Discrete Mathematics II

Koen Oostveen

March 26, 2026

1 Integers, primes, Euclidean algorithm

Theorem 1.1 (Properties of $\mathbb{Z}, \mathbb{N}$)**.**

- $Z_{\geq 0} := \mathbb{N}$ *is* ***well-ordered****, that is, any subset has a least element by* $\leq$*. More formally, if* $S \subseteq \mathbb{N}$*, then there exists a unique* $m \in S$ *such that for all* $x \in S$*,* $m \leq x$*.*
- $(Z, \cdot)$ *has no zero divisors, that is, for all* $a, b \in \mathbb{Z}$*, if* $ab = 0$*, then* $a = 0$ *or* $b = 0$*.*

Corollary. *If* $a, x, y \in \mathbb{Z}$ *and* $a \neq 0$*, as well as* $ax = ay$*, then* $x = y$*. That is, the map* $x \mapsto ax$ *is injective for all* $a \neq 0$*.*
The proof is very elementary:

$$ax = ay \implies ax - ay = 0 \implies a(x-y) = 0 \implies a = 0 \vee x - y = 0 \overset{a\neq 0}{\implies} x - y = 0 \implies x = y$$

Definition 1.1. Let $a, b \in \mathbb{Z}$. We say that a **divides** b if there exists an $n \in \mathbb{Z}$ such that $b = na$. We define

$$a \mid b :\iff a \text{ divides } b$$

Theorem 1.2 (Properties of $\mid$)**.**

- *If* $a \mid b$*, then for all* $x \in \mathbb{Z}$*,* $a \mid bx$*.*
- *If* $x = y + z$ *and some* a *exists that divides two of* $x, y, z \in \mathbb{Z}$*, then it divides all.*
- *If* $a \mid b$ *and* $a \mid c$ *then for all* $x, y \in \mathbb{Z}$*,* $a \mid bx + cy$

Theorem 1.3. *Let* $a, b \in \mathbb{Z}$*, let* $b > 0$*, then there exist unique* $q, r \in \mathbb{Z}$ *with* $0 \leq r < b$ *such that*

$$a = qb + r$$

Proof. Let r be chosen in the equivalence class of a modulo b, with the particular unique representatitive that satisfies $0 \leq r < b$, which exists uniquely. We have $a - r = qb$ for some unique $q \in \mathbb{Z}$. From this clearly follows $a = qb + r$. To show that these particular choices for q, r are unique, let $\widetilde{q}, \widetilde{r} \in \mathbb{Z}$ such that $a = \widetilde{q}b + \widetilde{r}$. Clearly we still must have that $r \equiv \widetilde{r} \pmod{b}$, so that $r - \widetilde{r} = nb$ for some $n \in \mathbb{Z}$. Then $0 = b(q - \widetilde{q}) + r - \widetilde{r} = b(q - \widetilde{q} + n)$ so that $\widetilde{q} - q = n$. But then if $n \neq 0$ we get that $\widetilde{q} = q + n \geq n$, which is a contradiction, so $n = 0$, whence $\widetilde{q} = q$, which gives $0 = r - \widetilde{r}$, so that $r = \widetilde{r}$. □

Definition 1.2 (GCD)**.** Let $a, b \in \mathbb{Z}$. The largest $x \in \mathbb{Z}$ by $\leq$ such that $x \mid a$ and $x \mid b$ defines $\gcd(a, b) := x$.
This definition is sound, because the set of $x \in \mathbb{Z}$ for which this holds is nonempty and bounded above; consider that $1 \mid a$ and $1 \mid b$, so the set is nonempty, and the bound from above can be realized by the fact that if $x \mid a$ and $x \mid b$, then $x \leq a$ and $x \leq b$ (because $x > 0$ wlog), so in particular we have the upper bound $\min\{a, b\}$.
Note that additionally this set is finite, so in particular the maximum is attained and is unique. In this analysis above, we clearly ignore the cases for $a = 0$ or $b = 0$.

Theorem 1.4. *Let $a, b \in \mathbb{Z}^*$, then*

$$\gcd(a,b) = \min \underbrace{\{ax + by > 0 \mid x, y \in \mathbb{Z}\}}_{=:S}$$

Proof. From $\gcd(a,b) \mid a$ and $\gcd(a,b) \mid b$ it follows that there exist $p, q \in \mathbb{Z}$ such that $a = p\gcd(a,b)$ and $b = q\gcd(a,b)$. Clearly, $\gcd(a,b) \mid s$ for all $s \in S$, which also gives $s \geq \gcd(a,b)$, so the gcd is a lower bound of S. Now it thus suffices to show it lies in S. This involves ugly calculations I won't do. □

Corollary. *Let d be any divisor of a and b, then $d \mid \gcd(a,b)$*

Definition 1.3. Let $\gcd(a,b) = ax + by$, we say that these $x, y \in \mathbb{Z}$ are Bézout coefficients of a, b.

Definition 1.4. $a, b \in \mathbb{Z}$ are said to be **coprime** if $\gcd(a,b) = 1$.

