

BU CS 332 – Theory of Computation

Lecture 4: Assignment Project Exam Help

- More on N
 - NFAs vs. DFAs
 - Closure Properties
- Reading:
pser Ch 1.1-1.2
- <https://eduassistpro.github.io/>
Add WeChat edu_assist_pro

Mark Bun

February 3, 2021

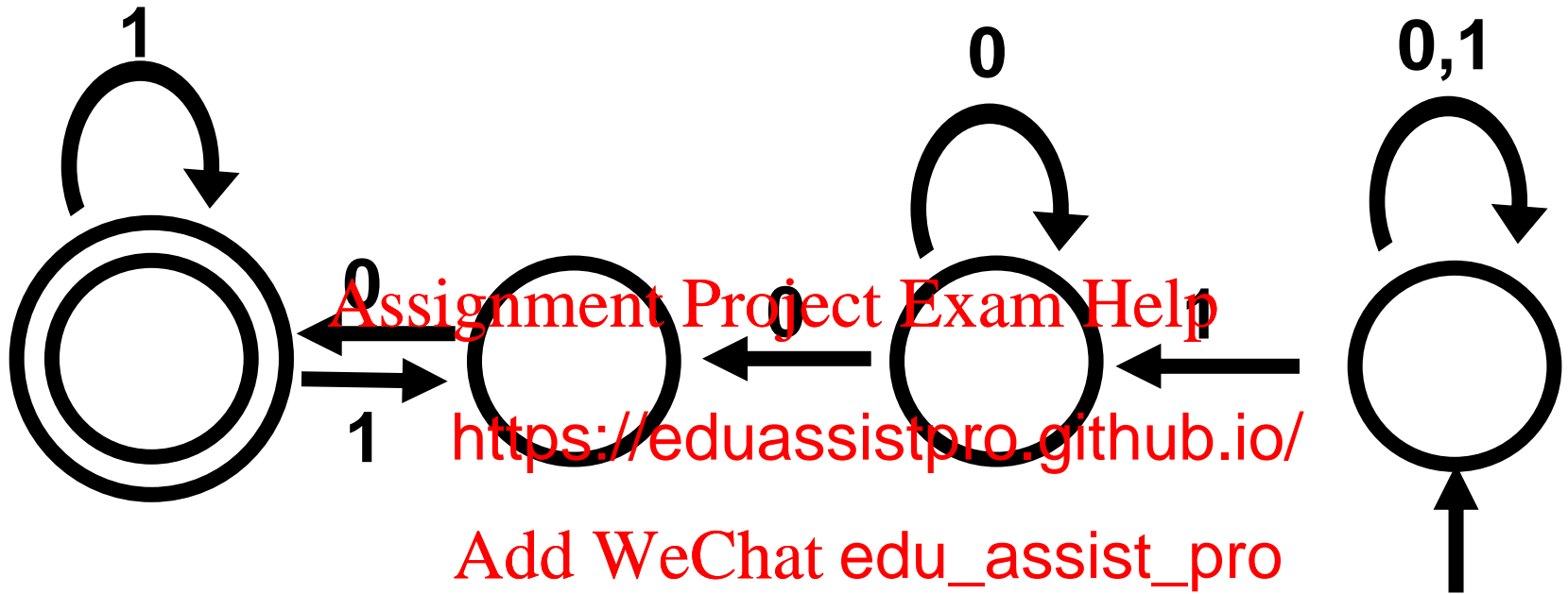
Last Time

- Deterministic Finite Automata (DFAs)
 - Informal description: State diagram
 - Formal description: What are they?
 - Formal description: How do they compute?
- A language is regular if and only if it is accepted by a DFA
- Intro to Nondeterministic FA

