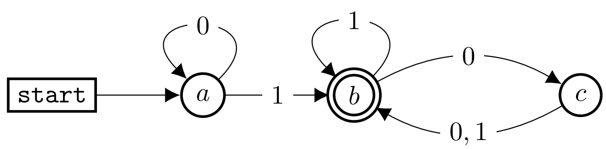
Finite Automata

Prepared by Mark and Nikita on May 9, 2025

Part 1: DFAs

This week, we will study computational devices called *deterministic finite automata*.
A DFA has a simple job: it will either "accept" or "reject" a string of letters.

Consider the automaton A shown below:



A takes strings of letters in the alphabet $\{0, 1\}$ and reads them left to right, one letter at a time. Starting in the state a, the automaton A will move between states along the edge marked by each letter.

Note that node b has a "double edge" in the diagram above. This means that the state b is *accepting*. Any string that makes A end in state b is *accepted*. Similarly, strings that end in states a or c are *rejected*.

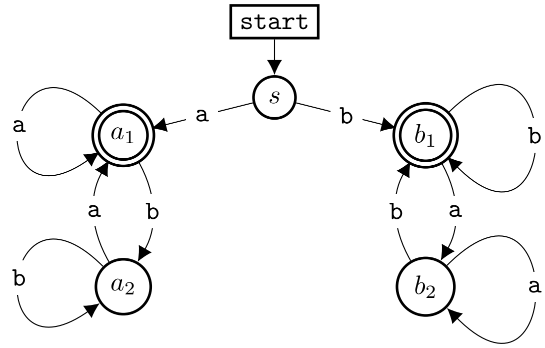
For example, consider the string `1011`.
A will go through the states $a - b - c - b - b$ while processing this string.

Problem 1:
Which of the following strings are accepted by A?

- `1`
- `1010`
- `1110010`
- `1000100`

Problem 2:
Describe the general form of a string accepted by A.
Hint: Work backwards from the accepting state, and decide what all the strings must look like at the end in order to be accepted.

Now consider the automaton B, which uses the alphabet $\{a, b\}$.
It starts in the state s and has two accepting states a_1 and b_1.



Problem 3:
Which of the following strings are accepted by B?

- `aa`
- `abba`
- `abbba`
- `baabab`

Problem 4:
Describe the strings accepted by B.

Before we continue, let's properly define all the words we've been using in the problems above.

Definition 5:
An *alphabet* is a finite set of symbols.

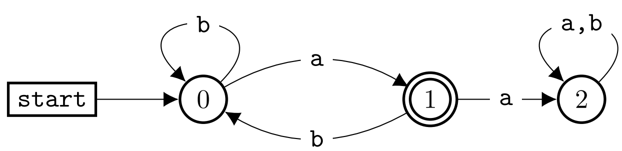
Definition 6:
A *string* over an alphabet Q is a finite sequence of symbols from Q.
We'll denote the empty string ε.

Definition 7:
Q^* is the set of all possible strings over Q.
For example, $\{\mathtt{0}, \mathtt{1}\}^*$ is the set $\{\varepsilon, \mathtt{0}, \mathtt{1}, \mathtt{00}, \mathtt{01}, \mathtt{10}, \mathtt{11}, \mathtt{000}, ...\}$
Note that this set contains the empty string.

Definition 8:
A *language* over an alphabet Q is a subset of Q^*.
For example, the language "strings of length 2" over $\{\mathtt{0}, \mathtt{1}\}$ is $\{\mathtt{00}, \mathtt{01}, \mathtt{10}, \mathtt{11}\}$

Definition 9:
The language *recognized* by a DFA is the set of strings that the DFA accepts.

Problem 10:
How many strings of length n are accepted by the automaton below?
Hint: Induction.



Problem 11:
Draw DFAs that recognize the following languages. In all parts, the alphabet is $\{0, 1\}$:

- $\{w \mid w$ begins with a `1` and ends with a `0`$\}$
- $\{w \mid w$ contains at least three `1`s$\}$
- $\{w \mid w$ contains the substring `0101` (i.e, $w = x\texttt{0101}y$ for some x and y)$\}$
- $\{w \mid w$ has length at least three and its third symbol is a `0`$\}$
- $\{w \mid w$ starts with `0` and has odd length, or starts with `1` and has even length$\}$
- $\{w \mid w$ doesn't contain the substring `110`$\}$

Problem 12:
Draw a DFA over an alphabet $\{\mathtt{a}, \mathtt{b}, \mathtt{@}, \mathtt{.}\}$ recognizing the language of strings of the form `user@website.domain`, where `user`, `website` and `domain` are nonempty strings over $\{\mathtt{a}, \mathtt{b}\}$ and `domain` has length 2 or 3.