Lemma 1.1. *Let $a, b \in \mathbb{Z}$, $n \in \mathbb{N}$, such that $nb \leq a$. Then $\gcd(a,b) = \gcd(a - nb, b)$.*

Corollary (Euclidean algorithm)**.** *Let $q, r \in \mathbb{Z}$ be according to a previous theorem such that $a = qb + r$. Then $\gcd(a,b) = \gcd(r,b)$, whence we have a sequence of remainders $(r_n)_{n\in\mathbb{N}}$ such that $r_0 = a$, $r_1 = b$, and then $r_{n+2} = r_n \mod r_{n+1}$ for all $n \in \mathbb{N}$, which eventually reaches 0, the last nonzero term then clearly equals $\gcd(a,b)$.*

Definition 1.5 (GCD)**.** Let $a, b \in \mathbb{Z}$. The smallest $x \in \mathbb{Z}$ by $\leq$ such that $a \mid x$ and $b \mid x$ defines $\operatorname{lcm}(a,b) := x$.

Theorem 1.5. *For all $a, b \in \mathbb{Z}$,*

$$ab = \gcd(a,b)\operatorname{lcm}(a,b)$$

Definition 1.6. Let $p \in \mathbb{N}^*$, then p is called **prime** if for all $x \in \mathbb{Z}$, $x \mid p$ implies $x = 1$ or $x = p$, whence for all $q \in \mathbb{N}$ with $q \neq p$ we have $\gcd(p,q) = 1$.

Lemma 1.2. *For all $n \in \mathbb{N}^*$, there exists a prime p such that $p \mid n$.*

Proof. If n is prime, we are done, great. Otherwise, there exists $1 < m < p$ such that $m \mid n$, so that there exists $q \in \mathbb{Z}$ with $n = qm$. By doing a hidden induction we know there exists a prime p such that $p \mid m$, whence $p \mid n$. □

Lemma 1.3. *If $p \mid ab$ then $p \mid a$ or $p \mid b$ for prime p.*

2 Graph theory

Definition 2.1. Let V be any set. Let $E \in \mathcal{P}(V \times V)$. We say that (V, E) is a (directed) **graph**. If for all $(v, w) \in E$ we also have $(w, v) \in E$, (V, E) is called **undirected**. We also identify E with the set

$$\{\{v, w\} \mid (v, w) \in E\}$$

for such an undirected graph, just for convenience. The remainder of this section deals only with undirected graphs.

Definition 2.2. Let $G = (V, E)$ be an undirected graph. Let $v \in V$ (called a **vertex**), let $e \in E$ (called an **edge**), then vertex v is called **incident** with edge e if $v \in e$.
Two vertices $v, w \in V$ are called **adjacent** if $\{v, w\} \in E$. We also define the following sets

$$N(v) := \{w \in V \mid w \neq v \wedge \{v, w\} \in E\}$$

$$\delta(v) := \{e \in E \mid v \in e\}$$

respectively called the **neighborhood** and **incident edges** of v. We say that the **degree** of v, denoted by $d(v) := |N(v)| = |\delta(v)|$.

Definition 2.3. Let $G = (V, E)$ be a graph. We say that a graph $\widetilde{G} = (\widetilde{V}, \widetilde{E})$ is a **subgraph** of G if $\widetilde{V} \subseteq V$, $\widetilde{E} \subseteq E$, and of course because $\widetilde{G}$ is a graph, also $\widetilde{E} \subseteq \widetilde{V} \times \widetilde{V}$.

Definition 2.4. Let $G = (V, E)$ be a graph. We say that a subgraph $\widetilde{G} = (\widetilde{V}, \widetilde{E})$ is **induced** by $\widetilde{V}$, if $\widetilde{E}$ is the **largest subset** of E such that $\widetilde{G}$ is a graph. Explicitely, this set is

$$\widetilde{E} := \{\{v, w\} \in E \mid v \in \widetilde{V} \wedge w \in \widetilde{V}\}$$

Definition 2.5. Let $G = (V, E)$ be a graph. A **walk** of G is a finite sequence $v_1, \ldots, v_n \in V$ such that for all $1 \leq i < n$ we have $\{v_i, v_{i+1}\} \in E$.
A **closed walk** of G is a walk $v_1, \ldots, v_n$ such that $v_1 = v_n$.
A **trail** of G is a walk of G such that every edge $\{v_i, v_{i+1}\}$ occurs as most once. A **circuit** of G is a closed walk of G that is also a trail.
A **path** of G is a walk of G such that every vertex v_i occurs at most once. A **cycle** of G is a path of G that is also a closed walk of G.

Lemma 2.1. *The number of vertices with odd degree is even.*

Definition 2.6. Let $G = (V, E)$ be a graph. G is said to be **connected** if for any two $v, w \in V$, there exists a path starting at v and ending in w. That is, a path $v_1, \ldots, v_n$ such that $v_1 = v$, $v_n = w$.