Assignment Project Exam Help

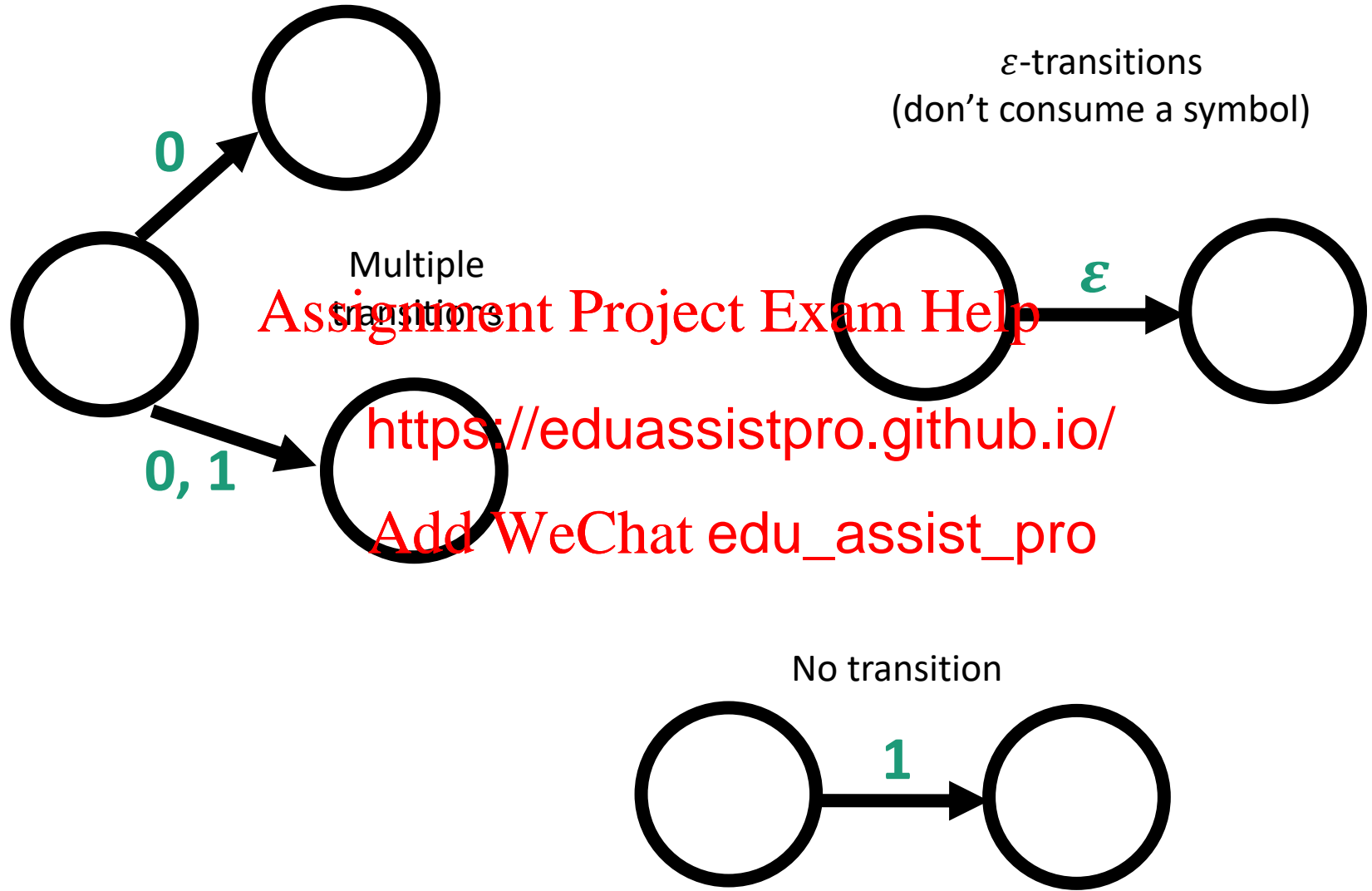
Add WeChat edu_assist_pro

Nondeterminism

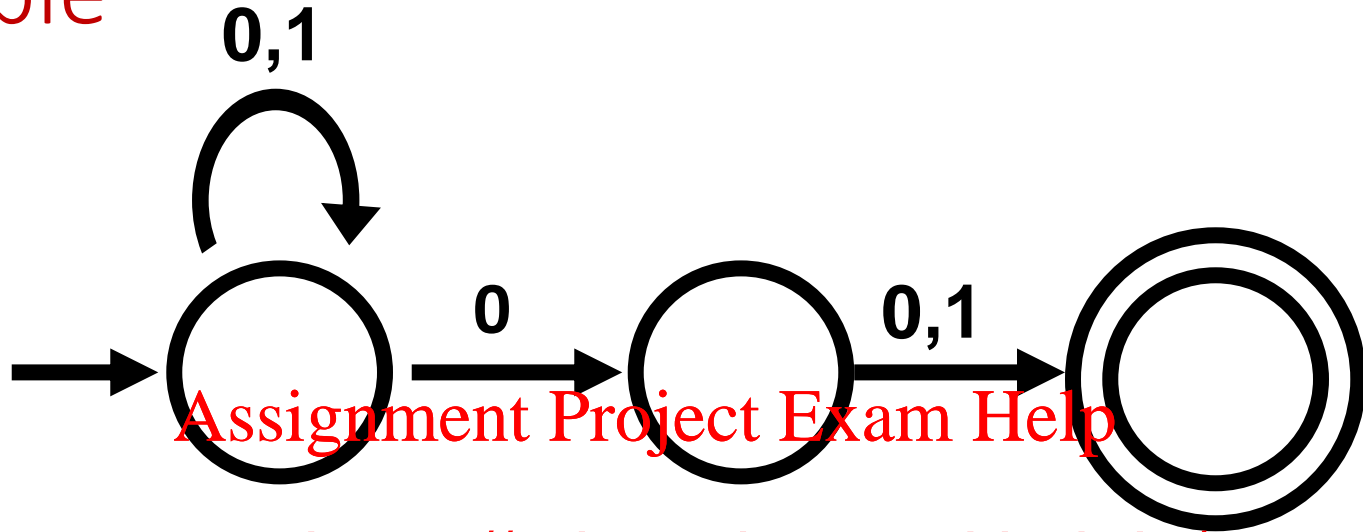


A Nondeterministic Finite Automaton (NFA) accepts if there *exists* a way to make it reach an accept state.

