NFA/DFA Equivalence
CSCI 338
DFA vs NFA

NFA

DFA
DFA vs NFA

Claim: Every NFA has an equivalent DFA.

Proof Approach:
How did we keep track of our location in a DFA?
DFA vs NFA

Claim: Every NFA has an equivalent DFA.

Proof Approach:
How did we keep track of our location in an NFA?

\[ \omega = 101 \]
DFA vs NFA

Claim: Every NFA has an equivalent DFA.

Proof Approach:

How did we keep track of our location in an NFA?
Set of all states we could possibly be in.
DFA vs NFA

Claim: Every NFA has an equivalent DFA.

Proof Approach:
How did we keep track of our location in an NFA?
Set of all states we could possibly be in.
DFA vs NFA

Claim: Every NFA has an equivalent DFA.

Proof Approach:
How did we keep track of our location in an NFA?
Set of all states we could possibly be in.

\[ \omega = 101 \]
DFA vs NFA

Claim: Every NFA has an equivalent DFA.

Proof Approach:

How did we keep track of our location in an NFA?
Set of all states we could possibly be in.

What is the set of all possible locations?
Power set of states! (set of all subsets)

\[ \omega = 101 \]
DFA vs NFA

NFA

DFA

DFA states = Power set of NFA states.
DFA vs NFA

NFA

DFA states = Power set of NFA states.
Start state = ?
DFA vs NFA

NFA

DFA

DFA states = Power set of NFA states.
Start state = NFA’s start state.
DFA vs NFA

**NFA**

- States: \( S_1, S_2, S_3 \)
- Transitions:
  - \( S_1 \) on '0' goes to \( S_3 \)
  - \( S_2 \) on '0' goes to \( S_2 \)
  - \( S_2 \) on '1' goes to \( S_3 \)
  - \( S_3 \) on '0' goes to \( S_2 \)

**DFA**

- States: \( \emptyset, \{S_1\}, \{S_2\}, \{S_1, S_2\}, \{S_3\}, \{S_1, S_3\}, \{S_2, S_3\}, \{S_1, S_2, S_3\} \)

DFA states = Power set of NFA states.
Start state = NFA’s start state.
Accept states = ?
DFA vs NFA

NFA

DFA

DFA states = Power set of NFA states.
Start state = NFA’s start state.
Accept states = Any state that includes accept state from NFA.
DFA vs NFA

NFA

\[
\begin{align*}
S_1 &\xrightarrow{0,1} S_3 \\
S_2 &\xrightarrow{0} S_3
\end{align*}
\]

DFA

\[
\begin{align*}
\emptyset &\rightarrow \{S_1\} \\
\{S_3\} &\rightarrow \{S_1, S_3\} \\
\{S_2\} &\rightarrow \{S_2, S_3\} \\
\{S_1, S_2\} &\rightarrow \{S_1, S_2, S_3\}
\end{align*}
\]

Where should transition out of \(\{S_1\}\) with character 1 go?
Where should transition out of \( \{S_1\} \) with character 1 go? Wherever \( S_1 \) goes with 1 in the NFA.
DFA vs NFA

NFA

Where should transition out of \{S_1\} with character 1 go?

Wherever \(S_1\) goes with 1 in the NFA.
Where should transition out of $\{S_1\}$ with character 0 go?
Where should transition out of \{S_1\} with character 0 go?
If transition is not handled by NFA, send it to \emptyset (junk state).
Where should transition out of \( \{S_2\} \) with character 0 go?
Where should transition out of $\{S_2\}$ with character 0 go?

NFA could stay in $S_2$ or go to $S_3$, so $\{S_2, S_3\}$
Where should transition out of \{S_1, S_3\} with character 1 go?
Where should transition out of $\{S_1, S_3\}$ with character 1 go?
DFA vs NFA

Where should transition out of \{S_1, S_3\} with character 0 go?
Where should transition out of \( \{S_1, S_3\} \) with character 0 go?
DFA vs NFA

Rule?

DFA state transitions to DFA state consisting of all states it’s NFA states transition to.
DFA vs NFA

Rule? For each DFA state \( R \) and \( e \in \Sigma \),
\[
\text{transition}(R, e) = \{ q \in \text{NFA} : q \in \text{transition}(r, e) \text{ for some } r \in R \}.
\]
DFA vs NFA

NFA

DFA
What about $\varepsilon$-transitions?