Lemma 2.2. *Any graph $G = (V, E)$ can be written as the disjoint union of at most $|V|$ connected subgraphs. This decomposition is unique, we call these connected subgraphs the **connected components** of G.*

Definition 2.7. An edge $e \in E$ in a graph $G = (V, E)$ is called a **bridge** if $(V, E \setminus \{e\})$ has more connected components than G.

Lemma 2.3. *If $G = (V, E)$ is connected, then $|V| \leq |E| + 1$.*

Definition 2.8. An undirected graph G is called a **forest** if G has no cycles, that is, for any closed walk of the graph, said walk is not a cycle. A forest G is called a **tree** if it is connected.

Theorem 2.1. *For an undirected graph $G = (V, E)$ tfae*

- *G is a tree.*
- *G is a forest with $|E| = |V| - 1$.*
- *G is connected with $|E| = |V| - 1$.*
- *There exists a unique path between any two vertices*
- *G is a forest, and introducing one $e \in \mathcal{P}(V \times V) \setminus E$ into G, that is, constructing $\widetilde{G} := (V, E \cup \{e\})$, gives $\widetilde{G}$ exactly one cycle.*

2.1 Euler graphs

Definition 2.9. A graph is called **Eulerian** if there exists a closed walk (circuit) that visits every edge exactly once. Said walk is called an **Euler walk** or **Euler tour**.

Theorem 2.2. *A graph is Eulerian if and only if it is connected and every vertex has even degree.*

2.2 Shortest path

Let $G = (V, E)$ be a **directed graph**. Note that undirected graphs are also allowed, because essentially they form a subset of the directed graphs, the identity is an inclusion, unless E was chosen to be a subset of all doublett sets of V, in which case we have the injection

$$E \mapsto \{(v, w) \mid \{v, w\} \in E\} \cup \{(w, v) \mid \{v, w\} \in E\}$$

We are also given a **cost function** $c : E \to \mathbb{R}$, and two vertices s, t. The goal is to find a path P (if it exists) consisting of vertices $v_1, \ldots, v_n$ such that $v_1 = s$, $v_n = t$, and

$$\sum_{i=1}^{n-1} c(v_i, v_{i+1}) =: c(P)$$

is minimized. Such a path is called a shortest path from s to t.
If we fix $s \in V$, then we can define the metric $\mathrm{dist} : V \to \mathbb{R}$ on V with

$$v \mapsto \inf\{c(P) \in \mathbb{R} \mid P \text{ is a walk from } s \text{ to } v\}$$

which gives rise to a type of triangle inequality, if $(u, v) \in E$ is an edge, then

$$\mathrm{dist}(v) \leq \mathrm{dist}(u) + c(u, v)$$

and if this equality is achieved, we call (u, v) **tight**. This equality is obviously achieved if $u = s$. Remember that '$\inf \varnothing := \infty$', so we need to be a bit careful, this inequality holds whenever the distances are real and defined.

Lemma 2.4. *A subpath of a shortest path is also a shortest path. A subpath, by the way, is just a subsequence of the path, which provably is still a path.*

So, as we can see, having this distance metric gives us a triangle inequality. It turns out that any labelling $d : V \to \mathbb{R}$ that **overestimates** dist, that is, for any $v \in V$, $d(v) \geq \mathrm{dist}(v)$, if it satisfies this triangle inequality, we get that $d = \mathrm{dist}$. This follows easily by 'walking the path' through the triangle inequality, i.e. if we have a shortest path $s = v_1, \ldots, v_n = t$, then

$$d(t) \leq d(v_{n-1}) + c(v_{n-1}, v_n) \leq d(v_{n-2}) + c(v_{n-2}, v_{n-1}) \leq \ldots \leq d(s) + c(P) = c(P) = \mathrm{dist}(t)$$

and because $d(t) \geq \mathrm{dist}(t)$ we are done.
This way, we can solve shortest path, by starting with a distance metric $d : V \to \mathbb{R}$ with $d(s) = 0$, $d(v) = \infty$ for all $v \in V \setminus \{s\}$, and then 'fixing' triangle inequalities as we go. If we detect that no more inequalities can be fixed, we have found a labelling from which we can infer the shortest path. If we choose the labelling d cleverly, we can also get the distances.
In practice this means the following: if we have a labelling d, we can 'fix' the inequality for an edge (u, v) if $d(v) > d(u) + c(u, v)$, by making the above an equality in a new labelling $\widetilde{d}$. The procedure that does this in-place is called `relax`.

2.2.1 Bellman-Ford

Call `relax` on every edge $|V| - 1$ times, which guarantees that at the end, $d = \mathrm{dist}$. Why? Let $s = v_1, \ldots, t = v_n$ be a shortest path. At every iteration i of the algorithm, we have the invariant that $d(v_i) = \mathrm{dist}(v_i)$, the weights kind of 'propagate' throughout the graph. Obviously this relaxation then won't update d any more if $i > n$, but $n \leq |V| - 1$, so that is how many iterations we need at most. This obviously **only works if there are no negative cycles in the graph**.

2.2.2 Dijkstra's algorithm

We can skip a lot of the work by performing the `relax` procedure in a certain order. Namely, we can relax those edges outgoing from vertices that we order by the smallest tentative label d. We can keep track of all vertices that we still need to process, and then store in a priority queue the tentative total distance of a vertex. Any time we remove the smallest element v from the queue and relax all of its outgoing edges, we have the invariant that $d(v) = \text{dist}(v)$, so the algorithm is correct. This does assume that **all weights are nonnegative**.

3 Max-flow min-cut

Definition 3.1. Let $G = (V, E)$ be a connected directed graph, let $c : E \to \mathbb{N}$, let $s, t \in V$. A function $f : E \to \mathbb{R}$ is called an (s, t)-flow of G if

- For all $e \in E$, $0 \leq f(e) \leq c(e)$.
- For all $u \in V \setminus \{s, t\}$,
$$\sum_{v|(u,v)\in E} f(u, v) = \sum_{v|(v,u)\in E} f(v, u)$$
Intuitively, the outgoing flow must equal the incoming flow.

We define the **value** of the flow f, $\text{val}(f)$, through

$$\text{val}(f) := \sum_{v|(s,v)\in E} f(s, v) - \sum_{v|(v,s)\in E} f(v, s)$$

i.e. how much flows out of s, which also equals how much flows into t due to the conservation property.

Definition 3.2. The max-flow problem is the following: given $G = (V, E)$ a connected directed graph, $c : E \to \mathbb{N}$ a capacity function, and $s, t \in V$ vertices, find a flow f that maximizes $\text{val}(f)$. It is useful to pack this data into a tuple, so we call (G, c, s, t) a max-flow problem.

Definition 3.3. Let (G, c, s, t) be a max-flow problem. Let f be a valid flow for this problem. The **residual capacity** with respect to f, $r_f : E \to \mathbb{R}$, is defined by first studying the simplified flow f. We can simplify f by 'cancelling' flow, if $f(u, v) > 0$ and $f(v, u) > 0$, we keep the greater edge, subtract, and delete the smaller edge (i.e. set its flow to 0). Then, for any edge $(u, v) \in E$, define

$$r_f(u, v) := \begin{cases} c(u, v) - f(u, v) & \text{if } f(u, v) > 0 \\ f(v, u) + c(u, v) & \text{otherwise} \end{cases}$$

Intuitively, this gives a notion of 'how much flow we can still send through, or back'.
We define the **residual graph**, G_f, with respect to f, by $G_f := (V, E_f)$, where

$$E_f := \{e \in V \times V \mid r_f(e) > 0\}$$

We define the **residual max-flow problem** to be (G_f, r_f, s, t).

Lemma 3.1. *Let f be a flow in (G, c, s, t) and let g be a flow in (G_f, r_f, s, t). Then the pointwise sum $h := f + g$, is a flow in (G, c, s, t), where* $\text{val}(h) = \text{val}(f) + \text{val}(g)$.

Definition 3.4. An **augmenting path** for the flow f is an (s, t)-path P (a path from s to t) in the residual graph G_f. Define the **residual capacity** of the path P through

$$r_f(P) := \min\{r_f(e) \mid e \in P\}$$

Lemma 3.2. *Let f be a flow in (G, c, s, t), let P be an augmenting path, we define the **augmented flow** $\widetilde{f} : E \to \mathbb{R}$ by*

$$\widetilde{f}(e) := \begin{cases} f(e) + r_f(P) & \text{if } e \in P \\ f(e) & \text{otherwise} \end{cases}$$

Then $\widetilde{f}$ is a flow in (G, c, s, t).

Proof. $\widetilde{f} = f + g$ where $g := r_f(P)\mathbf{1}_P$, and g is a flow in the residual problem. □

Theorem 3.1. *A flow f solves the max-flow problem (G, c, s, t) if and only if it has no augmenting paths.*

3.1 Ford-Fulkerson algorithm

Very simple, choose an augmenting path, augment the graph, repeat until no more augmenting graphs exist. Note that the trivial flow defined pointwisely by $f := 0$ yields a flow on any graph.
A result from linear optimization says

Theorem 3.2. *If $c : E \to \mathbb{R}$ are integers, that is, for all $e \in E$, $c(e) \in \mathbb{N}$, then the maximum flow $f : E \to \mathbb{R}$ also is integer-valued, that is, for all $e \in E$, $f(e) \in \mathbb{N}$.*

3.2 Edmonds-Karp algorithm

Essentially the same as before, but instead of selecting **any** augmenting path, we select the shortest, where the shortest path is defined by the amount of edges in the path. This can be found by way of breadth-first search.

3.3 Cuts

Definition 3.5. Let (V, E, c, s, t) be a max-flow problem. A **cut** (S, T) is a partition $V = S \sqcup T$, such that $s \in S$, $t \in T$. The **edges** of the cut (S, T) are those that do not lie in the subgraphs of V when restricting to either S or T, that must be in-edges in S and out-edges in T. That is, they are the set

$$\{(u, v) \in E \mid u \in S \wedge v \in T\} = (S \times T) \cap E$$

We say that the **capacity of the cut** (S, T), $c(S, T)$, is the sum of the capacities of these edges.
We say that the cut (S, T) is a min-cut if it minimizes the capacity.