Some special transitions



Example



Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

$L(N) =$

- a) $\{w \mid w \text{ contains } 001\}$
- b) $\{w \mid \text{the second to last symbol of } w \text{ is } 0\}$
- c) $\{w \mid w \text{ starts with } 00 \text{ or } 01\}$
- d) $\{w \mid w \text{ ends with } 001\}$

Formal Definition of a NFA

An **NFA** is a 5-tuple $M = (Q, \Sigma, \delta, q_0, F)$

Q is the set of states

Σ is the alphabet

$\delta: Q \times \Sigma_\epsilon$ is the transition function

$q_0 \in Q$ is the start state

$F \subseteq Q$ is the set of acc

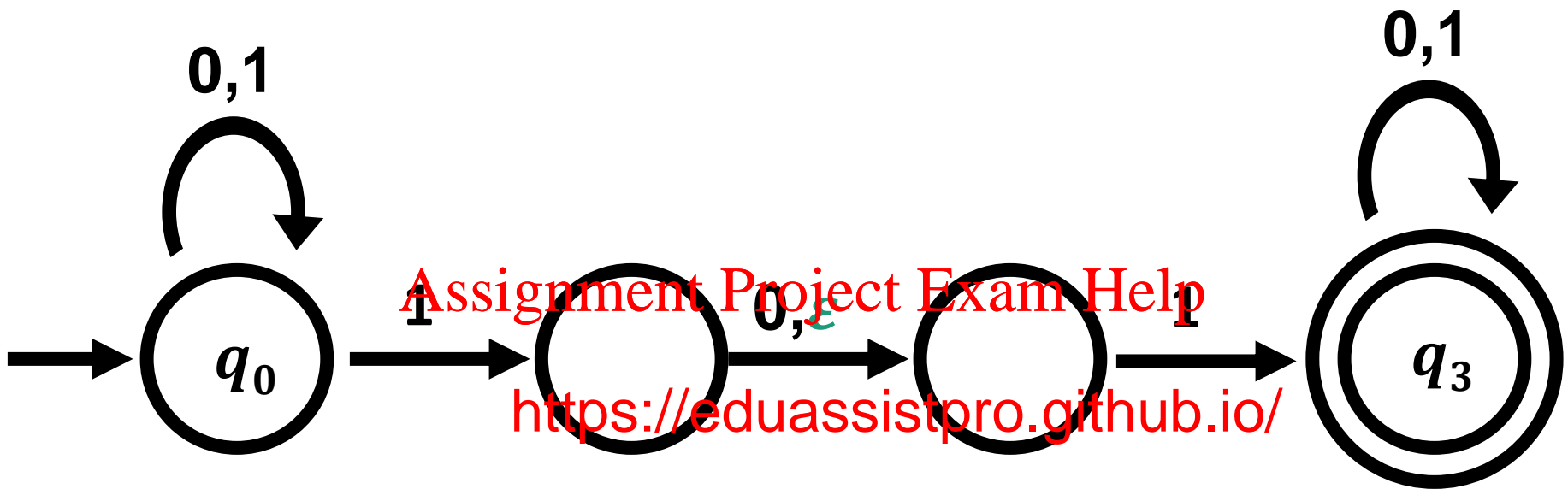
Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

M **accepts** a string w if **there exists** a path from q_0 to an accept state that can be followed by reading w .

Example



$$N = (Q, \Sigma, \delta, q_0, F)$$

$$Q = \{q_0, q_1, q_2, q_3\}$$

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

$$F = \{q_3\}$$

$$\delta(q_0, 1) =$$

$$\delta(q_1, \epsilon) =$$

$$\delta(q_2, 0) =$$

$$\delta(q_3, 0, 1) =$$

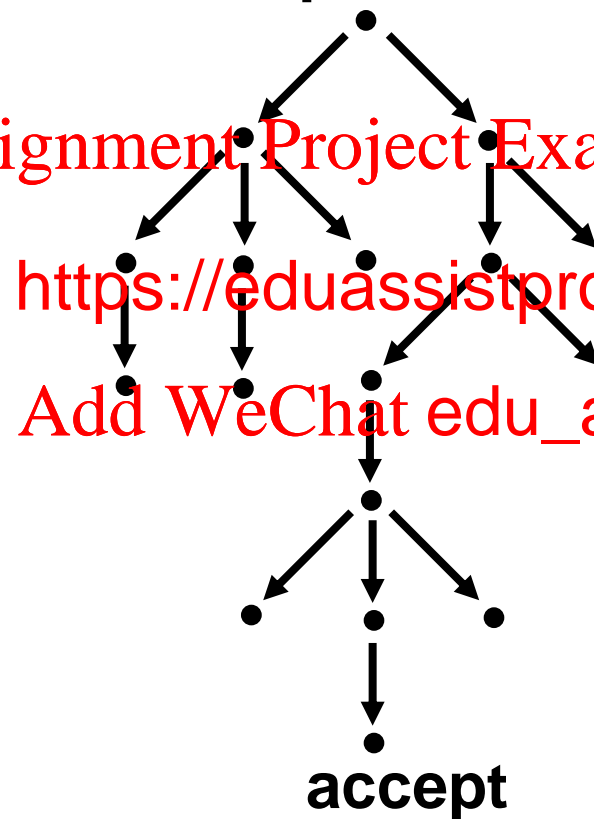
Nondeterminism

Deterministic Computation



accept or reject

Nondeterministic Computation



Ways to think about nondeterminism

- (restricted) parallel computation
- tree of possible computations
- guessing and verifying the “right” choice

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Why study NFAs?