Define extension of DFA state $R$:

$E(R) = \{q \in \text{NFA}: q \text{ reachable from } r \in R \text{ with } \geq 0 \varepsilon\text{-transitions}\}$
$E(R) = \{q \in \text{NFA}: \text{q reachable from } r \in R \text{ with } \geq 0 \varepsilon\text{-transitions}\}$

Example: $E(\{S_2, S_3\}) = \?$
DFA vs NFA

**NFA**

- States: $S_1$, $S_2$, $S_3$
- Transitions:
  - $S_1$ to $S_2$: 0
  - $S_1$ to $S_3$: $\varepsilon$
  - $S_2$ to $S_3$: 0, 1

**DFA**

- States: $\emptyset$, $\{S_1\}$, $\{S_2\}$, $\{S_1, S_2\}$, $\{S_3\}$, $\{S_1, S_3\}$, $\{S_2, S_3\}$, $\{S_1, S_2, S_3\}$

$E(R) = \{q \in \text{NFA}: q \text{ reachable from } r \in R \text{ with } \geq 0 \varepsilon\text{-transitions}\}$

Example:
- $E(\{S_2, S_3\}) = \{S_2, S_3\}$
- $E(\{S_1, S_2\}) = \ ?$
\[ E(R) = \{ q \in \text{NFA}: q \text{ reachable from } r \in R \text{ with } \geq 0 \varepsilon-\text{transitions} \} \]

Example:
\[ E(\{S_2, S_3\}) = \{S_2, S_3\} \]
\[ E(\{S_1, S_2\}) = \{S_1, S_2, S_3\} \]
DFA vs NFA

NFA

Make start state = ?

DFA
DFA vs NFA

NFA

```
0
1
ε
0,1
```

```
S_1
S_2
S_3
```

```
0
0,1
```

DFA

```
∅
{S_1}
{S_2}
{S_1, S_2}
```

```
{S_3}
{S_1, S_3}
{S_2, S_3}
{S_1, S_2, S_3}
```

Make start state = $E(\{S_1\}) = \{S_1, S_3\}$
Make transitions:
\[
E(\text{transition}(r, e)) = \{ q \in \text{NFA}: q \in \text{transition}(r, e) \text{ for some } r \in R \}
\]
DFA vs NFA

NFA

DFA

Make transitions:

\[
E(\text{transition}(r, e)) = \{ q \in \text{NFA} : q \in \text{transition}(r, e) \text{ for some } r \in R \}
\]
Make transitions:

\[
\text{transition}(R, e) = \{ q \in \text{NFA}: q \in \text{transition}(r, e) \text{ for some } r \in R \}
\]
DFA vs NFA

NFA

DFA

Make transitions:
transition\((R, e) = \{q \in \text{NFA}: q \in \text{transition}(r, e) \text{ for some } r \in R\} \)

\[ E(\text{transition}(r, e)) \]
DFA vs NFA

**NFA**

- States: $s_1, s_2, s_3$
- Transitions:
  - $s_2 \xrightarrow{0,1} s_3$
  - $s_1 \xrightarrow{1} s_3$
  - $s_1 \xrightarrow{\varepsilon} s_2$

**DFA**

- States: $\emptyset, \{S_1\}, \{S_2\}, \{S_1, S_2\}, \{S_1, S_2, S_3\}$
- Transitions:
  - $\emptyset \xrightarrow{0} \{S_1\}$
  - $\{S_1\} \xrightarrow{0} \{S_1, S_3\}$
  - $\{S_1, S_3\} \xrightarrow{0} \{S_2, S_3\}$
  - $\{S_2, S_3\} \xrightarrow{0} \{S_1, S_2, S_3\}$

Make transitions:

$$\text{transition}(R, e) = \{q \in \text{NFA}: q \in \text{transition}(r, e) \text{ for some } r \in R\}$$

$$E(\text{transition}(r, e))$$
DFA vs NFA

NFA

Finite set of states? ✓
Transition function for every state/character pair? ✓
Single start state? ✓

DFA

Finite Alphabet? ✓
Set of accept states? ✓
DFA vs NFA

NFA

DFA

Equivalent?
Suppose $w$ accepted by NFA.

DFA vs NFA

NFA

DFA

Equivalent?
Suppose $w$ accepted by NFA. At each step of its processing, DFA will be in state that corresponds to all possible NFA states.
DFA vs NFA

NFA

Suppose $w$ accepted by NFA. At each step of its processing, DFA will be in state that corresponds to all possible NFA states. If NFA ends on accept state, corresponding DFA state will accept too.
DFA vs NFA

Suppose $w$ accepted by DFA.
DFA vs NFA

NFA

DFA

Equivalent?

Suppose $w$ accepted by DFA. At each step of its processing, DFA will be in state that corresponds to all possible NFA states.
Suppose $w$ accepted by DFA. At each step of its processing, DFA will be in state that corresponds to all possible NFA states. If DFA ends on accept state, it includes an NFA accept state.
Suppose \( \omega \) is accepted by DFAs at each step of its processing, DFA will be in state that corresponds to all possible NFA states. If DFA ends on accept state, it includes an NFA accept state.

Anything that can be done with NFAs can be done with DFAs!
A language is called a **regular language** if some DFA or **NFA** recognizes it.

How do you prove a language is regular?
Make a DFA or **NFA** that recognizes it.