Theorem 3.3. *For any flow f and cut (S, T) of (V, E, c, s, t), we have that*

$$\mathrm{val}(f) \leq c(S, T)$$

Theorem 3.4. *For any max-flow problem (V, E, c, s, t), there exists a max-flow f and a min-cut (S, T), such that*

$$\mathrm{val}(f) = c(S, T)$$

4 Stable matchings

Definition 4.1. Let $G = (V, E)$ be an undirected graph. We say that a subset $M \subseteq E$ is a **matching** of G if every vertex is incident to at most one edge in M. If every vertex is incident to exactly one edge in M, we call the matching **perfect**.

Lemma 4.1. *Given a perfect matching M, there exists a unique function $p_M : V \to V$ that is an involution (self-inverse), such that for all $v \in V$, we have that $\{v, p_M(v)\} \in M$. This function we call the **partner function** on M.*

Definition 4.2. A graph $G = (V, E)$ is called **bipartite** if there exists a decomposition $V = V_M \sqcup V_W$ such that every edge is indicent to exactly one vertex in V_M and exactly one in V_W.

Definition 4.3. A bipartite graph $(V_M \sqcup V_W, E)$ is called **bipartite complete** if for all $m \in V_M$ and $w \in V_W$, $\{m, w\} \in E$.

Definition 4.4. Let $(V_M \sqcup V_W, E)$ be a bipartite graph. Suppose that for every $v \in V_*$ (the asterisk is a placeholder), there exists a strict total order $<_v$ on $V \setminus V_*$. These are, in a sense, **preference lists** for matchings. A pair $(m, w) \in V_M \times V_W$ is said to be **unstable** with respect to a matching $M \subseteq V$ if both $p(m) <_m w$ and $p(w) <_w m$.

Definition 4.5. A perfect matching M on a bipartite graph $(V_M \sqcup V_W, E)$ is said to be **stable** if no unstable pairs exist, that is, there do not exist $m \in V_M$ and $w \in V_W$ such that (m, w) is unstable. The **stable matching problem** is concerned with whether, given a bipartite (complete) graph, a stable matching exists. We study the case where $|V_M| = |V_W|$.

4.1 Gale-Shapley algorithm

A solution to this problem is simple: add the 'best possible pair' (m, w) if both m and w are not matched yet, replace (when encountered) unstable pairs with stable ones. Rinse and repeat until every $v \in V$ is matched. The 'best possible pair' is decided by either the preference of m or w. This always yields a perfect stable matching.
Problem: a stable matching is definitely not unique. In particular, when one decides by preference of w rather than m for example, we probably get a different stable matching. One matching, in this setup, is not better than the other in any sense, so long as they are stable. Individual $v \in V$ with their preference lists may prefer one matching over the other, though. The Gale-Shapley algorithm favors the 'proposing side', that is, the part of the graph (either V_M or V_W) whose preference is considered first.

Theorem 4.1. *The Gale-Shapley algorithm is proposing-optimal, that is, if the preference of V_* is considered first, every element of V_* will be matched up with the best possible element of $V \setminus V_*$.*
The Gale-Shapley algorithm is proposee-pessimal, that is, if the preference of V_* is considered last (not first), every element of V_* will be matched up with the worst possible element of $V \setminus V_*$.

5 Recurrence relations

Definition 5.1. A **recurrence relation** of order k is given by a function $g : \mathbb{R}^k \to \mathbb{R}$, numbers $a_0, \ldots, a_{k-1} \in \mathbb{R}$, such that a sequence $a : \mathbb{N} \to \mathbb{R}$ satisfies the following: for all $n \in \mathbb{N}$, if $n < k$ we have $a(n) = a_n$, otherwise: $a(n) = g(a_{n-1}, \ldots, a_{n-k})$. If this is the case for a it is called a **solution to the recurrence relation**. We can pack the numbers $a_0, \ldots, a_{k-1}$ in a vector $b \in \mathbb{R}^k$, so we can represent the recurrence relation as a tuple (k, g, b). Informally, we simply write: given $a_0, \ldots, a_{k-1}$, we have

$$a_n = g(a_{n-1}, \ldots, a_{n-k}), \quad \forall n \geq k$$

Theorem 5.1. *Every recurrence relation has a unique solution.*

Proof. Kind of trivial, the unique solution is the function defined by the recurrence relation. The difficulty, of course, is finding a representation of the function a independent of g. □

Example 5.1. Consider the relation $a_n = Ca_{n-1}$ for all $n \geq 1$, $a_0 \in \mathbb{R}$ and $C \in \mathbb{R}$ are given. The unique solution is given by $a_n = a_0 C^n$ for all $n \in \mathbb{N}$, which we can show: if $n = 0$ then $a_0 = a_0 \cdot C^0 = a_0$. If $n \geq 1$ we consider that

$$a_0 C^n = a_n = Ca_{n-1} = Ca_0 C^{-1} = a_0 C^n$$

Remark 5.1. This previous example is that of a linear first-order recurrence relation. Those are quite boring, for the rest of the lecture we deal with linear second-order recurrence relations of the form

$$a_{n+2} + C_1 a_{n+1} + C_2 a_n = f(n), \quad \forall n \in \mathbb{N}$$

with $C_1, C_2 \in \mathbb{R}^*$ and given $a_0, a_1 \in \mathbb{R}$.