- Not really a realistic model of computation: Real computing devices can't really try many possibilities in parallel

Assignment Project Exam Help

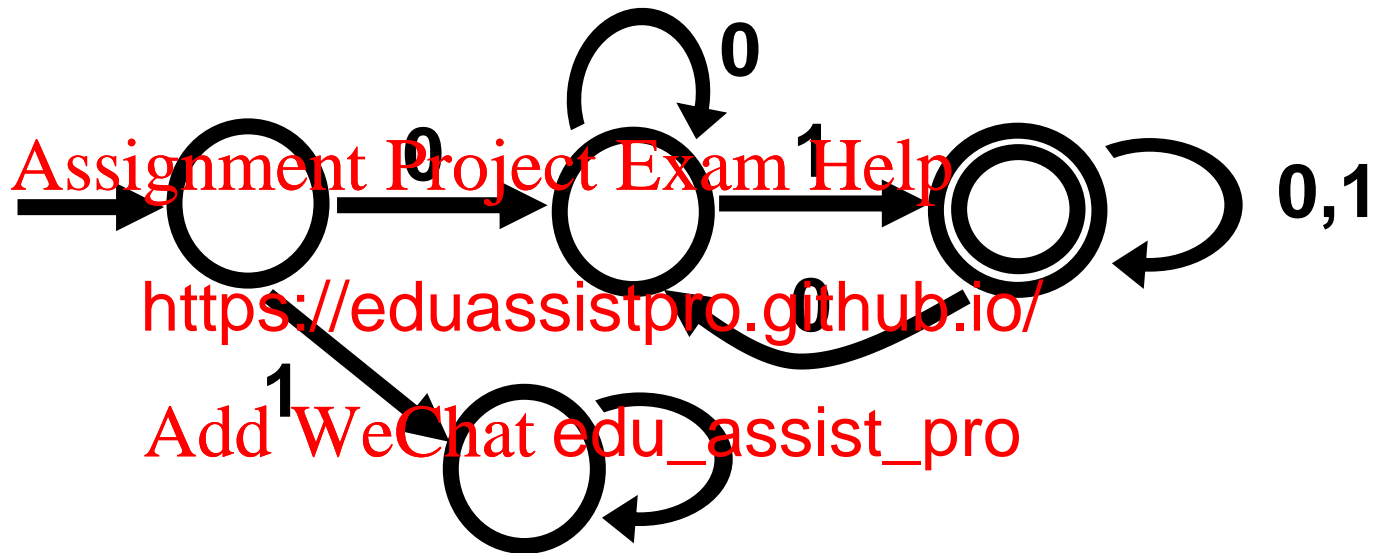
But:

<https://eduassistpro.github.io/>

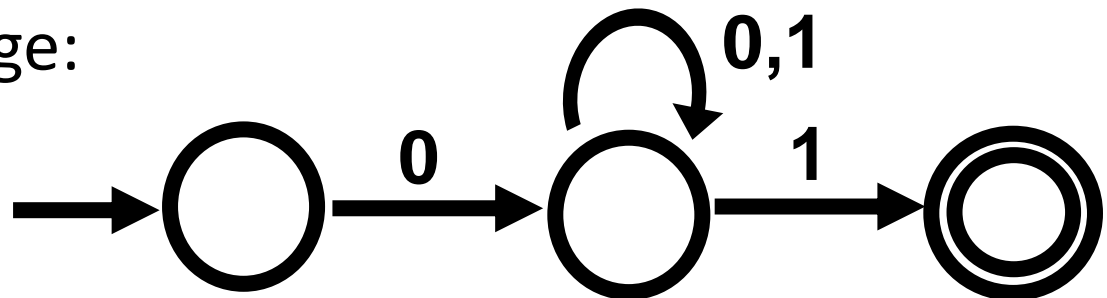
- Useful tool for understanding DFAs/regular languages
- NFAs can be simpler than DFAs
- Lets us study “nondeterminism” as a resource (cf. P vs. NP)

NFAs can be simpler than DFAs

A DFA that recognizes the language
 $\{w \mid w \text{ starts with } 0 \text{ and ends with } 1\}$:



An NFA for this language:



Equivalence of NFAs and DFAs

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Equivalence of NFAs and DFAs

Every DFA *is* an NFA, so NFAs are *at least* as powerful as DFAs

Assignment Project Exam Help

Theorem: For every language L , there exists a DFA M such that $L(M) = L$

<https://eduassistpro.github.io/>

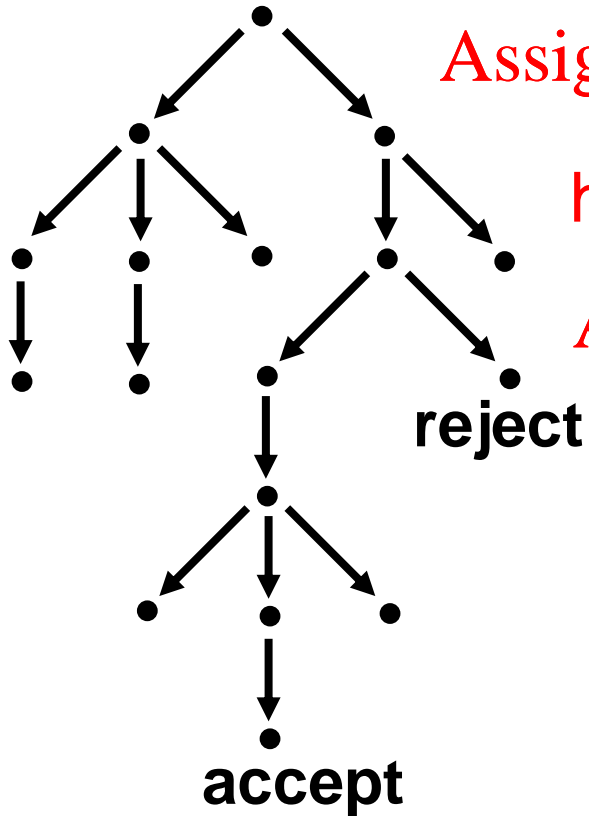
Add WeChat edu_assist_pro

Corollary: A language is regular if and only if it is recognized by an NFA

Equivalence of NFAs and DFAs (Proof)

