NFAs
NFAs

• An **NFA** is a
  • **N**ondeterministic
  • **F**inite
  • **A**utomaton

• Structurally similar to a DFA, but represents a fundamental shift in how we'll think about computation.
(Non)determinism

• A model of computation is **deterministic** if at every point in the computation, there is exactly one choice that can be made.

• The machine accepts if that series of choices leads to an accepting state.

• A model of computation is **nondeterministic** if the computing machine may have multiple decisions that it can make at one point.

• The machine accepts if **any** series of choices leads to an accepting state.
A Simple NFA

$q_0$ has two transitions defined on $1$!
A Simple NFA

0 1 0 1 1
A Simple NFA
A Simple NFA

- Start state: $q_0$
- Transitions:
  - $q_0 \rightarrow q_1$ on 1
  - $q_1 \rightarrow q_2$ on 1
  - $q_2$ is a self-loop on 0, 1
  - $q_1 \rightarrow q_3$ on 0
  - $q_3 \rightarrow q_3$ on 0, 1

Input sequence: 0 1 0 1 1
A Simple NFA

start

$q_0$

$1$ $0, 1$

$q_1$

$1$

$q_2$

$q_3$

$0$

$0, 1$

$0, 1$

$0, 1$

$0$ $1$ $1$ $1$
A Simple NFA
A Simple NFA

0 1 0 1 1
A Simple NFA

start

$q_0$ 1 $q_1$

$q_1$ 1 $q_2$

$q_2$

$q_3$

0, 1

0

0, 1

0, 1
A Simple NFA

\[ \begin{array}{c}
0, 1 \\
1
\end{array} \quad \begin{array}{c}
0, 1 \\
1
\end{array} \quad \begin{array}{c}
0, 1 \\
0
\end{array} \quad \begin{array}{c}
0, 1 \\
0
\end{array} \]
A Simple NFA

\begin{verbatim}
0 1 0 1 1
\end{verbatim}
A Simple NFA

start

$q_0$ 1 $q_1$

0, 1

$0, 1$

$q_1$ 1 $q_2$

$q_3$ 0 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$

$q_3$ 0, 1 $q_2$
A Simple NFA

- Start state: $q_0$
- Transitions:
  - From $q_0$: 1 to $q_1$, 0, 1 to $q_3$
  - From $q_1$: 1 to $q_2$
  - From $q_3$: 0 to $q_2$, 0, 1 to $q_3$

Input string: 0 1 0 1 1 1
A Simple NFA
A Simple NFA

The figure shows a nondeterministic finite automaton (NFA) with states $q_0$, $q_1$, $q_2$, and $q_3$. The transitions are labeled as follows:

- From $q_0$ to $q_1$ on input 1.
- From $q_1$ to $q_2$ on input 1.
- From $q_3$ to $q_3$ on input 0.
- From $q_3$ to $q_3$ on input 1.

The accepting state is $q_3$. The string 01011 is shown as an input sequence.
A Simple NFA

\[
\begin{array}{c}
\text{start} \\
\rightarrow q_0 \rightarrow q_1 \rightarrow q_2 \\
\downarrow 1 \downarrow 1 \\
q_1 \rightarrow q_3 \rightarrow q_2 \\
\downarrow 0, 1 \\
q_3 \\
\downarrow 0, 1 \\
\end{array}
\]

Input sequence: 0 1 0 1 1 1
A Simple NFA

\[
\begin{array}{c}
\text{start} \\
q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2 \\
q_0 \xrightarrow{0, 1} q_1 \xrightarrow{0} q_3 \xrightarrow{0, 1} q_2 \\
q_3 \xrightarrow{0, 1} q_3
\end{array}
\]
A Simple NFA
A Simple NFA

- Start state: $q_0$
- Transitions:
  - From $q_0$ to $q_1$: 1
  - From $q_0$ to $q_3$: 0, 1
  - From $q_1$ to $q_2$: 1
  - From $q_1$ to $q_3$: 0, 1
  - From $q_3$ to $q_2$: 0, 1

Input string: 0 1 0 1 1
A Simple NFA

Start state: $q_0$

Transitions:
- $q_0$ to $q_1$: on '1'
- $q_1$ to $q_2$: on '1'
- $q_1$ to $q_3$: on '0'
- $q_3$ to itself: on '0' and '1'

Input sequence: 0 1 0 1 1
A Simple NFA

0 1 0 1 1
A Simple NFA

0 1 0 1 1 1
A Simple NFA

- Start state: $q_0$
- Transitions:
  - $q_0 \xrightarrow{0, 1} q_1$
  - $q_1 \xrightarrow{1} q_2$
  - $q_1 \xrightarrow{0} q_3$
  - $q_3 \xrightarrow{0, 1} q_2$
  - $q_3 \xrightarrow{0, 1} q_3$

Input sequence: 0 1 0 1 1 1
A Simple NFA

\[ q_0 \xrightarrow{0, 1} q_1 \xrightarrow{1} q_2 \xrightarrow{0, 1} q_3 \xrightarrow{0, 1} q_2 \]
A Simple NFA

0 1 0 1 1
A Simple NFA

010111
A Simple NFA

\[ q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2 \]

\[ q_0 \xleftarrow{0, 1} q_1 \xleftarrow{0} q_3 \xleftarrow{0, 1} q_3 \xleftarrow{0, 1} q_3 \]