Remark 5.2. We can identify a k-th order linear recurrence relation with just the function f, the coefficients $v \in \mathbb{R}^k$ and the initial sequence $a_0, \ldots, a_{k-1} \in \mathbb{R}$ with a vector b.

Definition 5.2. A linear kth-order recurrence relation $a_{n+k} + C_1 a_{n+k-1} + \ldots + C_k a_n = f(n)$ is called **homogeneous** if $f = 0$ pointwisely.

Theorem 5.2. *Let $k \in \mathbb{N}$. Let $\mathcal{A}_k$ be the set of all solutions to homogeneous linear k-th order recurrence relations (which are functions $a : \mathbb{N} \to \mathbb{R}$) with **the same coefficients**. Then $\mathcal{A}_k$ can be equipped with a vector space structure over $\mathbb{R}$ with pointwise addition and scalar multiplication in $\mathbb{R}$. Furthermore,* $\dim \mathcal{A}_k = k$.

Proof. The vector space structure with $+(a, b) \mapsto a + b$ defined by $a + b : \mathbb{N} \to \mathbb{R}$ with $(a + b)(n) := a(n) + b(n)$ for all $n \in \mathbb{N}$ and $a, b : \mathbb{N} \to \mathbb{R}$ as well as $\cdot(\lambda, a) \mapsto \lambda a$ defined by $\lambda a : \mathbb{N} \to \mathbb{R}$ with $(\lambda a)(n) := \lambda a(n)$ for all $n \in \mathbb{N}$ and $a : \mathbb{N} \to \mathbb{R}$ is clear. It suffices to show that $a + b$ also lies in $\mathcal{A}_k$. Let

$$a_{n+k} + C_1 a_{n+k-1} + \ldots + C_k a_n = 0$$

and

$$b_{n+k} + C_1 b_{n+k-1} + \ldots + C_k b_n = 0$$

be the recurrence relations of a and b, respectively. Then $a_0 + b_0, \ldots, a_{k-1} + b_{k-1} \in \mathbb{R}$ are the initial values of $a + b$ (clearly), and

$$\begin{aligned}(a + b)_{n+k} + C_1 (a + b)_{n+k-1} + \ldots + C_k (a + b)_n &= a_{n+k} + C_1 a_{n+k-1} + \ldots + C_k a_n \\ &\quad + b_{n+k} + C_1 b_{n+k-1} + \ldots + C_k b_n \\ &= 0 + 0 = 0\end{aligned}$$

Scalar multiplication is also trivial. To show that $\dim \mathcal{A}_k = k$, we define the map $\pi : \mathcal{A}_k \to \mathbb{R}^k$ by $a \mapsto (a_0, \ldots, a_{k-1})$ and show it is an isomorphism of vector spaces. It is easily linear. Bijectivity is also clear. □

Remark 5.3. Now then to solve a linear homogeneous recurrence relation, all we need is to find a basis for $\mathcal{A}_k$ (in terms of $C_1, \ldots, C_K$), and then take a linear combination to arrive at any solution, which linear combination to take is essentially a linear system. Namely, suppose we have a basis $a^1, \ldots, a^k$, and we want to find $c_1, \ldots, c_k$ such that a has solution $a_n = c_1 a_n^1 + \ldots + c_k a_n^k$, then all we need is for this to hold for $n < k$, which gives rise to a linear map $A : \mathbb{R}^k \to \mathbb{R}^k$ such that $A(e_i) = \pi(a^i)$ in the canonical basis. The solution coefficients $c = (A^*)^{-1}(a_0, \ldots, a_{k-1})$. Why? Let's consider the situation componentwise. Let $i < k$.

$$a_i = c_1 a_i^1 + \ldots + c_k a_i^k$$

and

$$\langle c, a_i \rangle = \langle c, A^* e_i \rangle = \langle Ac, e_i \rangle = (Ac)^i = A^i_j c^j = \sum_j \pi(a^j)^i c^j = \sum_j a^j_i c^j$$

which are equal, so this choice of A leads to the right linear system.
Solving now amounts to finding a basis, which we can do if we find linearly independent solutions (note that these do not depend on the initial conditions).

Theorem 5.3. *Let $a_{n+k} + C_1 a_{n+k-1} + \ldots + C_k a_n = 0$ be a linear homogeneous recurrence relation. Then the sequence $a : \mathbb{N} \to \mathbb{R}$ defined by $a_n := \lambda^n$ is a solution if and only if $\lambda \in \mathbb{R}$ satisfies $\lambda^k + C_1 \lambda^{k-1} + \ldots + C_k = 0$ (this polynomial on the left in called the **characteristic polynomial**).*

Proof. We first prove 'if': suppose λ satisfies the characteristic polynomial equation. We compute

$$\begin{aligned} a_{n+k} + C_1 a_{n+k-1} + \ldots + C_k a_n &= \lambda^{n+k} + C_1 \lambda^{n+k-1} + \ldots + C_k \lambda^n \\ &= \lambda^n \left(\lambda^k + C_1 \lambda^{k-1} + \ldots + C_k \right) = 0 \end{aligned}$$

The initial conditions that we need are simply $a_i := \lambda^i$, $i < k$, which is satisfied by a as well.
What about only if? This is essentially the same logic which we can also do by contraposition. □

Example 5.2. Fibonacci thyme. Let $F_0 := 0$, $F_1 := 1$ and

$$F_{n+2} := F_{n+1} + F_n \iff F_{n+2} - F_{n+1} - F_n = 0$$

Then the roots of the characteristic polynomial are ϕ and ϕ^*, the golden ratio and its complement. Proof:

$$\lambda^2 - \lambda - 1 = 0 \iff \lambda = \frac{1 \pm \sqrt{1+4}}{2} = \frac{1 \pm \sqrt{5}}{2}$$

So $a^1(n) := \phi^n$ and $a^2(n) := (\phi^*)^n$ are basic solutions. We let $F := c_1 a^1 + c_2 a^2$. We need to solve:

$$\begin{bmatrix} 1 & 1 \\ \phi & \phi^* \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

so $c_1 = -c_2$ and $c_1 = 1/(\phi - \phi^*) = \frac{1}{\sqrt{5}}$, whence

$$F_n = \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^n - \left(\frac{1-\sqrt{5}}{2} \right)^n \right)$$

Example 5.3. $a_0 := 1$, $a_1 := 2$, and

$$a_{n+2} - 2a_{n+1} + 2a_n = 0$$

The characteristic polynomial:

$$\lambda^2 - 2\lambda + 2 = 0 \iff \lambda = \frac{2 \pm 2i}{2} = 1 \pm i$$

Note that we have basic solutions

$$(a^1)_n := \lambda^n, \quad (a^2)_n := \overline{\lambda^n}$$

which do give a valid solution, but we can find a real basis from this, to get real solutions. Consider that $z + \bar{z} = 2 \operatorname{Re} z$ and $z - \bar{z} = 2i \operatorname{Im} z$. And indeed, these are linearly independent, so in principle we can take

$$(\widetilde{a}^1)_n := \operatorname{Re} \lambda^n, \quad (\widetilde{a}^2)_n := \operatorname{Im} \lambda^n$$

How do we find these? Find the polar form of $\lambda = r\exp(i\theta)$, then $\lambda^n = r^n\exp(in\theta)$. Finally:

$$(\widetilde{a}^1)_n = \operatorname{Re}\lambda^n = \operatorname{Re} r^n\exp(in\theta) = r^n\cos(n\theta), \quad (\widetilde{a}^2)_n = r^n\sin(n\theta)$$

The polar form is clearly $1 + i = \sqrt{2}\exp(i\pi/4)$

Example 5.4. Consider the recurrence relation

$$a_{n+2} - 2\lambda a_{n+1} + \lambda^2 a_n = 0$$

Note that its characteristic polynomial has only the root λ with multiplicity 2, it doesn't generate two linearly independent solutions. But in this case, we can get another: $n \mapsto n\lambda^n$

$$(n+2)\lambda^{n+2} - 2\lambda(n+1)\lambda^{n+1} + \lambda^2 n\lambda^n = \lambda^{n+2}(n + 2 - 2n - 2 + n) = 0$$

6 RSA

Definition 6.1. Let G be a set, and let $\cdot : G \times G \to G$ be a binary operation on G. The pair $(G, \cdot)$ is called a **group** if:

- For all $x, y, z \in G$, we have that
$$(x \cdot y) \cdot z = x \cdot (y \cdot z)$$
- There exists a $e \in G$ such that for all $x \in G$, we have that $e \cdot x = x \cdot e = x$. Note that this element is unique: let $\widetilde{e} \in G$ be another satisfying this axiom, then
$$\widetilde{e} = \widetilde{e} \cdot e = e \cdot \widetilde{e} = e$$
whence $\widetilde{e} = e$.
- For all $x \in G$, there exists an $x^{-1} \in G$, such that $x \cdot x^{-1} = x^{-1} \cdot x = e$. Note that for all particular $x \in G$ this element is unique, let $\widetilde{x}^{-1}$ be another, then
$$\widetilde{x}^{-1} = \widetilde{x}^{-1} \cdot e = \widetilde{x}^{-1} \cdot (x \cdot x^{-1}) = (\widetilde{x}^{-1} \cdot x) \cdot x^{-1} = e \cdot x^{-1} = x^{-1}$$
whence $\widetilde{x}^{-1} = x^{-1}$.