Let $N = (Q, \Sigma, \delta, q_0, F)$ be an NFA

Goal: Construct DFA $M = (Q', \Sigma, \delta', q_0', F')$ recognizing $L(N)$



Assignment Project Exam Help

threads of N in

<https://eduassistpro.github.io/>ing the set of

states w

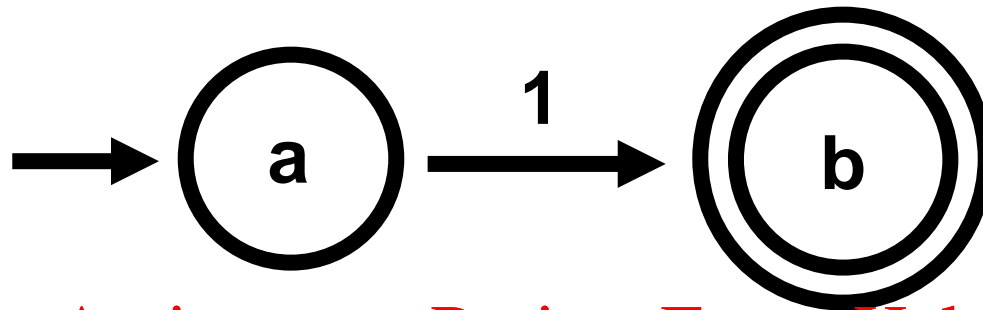
reads are.

Add WeChat edu_assist_pro

Formally: $Q' = P(Q)$

“The Subset Construction”

NFA -> DFA Example



Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Subset Construction (Formally, first attempt)

Input: NFA $N = (Q, \Sigma, \delta, q_0, F)$

Output: DFA $M = (Q', \Sigma, \delta', q_0', F')$

Q' **Assignment Project Exam Help**

$\delta' : Q' \times \Sigma \rightarrow Q'$ <https://eduassistpro.github.io/>

$\delta'(R, \sigma) =$ **Add WeChat edu_assist_pro** $\subseteq Q$ and $\sigma \in \Sigma$.

$q_0' =$

$F' =$

Subset Construction (Formally, for real)

Input: NFA $N = (Q, \Sigma, \delta, q_0, F)$

Output: DFA $M = (Q', \Sigma, \delta', q_0', F')$

$Q' = P(Q)$ **Assignment Project Exam Help**

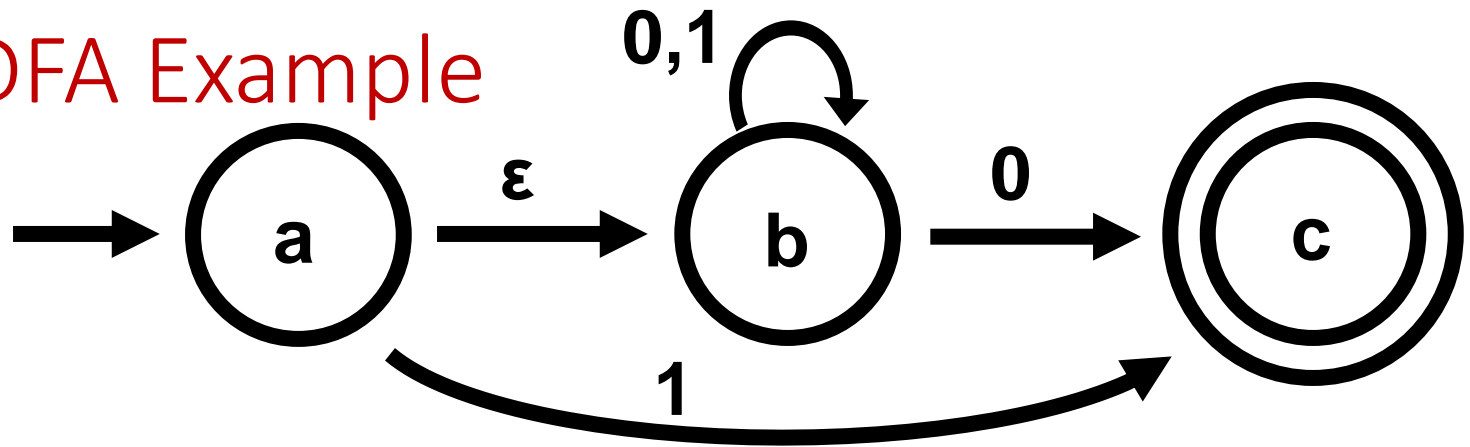
$\delta' : Q' \times \Sigma \rightarrow Q'$ <https://eduassistpro.github.io/>

$\delta'(R, \sigma) = \bigcup_{r \in R} \delta(r, \sigma) \subseteq Q$ and $\sigma \in \Sigma$.
Add WeChat edu_assist_pro

$q_0' = \{q_0\}$

$F' = \{ R \in Q' \mid R \text{ contains some accept state of } N \}$

NFA -> DFA Example



Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Proving the Construction Works