Input: 010111
A Simple NFA
A More Complex NFA
A More Complex NFA

If a NFA needs to make a transition when no transition exists, the automaton **dies** and that particular path does not accept.
A More Complex NFA
A More Complex NFA

start \rightarrow q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2

\text{0 1 0 1 1}
A More Complex NFA

Start \( q_0 \) on input 1 to \( q_1 \) on input 1, then loop to \( q_2 \).
A More Complex NFA
A More Complex NFA

start → $q_0$ → 1 → $q_1$ → 1 → $q_2$

0, 1
A More Complex NFA

The diagram shows a non-deterministic finite automaton (NFA) with three states: $q_0$, $q_1$, and $q_2$. The transitions are as follows:

- From $q_0$, on input 1, it transitions to $q_1$.
- From $q_1$, on input 1, it stays in $q_1$.
- From $q_1$, on input 0 or 1, it transitions back to $q_0$.
- $q_2$ is a final state.

The input string is 010111.
A More Complex NFA

Oh no! There's no transition defined!
A More Complex NFA
A More Complex NFA

start $q_0$ 1 $q_1$ 1 $q_2$

0, 1

0 1 0 1 1
A More Complex NFA
A More Complex NFA

start → \( q_0 \) 1 \( q_1 \) 1 \( q_2 \)
0, 1

0 1 0 1 1 1
A More Complex NFA

![Diagram of a more complex NFA with states q₀, q₁, and q₂, and transitions labeled for 0, 1, and 1 respectively.]

Input sequence: 0 1 0 1 1 1
A More Complex NFA

- Start state: $q_0$
- States: $q_0, q_1, q_2$
- Transitions:
  - $q_0 \xrightarrow{1} q_1$
  - $q_1 \xrightarrow{1} q_2$
  - $q_1 \xrightarrow{0, 1} q_1$

Input string: 0 1 0 1 1 1
A More Complex NFA
A More Complex NFA

1. Start at state $q_0$.
2. Move to state $q_1$ on input 1.
3. Move to state $q_2$ on input 1.

Input sequence: 0 1 0 1 1 1
A More Complex NFA

\[ q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2 \]

\[ \{0, 1\} \xrightarrow{0, 1} q_0 \]

\[ \{0, 1\}^{*} \]

\[ 010111 \]
A More Complex NFA
A More Complex NFA

Start → $q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_2$
As with DFAs, the language of an NFA $N$ is the set of strings that $N$ accepts:

$$\mathcal{L}(N) = \{ w \in \Sigma^* | N \text{ accepts } w \}.$$ 

What is the language of the NFA shown above?

A. $\{01011\}$
B. $\{w \in \{0, 1\}^* | w \text{ contains at least two } 1\text{s}\}$
C. $\{w \in \{0, 1\}^* | w \text{ ends with } 1\}$
D. $\{w \in \{0, 1\}^* | w \text{ ends with } 11\}$
E. $\{w \in \{0, 1\}^* | w \text{ ends with } 111\}$
F. None of these, or two or more of these.

Answer at PollEv.com/cs103 or text CS103 to 22333 once to join, then A, ..., or F.
NFA Acceptance

• An NFA $N$ accepts a string $w$ if there is some series of choices that lead to an accepting state.
• Consequently, an NFA $N$ rejects a string $w$ if no possible series of choices lead it into an accepting state.
• It's easier to show that an NFA does accept something than to show that it doesn't.
ε-Transitions

• NFAs have a special type of transition called the ε-transition.
• An NFA may follow any number of ε-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the \textbf{ε-transition}.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
\(\varepsilon\)-Transitions

- NFAs have a special type of transition called the **\(\varepsilon\)-transition**.
- An NFA may follow any number of \(\varepsilon\)-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the \textit{ε-transition}.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
**ε-Transitions**

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
\( \varepsilon \)-Transitions

- NFAs have a special type of transition called the \textbf{\( \varepsilon \)-transition}.
- An NFA may follow any number of \( \varepsilon \)-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
£-Transitions

- NFAs have a special type of transition called the £-transition.
- An NFA may follow any number of £-transitions at any time without consuming any input.
\( \varepsilon \text{-Transitions} \)

- NFAs have a special type of transition called the **\( \varepsilon \)-transition**.
- An NFA may follow any number of \( \varepsilon \)-transitions at any time without consuming any input.
**ε-Transitions**

- NFAs have a special type of transition called the *ε-transition*.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
**ε-Transitions**

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
**ε-Transitions**

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
**ε-Transitions**

- NFAs have a special type of transition called the *ε-transition*.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
**ε-Transitions**

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
\(\varepsilon\)-Transitions

- NFAs have a special type of transition called the \(\varepsilon\)-transition.
- An NFA may follow any number of \(\varepsilon\)-transitions at any time without consuming any input.
**ε-Transitions**

- NFAs have a special type of transition called the **ε-transition**.
- An NFA may follow any number of ε-transitions at any time without consuming any input.
ε-Transitions

- NFAs have a special type of transition called the **ε-transition**.
- An NFA *may* follow any number of ε-transitions at any time without consuming any input.
- NFAs are not *required* to follow ε-transitions every time one is available. It's simply another option at the machine's disposal (stay or move).
Suppose we run the above NFA on the string 10110. How many of the following statements are true?

- There is at least one computation that finishes in an accepting state.
- There is at least one computation that finishes in a rejecting state.
- There is at least one computation that dies.
- This NFA accepts 10110.
- This NFA rejects 10110.

Answer at PollEv.com/cs103 or text CS103 to 22333 once to join, then a number.