Problem 13:
Draw a state diagram for a DFA over an alphabet of your choice that accepts exactly $f(n)$ strings of length n if

- $f(n) = n$
- $f(n) = n + 1$
- $f(n) = 3^n$
- $f(n) = n^2$
- $f(n)$ is a Tribonacci number.
  Tribonacci numbers are defined by the sequence $f(0) = 0$, $f(1) = 1$, $f(2) = 1$, and $f(n) = f(n-1) + f(n-2) + f(n-3)$ for $n \geq 3$
  Hint: Fibonacci numbers are given by the automaton prohibiting two letters "`a`" in a row.

Problem 14:
Draw a DFA recognizing the language of strings over $\{\texttt{0}, \texttt{1}\}$ in which `0` is the third digit from the end. Prove that any such DFA must have at least 8 states.

Problem 15:
Construct a DFA that accepts an binary integer if and only if it is divisible by six.
Strings are read from most to least significant digit. (that is, 18 will be read as 1,0,0,1,0)

Problem 16:
Construct a DFA that satisfies **??**, but reads digits in the opposite order.

Part 2: Regular languages

Definition 17:
We say a language is *regular* if it is recognized by some DFA.

Problem 18:
Draw a DFA over $\{A, B\}$ that accepts strings which do not start and end with the same letter.
Hint: Modify the DFA in **??**.

Problem 19:
Let L be a regular language over an alphabet Q.
Show that $Q^* - L$ is also regular.
Hint: $Q^* - L$ is the set of objects in Q^* but not in L. This is often called the *complement* of L.

Problem 20:
Draw a DFA over the alphabet $\{A, B\}$ that accepts strings which have even length and do not start and end with the same letter.

Problem 21:
Let L_1, L_2 be two regular languages over an alphabet Q.
Show that their union and intersection are also regular.

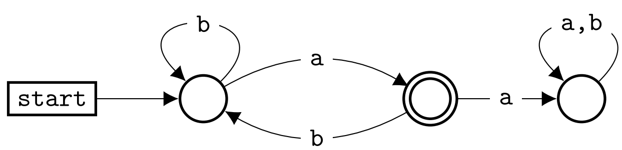
Theorem 22: Pumping Lemma
Let A be a regular language.
There then exists a number p, called the *pumping length*, so that any string $s \in A$ of length at least p may be divided into three pieces $s = xyz$ satisfying the following:

- $|y| > 0$ — *Hint:* In other words, the segment y is not the empty string.
- $|xy| \leq p$. — *Hint:* $|s|$ is the length of a string.
- $\forall i > 0$, $xy^iz \in A$ — *Hint:* y^i means that y is repeated i times. y^0 is the empty string.

When s is divided into xyz, either x or z may be the empty string, but y must not be empty.
Notice that without the first condition, this theorem is trivially true.

In english, the pumping lemma states that in any regular language, any string of sufficient length contains a substring that can be "pumped" (or repeated) to generate more strings in that language.

Problem 23:
Check that the pumping lemma holds with $p = 3$ for the following DFA.
Hint: This is the same DFA as in **??**. What kind of strings does it accept?



Problem 24:
How can we use the pumping lemma to show that a language is **not** regular?

Problem 25:
Prove the pumping lemma.
Hint: Look at the first cycle in the DFA you get while reading s.

Problem 26:
Show that the following languages are not regular:

A: $\{0^n1^n \mid n \in \mathbb{Z}_0^+\}$ over $\{0, 1\}$, which is the shorthand for the set $\{\varepsilon, 01, 0011, \dots\}$

B: The language ADD over the alphabet $\Sigma = \{0, 1, +, =\}$ where
ADD = { "x=y+z" | x, y, z are binary integers, and x is the sum of y and z}

C: The language of all palindromes over the english alphabet

Definition 27:
Let w be a string over an alphabet A.
If $a \in A$, $|w|_a$ is the number of times the letter a occurs in w.

For the following problems, we will use the alphabet $\{a, b\}$.

Problem 28:
Show that the language $L_p = \left\{ w \;\middle|\; p \text{ divides } |w|_a - |w|_b \right\}$ is regular for any prime p.

Problem 29:
Show that $L = \left\{ w \;\middle|\; |w|_a - |w|_b = \pm 1 \right\}$ is not regular.

Problem 30:
Prove that there are infinitely many primes.