Claim: For every string w , running M on w leads to state

$\{q \in Q \mid \text{There exists a computation path of } q\}$

<https://eduassistpro.github.io/>

Proof idea: By induction on $|w|$

Historical Note

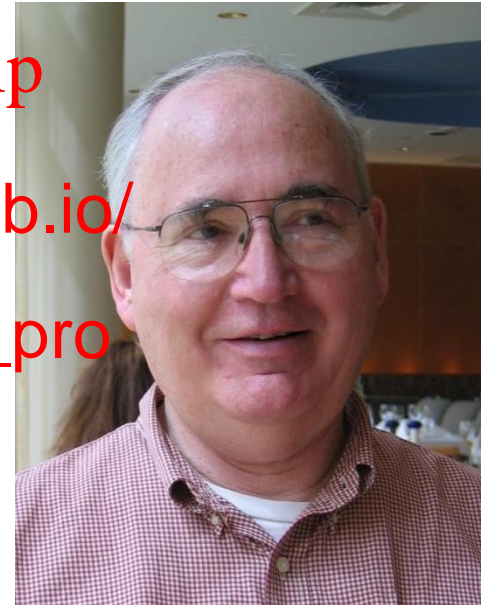
Subset Construction introduced in Rabin & Scott's 1959 paper "Finite Automata and their Decision Problems"



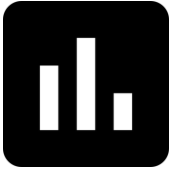
1976 ACM Turing Award citation
Assignment Project Exam Help

For t and
Thei <https://eduassistpro.github.io/>
the idea of nondetermi
which has proved to be
valuable concept. Their
classic paper has been a continuous source
of inspiration for subsequent work in this
field.

Add WeChat edu_assist_pro



NFA \rightarrow DFA: The Catch



If N is an NFA with s states, how many states does the DFA obtained using the subset construction have?

Assignment Project Exam Help

a) s

<https://eduassistpro.github.io/>

b) s^2

Add WeChat edu_assist_pro

c) 2^s

d) None of the above

Is this construction the best we can do?

Subset construction converts an n state NFA into a 2^n -state DFA

Could there be a construction that always produces, say, an n^2 -state DFA?

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Theorem: For every $n \geq 1$, there exists L_n such that

1. There is an $(n + 1)$ -state NFA recognizing L_n .
2. There is no DFA recognizing L_n with fewer than 2^n states.

Conclusion: For finite automata, nondeterminism provides an exponential savings over determinism (in the worst case).

Closures Assignment Project Exam Help
<https://eduassistpro.github.io/>
Add WeChat edu_assist_pro

An Analogy

In algebra, we try to identify operations which are common to many different mathematical structures

Example: The integers $\mathbb{Z} = \{\dots - 2, -1, 0, 1, 2, \dots\}$ are **closed** under

- Addition: $x +$
- Multiplication: $x \times y$
- Negation: $-x$
- ...but **NOT** Division: x / y

We'd like to investigate similar closure properties of the **class of regular languages**

Regular operations on languages

Let $A, B \subseteq \Sigma^*$ be languages. Define

Union: $A \cup B = \{w \mid w \in A \text{ or } w \in B\}$

Assignment Project Exam Help

Concatenation: $A \cdot B = \{w \mid w = uv, u \in A, v \in B\}$

Add WeChat edu_assist_pro

Star: $A^* =$

Other operations

Let $A, B \subseteq \Sigma^*$ be languages. Define

Complement: $\bar{A} = \{w \mid w \notin A\}$

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Intersection: $A \cap B = \{w \mid w \in A \text{ and } w \in B\}$

Add WeChat edu_assist_pro

Reverse: $A^R = \{w \mid w^R \in A\}$

Closure properties of the regular languages

Theorem: The class of regular languages is **closed** under all three regular operations (union, concatenation, star), as well as under complement, intersection, and reverse.

Assignment Project Exam Help

<https://eduassistpro.github.io/>

i.e., if A and B are regular, app f these
operations yields a regular lang

Add WeChat edu_assist_pro

Assignment Project Exam Help
Proving properties

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Complement

Complement: $\bar{A} = \{ w \mid w \notin A \}$

Theorem: If A is regular, then \bar{A} is also regular

Proof idea:

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Complement, Formally



Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA recognizing a language A . Which of the following represents a DFA recognizing \bar{A} ?

- a) $(F, \Sigma, \delta, q_0, Q)$
- b) $(Q, \Sigma, \delta, q_0, Q \setminus F)$ the set of states in Q that are not in F
- c) $(Q, \Sigma, \delta', q_0, F)$ where $\delta'(q, s) = q$ such that
- d) None of the above

Closure under Concatenation

Concatenation: $A \circ B = \{ xy \mid x \in A, y \in B \}$

Theorem. If A and B are regular, $A \circ B$ is also regular.

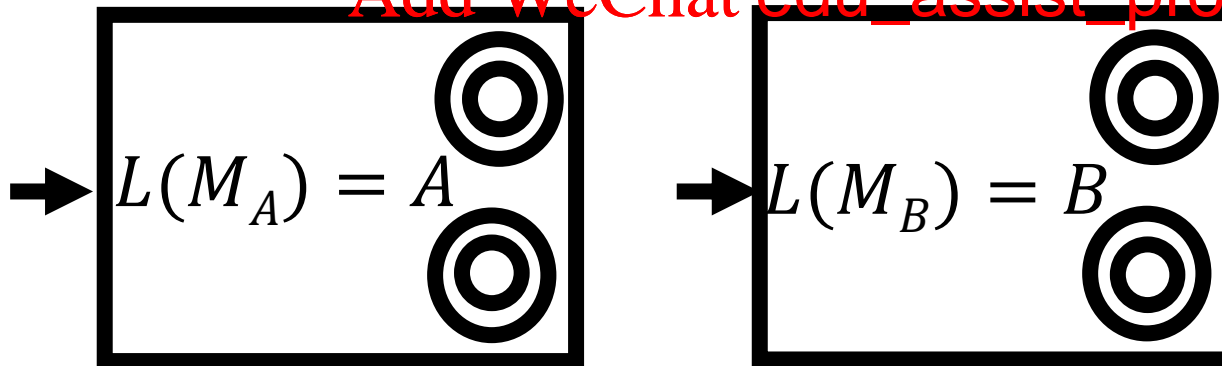
Proof idea: Given DFAs M_A and M_B , construct NFA by

- Connecting all acc
- Make all states in

rt state in M_B .

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro



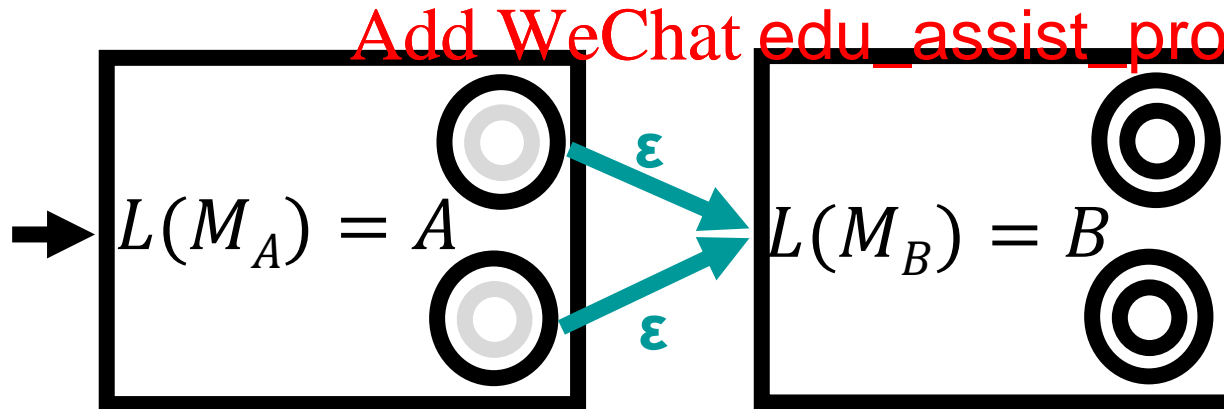
Closure under Concatenation

Concatenation: $A \circ B = \{ xy \mid x \in A, y \in B \}$

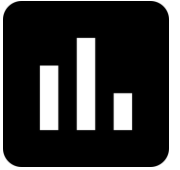
Theorem. If A and B are regular, $A \circ B$ is also regular.

Proof idea: Given DFAs M_A and M_B , construct NFA by

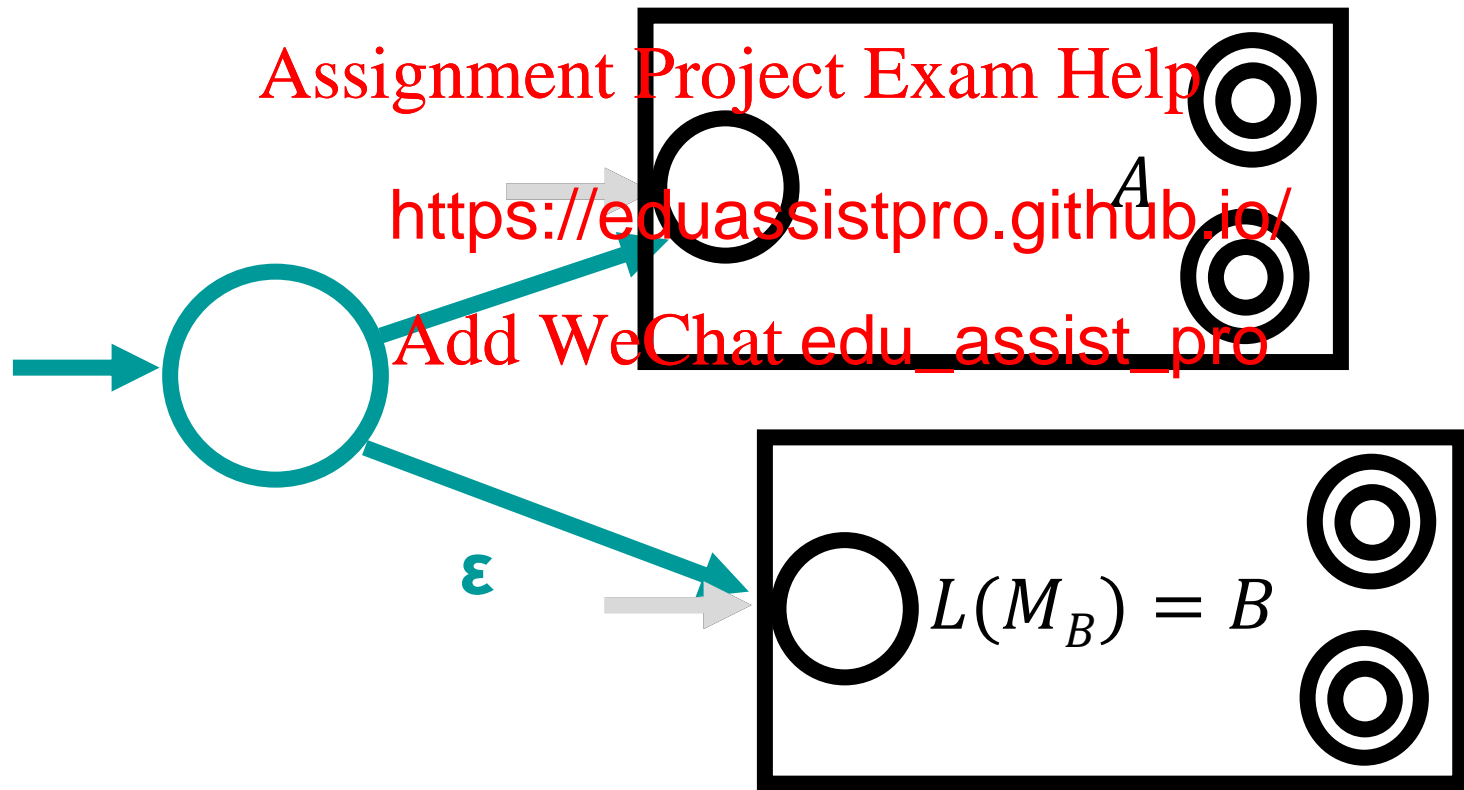
- Connecting all accept states in M_A to the start state in M_B .
- Make all states in M_B accepting.



