S. KIMMEL

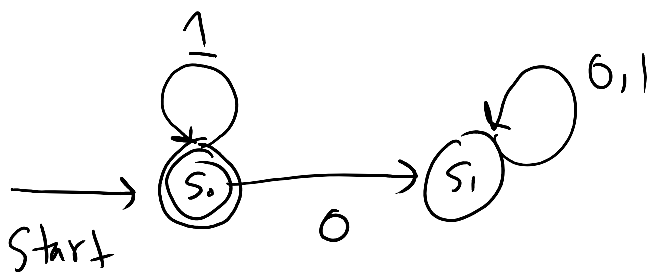
DFA's = Deterministic Finite Automata
= FSM with no output

$M = (S, I, f, s_0, F)$

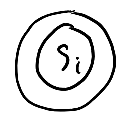
set of accepting states

* No output

ex



Double circle means



s_i is accepting state

Q: What strings in $\{0,1\}^*$ will this string accept?

A) Strings that start with 1

B) Strings that contain only 1's $\{1^n : n \in \mathbb{N}\}$

C) Strings that don't contain 0's

D) Strings that contain only 0's.

S. KIMMEL

Powerpoint...

Can every set of strings be identified by a DFA?

- No, only regular expressions

DFA's are weaker than Computers (turing machines), so

Can a computer identify any set of strings?

- No. Halting problem

Recall:

The next sentence is false
The previous sentence is true.

- Suppose there is a program

$$\text{Halt}(\underset{\substack{\uparrow \\ \text{Program}}}{P}, \underset{\substack{\uparrow \\ \text{input}}}{x}) = \begin{cases} 1 & \text{if } P(x) \text{ halts} \\ 0 & \text{if } P(x) \text{ gets stuck in } \infty \text{ loop} \end{cases}$$

- This would be great! Could check inputs before running them.

S.KIMMEL

We will use a proof by contradiction to show Halt will not run in finite time on all inputs.

For contradiction, suppose Halt does run in finite time on all inputs

Then we can create the program Z

```
Z(P):
  If Halt(P,P):
      Loop forever
  Else:
      end
```

- Case 1: Z(Z) loops forever, then Halt(Z,Z) = 0, doesn't loop forever X
- Case 2: Z(Z) doesn't loop forever, then Halt(Z,Z) = 1, loops forever. X

Either way, we get a contradiction!

So Halt will be incorrect or take infinite time on some inputs.