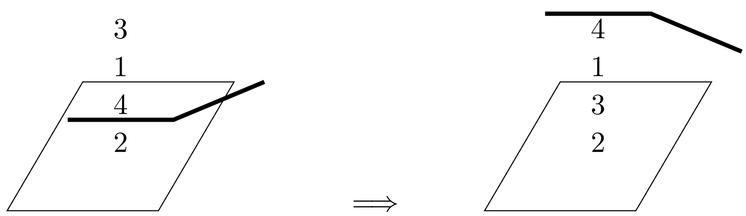
Seminar Theoretical Computer science. Februari 26th, 2018.

The class P

Exercise 1. Suppose some pancakes are stacked on a surface. No two pancakes have the same size. For convenience denote the smallest pancake by 1, the second smallest by 2, etc. The only permissible operation on the stack of pancakes is as follows: Insert a spatula between 2 pancakes or between the bottom pancake and the surface, then rotate the pancakes above the spatula. Give an algorithm that sorts any stack of n pancakes (smallest at the top largest at the bottom) with at most poly(n) many flips.



A solution for this example: $314|2 \to 4132| \to 23|14 \to 321|4 \to 1234.$[1]

Exercise 2. Construct a polynomial time algorithm that for any given undirected graph G finds out whether the nodes of G can be colored in red or blue so that every two adjacent nodes have different colors.

Exercise 3. Construct a polynomial time algorithm that for any given integers a, b, c computes $(a^b \bmod c)$. The numbers a, b, c are written in binary.

Exercise 4. Show that the following problem is in the class P.

Input:	A sequence of words $a_0, a_1, \ldots, a_n$ in the 1-letter alphabet $\{1\}$.
Question:	Is a_0 a concatenation of some words $a_1, \ldots, a_n$, i.e., $a_0 = a_{i_1} a_{i_2} \ldots a_{i_s}$ with $1 \le i_1 < \cdots < i_s \le n$.

Exercise 5. Let A be a language over an alphabet Σ. The Kleene star A^* is a language consisting of words in the form

$$a_1 a_2 \ldots a_k, \quad \text{where } a_i \in A,\ k \ge 0.$$

Prove that if $A \in \mathsf{P}$ then $A^* \in \mathsf{P}$.

The class NP

Exercise 6. Suppose $A, B \in \mathsf{NP}$. Is it true that $A \cup B \in \mathsf{NP}$? Is it true that $A \cap B \in \mathsf{NP}$?

Exercise 7. Prove that if $A \in \mathsf{NP}$ then $A^* \in \mathsf{NP}$.

Exercise 8. Show that if $\mathsf{P} = \mathsf{NP}$, then there exists an algorithm that on input a Boolean formula ϕ decides whether ϕ is satisfiable, and if it is satisfiable, returns an assignment of the variables that makes ϕ true. Moreover, the algorithm runs in time polynomial in the size of ϕ. (Note that with this technique, one can compute certificates for any problem in NP.)

---

[1] *Remark:* B. Gates and C. Papadimitriou gave a better algorithm that uses at most $5/3n$ flips in the worst case. Currently, the best known algorithm uses $\frac{18}{11}n$ flips. It is also known that no algorithm exists that always uses less than $\frac{15}{14}n$ flips. No better bounds are known.

Reductions and NP-completeness

Exercise 9. Show that if $\mathsf{P} = \mathsf{NP}$ then every language $A \in \mathsf{P}$ is NP-complete, except $A = \varnothing$ and $A = \Sigma^*$.

Exercise 10. Consider the following problems:

The problem IndSet:

Input:	A graph $G = (V, E)$ and a number $k \in \mathbb{Z}_{\geq 0}$.
Question:	Does there exist k points in V that are pairwise unconnected?

The problem 0-1 integer programming:

Input:	A matrix $A \in \mathbb{Z}^{m \times n}$ and a vector $b \in \mathbb{Z}^m$.
Question:	Is there an $x \in \{0, 1\}$ such that $Ax \leq b$.

Show that IndSet $\leq_p$ 0-1 integer programming.

Exercise 11. Prove that 3SAT reduces to halting problem

Input:	A description of a Turing machine M and a string w.
Question:	Does M halt on the input w?

Exercise 12. Show that Clique reduces to

The problem HalfClique:

Input:	A graph $G = (V, E)$
Question:	Does G have a clique of size more than $\lvert V \rvert/2$.