Definition 6.2. Let $(G, \cdot)$ be a group. Let $n \in \mathbb{N}$. Let $x \in G$. We define the map $m_x : G \to G$ through $y \mapsto x \cdot y$. Then we define the following symbol:
$$x^n := (m_x)^n(e) = \underbrace{x \cdot x \cdot \ldots x}_{n \text{ times}}$$
We additionally define (to extend to $n \in \mathbb{Z}$) $x^{-n} := (x^n)^{-1}$. Note that this notation is consistent with our existing x^{-1} notation, as well as: $x^0 = (m_x)^0(e) = \mathrm{id}_G(e) = e$.

Lemma 6.1. *Let $j, k \in \mathbb{Z}$, let $x \in G$, then*
$$(x^j)^k = x^{jk}, \qquad x^j \cdot x^k = x^{j+k}$$

Definition 6.3. Let $(G, \cdot)$ be a group. A **subgroup** of G is a group $(F, \cdot|_F)$ where $F \subseteq G$ and the group relation is inherited. Note that G is then a subgroup of G.

Definition 6.4. Let $(G, \cdot)$ be a group. Let $a \in G$. Then we define the following set
$$\langle a \rangle := \{a^k \mid k \in \mathbb{Z}\}$$
We equip it with $\cdot$, which makes $(\langle a \rangle, \cdot|_{\langle a \rangle})$ a group, called the **subgroup of G generated by a**.

Definition 6.5. A group $(G, \cdot)$ is called **cyclic** if there exists an $x \in G$ such that $G = \langle x \rangle$.

Theorem 6.1. *If the subgroup $\langle x \rangle$ generated by $x \in G$ is finite, there exists $n \in \mathbb{N}$ such that*
$$\langle x \rangle = \left\{x, x^2, \ldots, x^{n-1}, x^n = e\right\}$$

Theorem 6.2. *Let $(G, \cdot)$ be a finite cyclic group of size $n \in \mathbb{N}^*$ with generator a. Then for all $k \in \mathbb{N}^*$, a^k is also a generator for G if and only if* $\gcd(k, n) = 1$.

Corollary. *Every finite subgroup of the same size has the same amount of generators. The function $\phi : \mathbb{N}^* \to \mathbb{N}^*$ that counts the amount of generators $\phi(n)$ for a finite cyclic group of size n is called the* ***Euler totient function****.*

Lemma 6.2. *If $a, b \in \mathbb{N}^*$ are coprime, then $\phi(ab) = \phi(a)\phi(b)$.*

Lemma 6.3. *Let $p > 1$ be prime and let $e \in \mathbb{N}^*$, then $\phi(p^e) = p^e - p^{e-1}$.*

Theorem 6.3. *Let $p_1, \ldots, p_k$ be all prime divisors of $n \in \mathbb{N}^*$, then*

$$\phi(n) = n \prod_{i=1}^{k} (1 - 1/p_i)$$

Proof. Let $n = p_1^{e_1} \cdot \ldots \cdot p_k^{e_k}$ be the prime factorisation of n. Then

$$\begin{aligned}
\phi(n) &= \phi(p_1^{e_1} \cdot \ldots \cdot p_k^{e_k}) \\
&= \phi(p_1^{e_1}) \cdot \ldots \cdot \phi(p_k^{e_k}) \\
&= \prod_{i=1}^{k} p_i^{e_i}(1 - 1/p_i) \\
&= \prod_{i=1}^{k} p_i^{e_i} \prod_{i=1}^{k} (1 - 1/p_i) \\
&= n \prod_{i=1}^{k} (1 - 1/p_i)
\end{aligned}$$

□

Definition 6.6. Let $n \in \mathbb{N}$ with $n \geq 2$, define the following set

$$U_n := \{a \in \mathbb{Z}/n\mathbb{Z} \mid \forall x \in \mathbb{Z}/n\mathbb{Z} : ax = 1\} = \{a \in \mathbb{Z}/n\mathbb{Z} \mid \gcd(a, n) = 1\}$$

This is a subgroup of $\mathbb{Z}/n\mathbb{Z}$. Its size (order) is $\phi(n)$ (trivially).

Theorem 6.4 (Lagrange)**.** *Let $(G, \cdot)$ be a finite group. Let $(H, \cdot|_H)$ be a subgroup of G. Then $|H|$ divides $|G|$.*

Corollary. *If a finite subgroup has prime $|G|$, then G is cyclic.*

Corollary. *Let $a \in G$, then $a^{|G|} = e$.*

Proof. Let $H := \langle a \rangle$, then $|H|$ divides $|G|$, so that there exists $k \in \mathbb{N}$ such that $|G| = k|H|$. Notice that

$$a^{|G|} = a^{k|H|} = (a^|H|)^k = e^k = e$$

since $a^{|H|} = e$ because of the subgroup property. □

Theorem 6.5. *Let $n > 1$, $M \in \mathbb{Z}$. If $\gcd(M, n) = 1$, then*

$$M^{\phi(n)} \equiv 1 \pmod 1$$