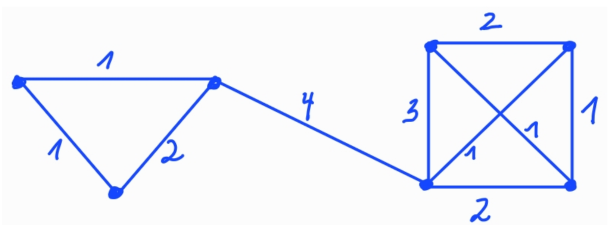
(a) Give a recursive formula for counting the number ways you can reach the top. Explain, why your formula works.

(b) Use this formula to provide a pseudocode for a dynamic programming approach that runs in polynomial time. Shortly explain that your algorithm is correct and has polynomial runtime.

(c) What is the number ways you can reach the top for $n = 20$ stairs?

Exercise 2: Kruskal's Algorithm and Matroids 1+(2+4)=7

Consider the following graph $G = (V, E)$:



(a) Draw into the graph G a minimum spanning tree that you find by applying Kruskal's algorithm.

(b) To recap, we defined the matroid $\mathcal{M}_G = (E, \mathbb{F})$ for a given graph G via

$$\mathbb{F} := \{F \subseteq E \mid (V, F) \text{ is acyclic } \}.$$

For G, as in the figure, determine …

(i) …the number of elements of a basis of $\mathcal{M}_G$.

(ii) …the number of different bases that $\mathcal{M}_G$ has.

Exercise 3: Greedy and Matroids 4+4+1=9

A *matching* in an undirected graph G is a subset $M \subseteq E(G)$ of edges such that no two edges in M share a common vertex, i.e., $e \cap f = \emptyset$ for all distinct $e, f \in M$.
Our goal is to find a (inclusion-)maximal matching, resp., maximum(-sized) matching M in a given undirected graph G.

(a) Show that finding a maximal matching can be achieved in linear-time in the size of the input $G = (V, E)$ if, in addition, the adjacency-list of G is provided as input.

See `https://en.wikipedia.org/wiki/Adjacency_list` for further details about adjacency-lists.

(b) Define the independence system $(E, \mathbb{F})$ that describes the problem of finding a maximum matching and also prove that $(E, \mathbb{F})$ is an independence system.

(c) Prove that a greedy algorithm will in general not optimally solve the problem of finding a maximum matching by showing that $(E, \mathbb{F})$ is not a matroid.

Deadline: Wednesday - October 1, 2025