A Mystery Construction



Given DFAs M_A recognizing A and M_B recognizing B , what does the following NFA recognize?



Closure under Star

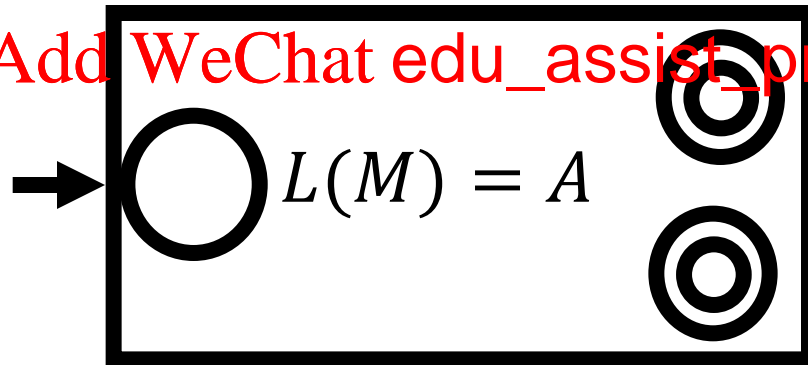
Star: $A^* = \{ a_1 a_2 \dots a_n \mid n \geq 0 \text{ and } a_i \in A \}$

Theorem. If A is regular, A^* is also regular.

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro



Closure under Star

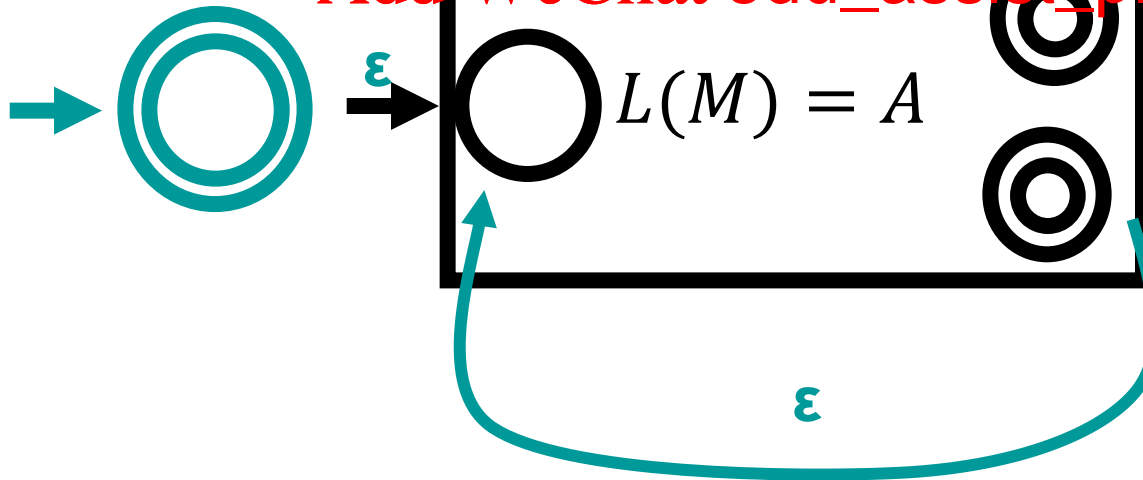
Star: $A^* = \{ a_1 a_2 \dots a_n \mid n \geq 0 \text{ and } a_i \in A \}$

Theorem. If A is regular, A^* is also regular.

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro



On proving your own closure properties

You'll have homework/test problems of the form “show that the regular languages are closed under operation op ”

What would Sipser do? **Assignment Project Exam Help**

- Give the “proof recognizing regular language(<https://eduassistpro.github.io/>) take machine(s) ate a new machine
Add WeChat edu_assist_pro
- Explain in a few sentences why the construction works
- Give a formal description of the construction
- No need to formally prove the construction works