$\Sigma = \{a, b, c\}$ or $\{0, 1\}$

the alphabet

A language is a set of strings taken

some alphabet.

For $\Sigma = \{0, 1\}$,

$\{ 0 1^n 0 \mid n \in \mathbb{N} \}$ is a language

$\mathbb{N} = \{0, 1, 2, 3, \ldots\}$

$\{ 0x \mid x \text{ is a string} \}$ is a languge

$\emptyset$ is a language

The biggest language is the set of all strings on Σ,

written Σ^* ← "the Kleene closure" of Σ.

$\{ 0x \mid x \in \Sigma^* \}$

A spoken language has an alphabet, and the language is the set of all allowable sentences.

$\Sigma = \{a, b, c, \ldots, z, \_\}$, english language $= \mathcal{E}$

then

$$\text{i have a dog} \in \mathcal{E}$$

$$\text{i hawe a dog} \notin \mathcal{E}$$

$$\text{i have i have i have} \notin \mathcal{E}$$

A programming language is all giant strings which make legal programs.

The interpreter acts as a validity checker

A simple machine to "accept" or "reject" strings.

Deterministic Finite Automaton (DFA)

no randomness
same input always gives the same result.

the machine has only finitely many <u>states</u>

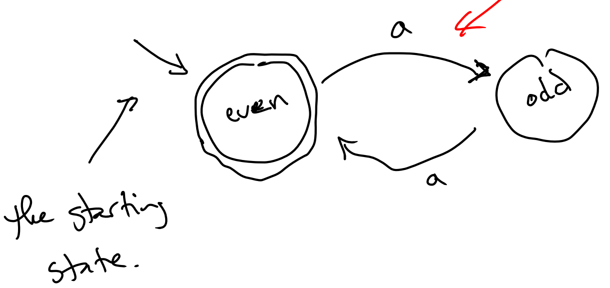
machine

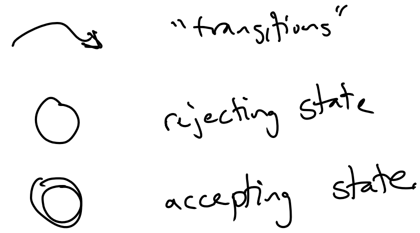
Ex1 An even-length machine on $\Sigma = \{a\}$

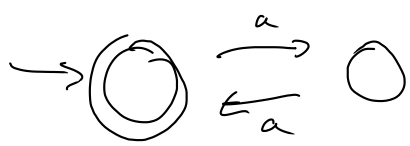
We'll make a DFA which reads the string and accepts if the length is even, rejects if the length is odd.

A DFA reads the string 1 letter at a time, L to R, and after each letter, maybe changes state.
At the end, it either accepts or rejects.

if I'm in the even state, and I read one a, then switch to odd state.







To evaluate a string, just follow the arrows.

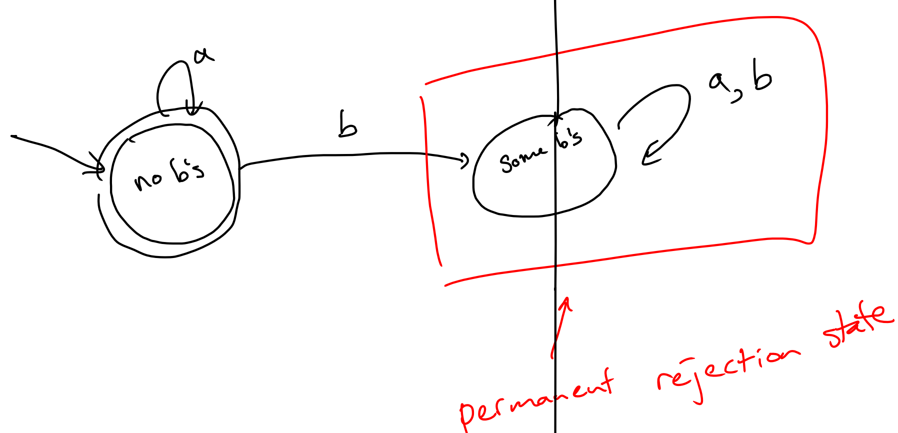
aaa is rejected

aaaa is accepted. etc.

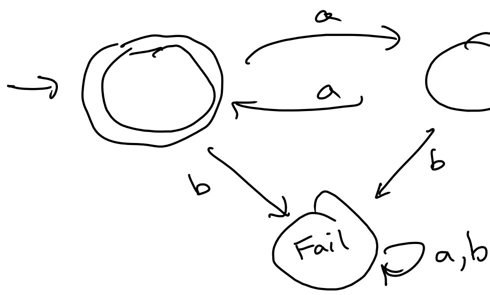
ε is accepted.

---

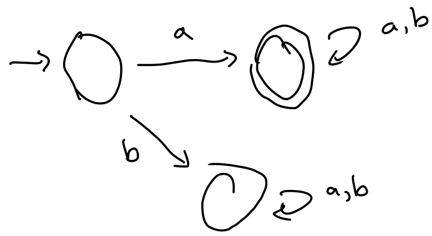
on $\Sigma = \{a, b\}$ make a DFA to accept any string with no b's.



<u>Ex</u> A DFA on $\Sigma = \{a, b\}$ which accepts any string with an even # of a's and no b's.



- Accepts any string starting with a $\Sigma = \{a,b\}$
- \- - - - - ending with a.



ending with a: