

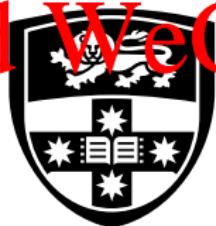
COMP2022: Formal Languages and Logic

2018 Semester 2, Week 5

Assignment Project Exam Help

<https://eduassistpro.github.io>

Add WeChat `edu_assist_pro`



THE UNIVERSITY OF
SYDNEY

COMMONWEALTH OF AUSTRALIA

Copyright Regulations 1969

Assignment Project Exam Help

WARNING

This
on be
Cop

<https://eduassistpro.github.io>

The material in this communication may be subject
under the Act. Any further copying or communicat
material by you may be subject of copyright protec

Add WeChat edu_assist_pro

Do not remove this notice.

OUTLINE

Assignment Project Exam Help

- Non-Deterministic Finite Automata (NFA)
 - Non-determinism
 - ϵ -transitions

- ▶ <https://eduassistpro.github.io>
 - ▶ Minimal DFA
 - ▶ Regular Languages and Closure properties
 - ▶ Add WeChat edu_assist_pro
 - ▶ Regular Expressions (introduction)

Add WeChat `edu_assist_pr`

NON DETERMINISTIC FINITE AUTOMATA (NFA)

DFA

- ▶ Has exactly one transition per input from each state
- ▶

NFA

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

- ▶ can have any number of transitions per input f
- ▶ so some steps of the computation might be static
- ▶ can also have ϵ -transitions

- ▶ i.e. transitions which the automaton can follow without scanning any input

Add WeChat [edu_assist_pro](https://eduassistpro.github.io/)

NON-DETERMINISM

Assignment Project Exam Help

- ▶ As we scan through a string, the NFA can be in many states at the same time
- ▶ <https://eduassistpro.github.io>
- ▶ to a final state
 - ▶ i.e. if we are in at least one accept state, after reading input
- ▶ Parallel computation

Add WeChat `edu_assist_pro`

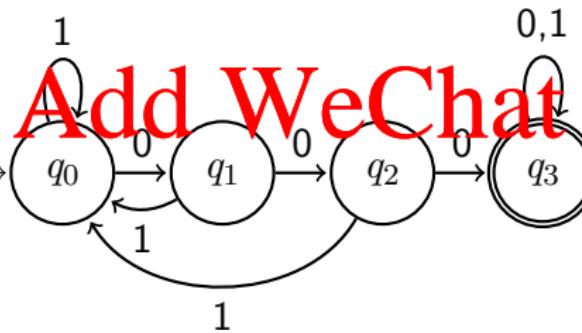
EXAMPLE

NFA accepting strings over $\{0, 1\}$ containing substring “000”

Assignment Project Exam Help



The c



Add WeChat edu_assist_pro

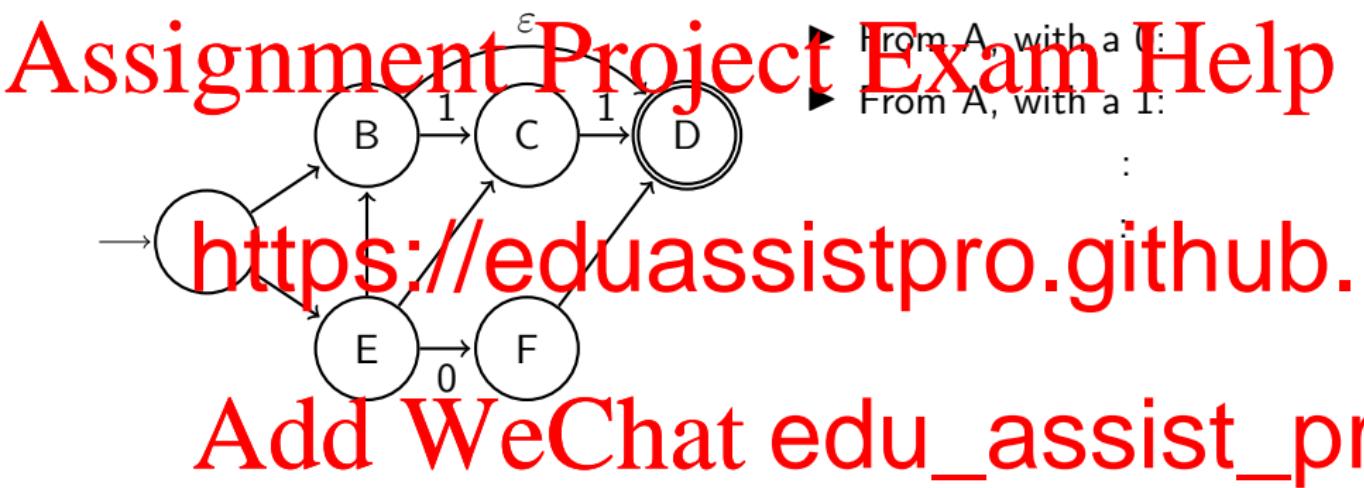
NFA WITH ϵ -TRANSITIONS

Assignment Project Exam Help

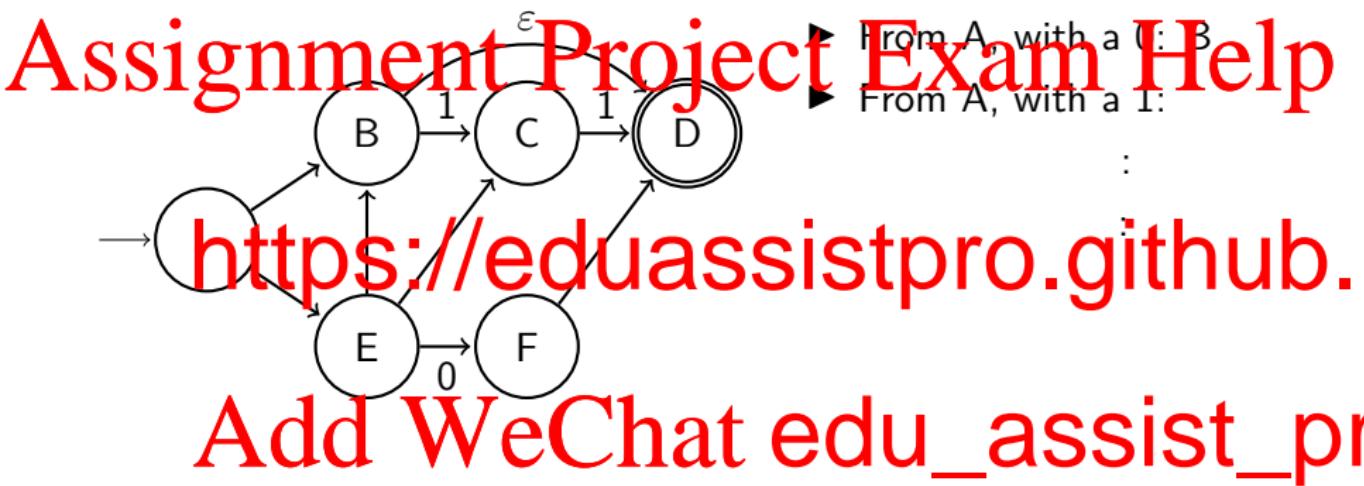
► <https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

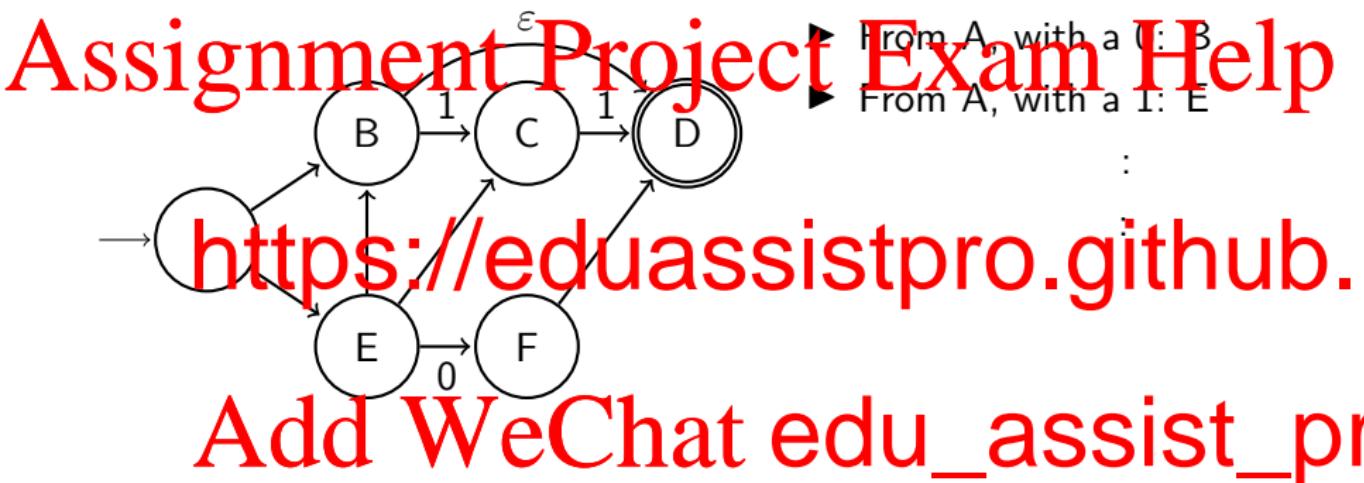
EXAMPLE



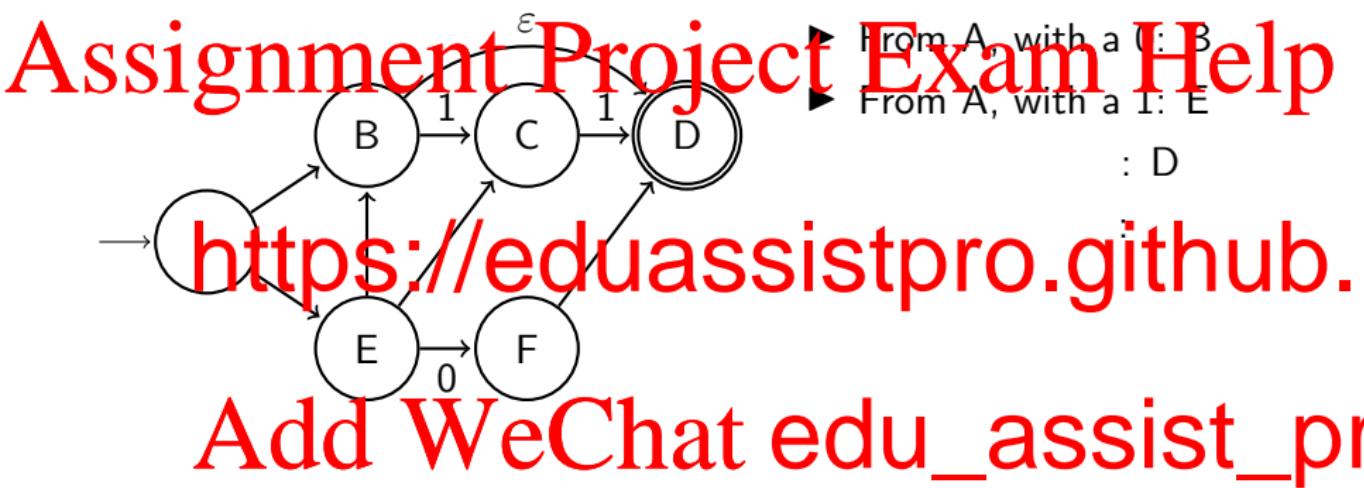
EXAMPLE



EXAMPLE

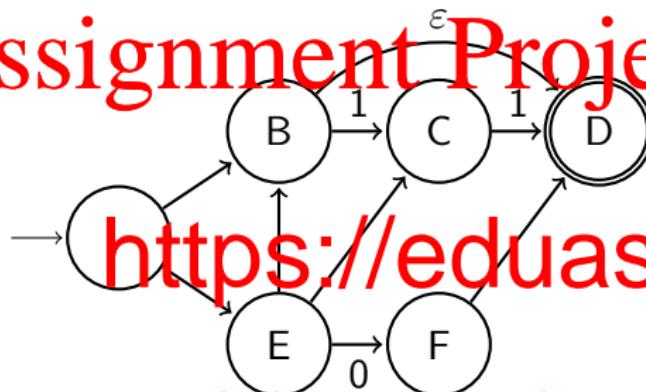


EXAMPLE



EXAMPLE

Assignment Project Exam Help



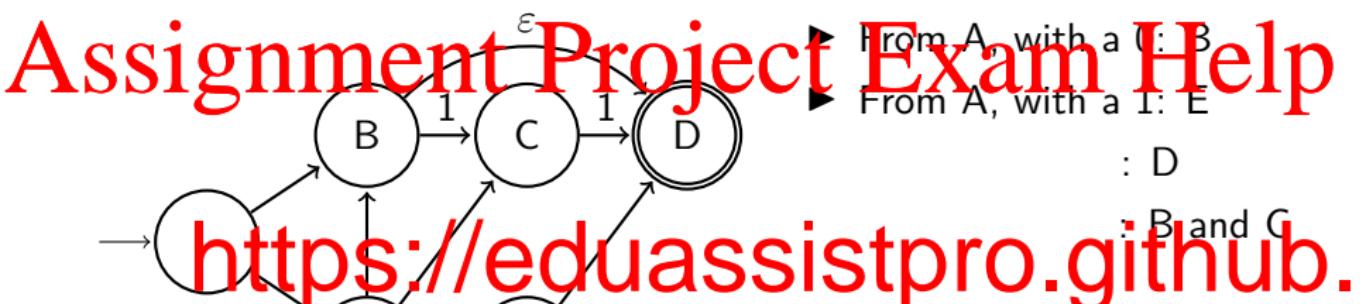
Exam H

1

Band C

Add WeChat edu_assist_pr

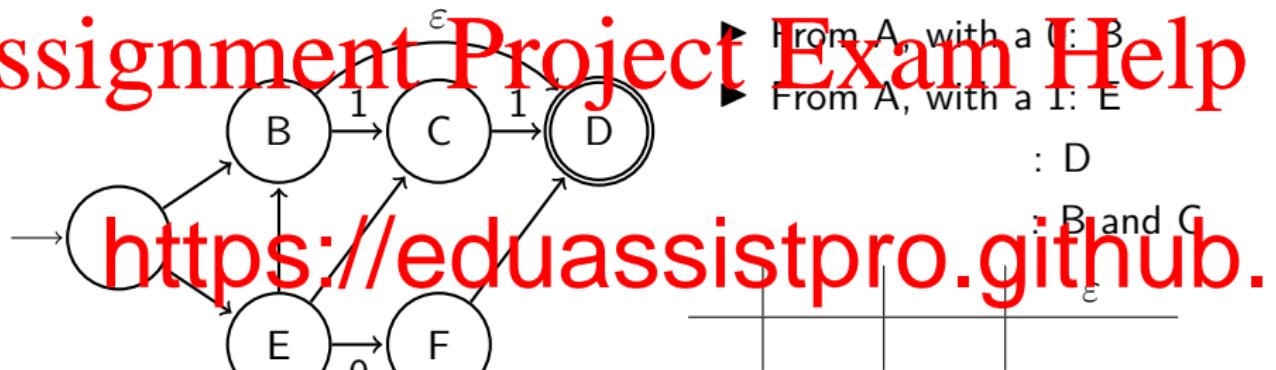
EXAMPLE



So from A with 0, we can potentially also reach D without needing to scan more input.
(Epsilon closure (*later in the lecture*))

EXAMPLE

Assignment Project Exam Help



So from A with 0, we can potentially also reach D without needing to scan more input.
(Epsilon closure (*later in the lecture*))

Add WeChat [edu_assist_pro](https://eduassistpro.github.io)

E	{F}	\emptyset	{B, C}
F	{D}	\emptyset	\emptyset

FORMAL DEFINITION OF NFA

Definition: A non-deterministic finite automaton is a 5-tuple

Assignment Project Exam Help

► $(Q, \Sigma, \delta, q_0, F)$ where:

► Q is a finite set called the states,

►

► <https://eduassistpro.github.io>

► $F \subseteq Q$ is the set of accept states.

*Recall:
Add WeChat edu_assist_pro

► if $A = \{a, b, c\}$ then

$$\mathcal{P}(A) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$$

► $ab\varepsilon = a\varepsilon b = \varepsilon ab = ab$

► $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$

TRANSITION FUNCTION FOR NFA

~~Assignment Project Exam Help~~

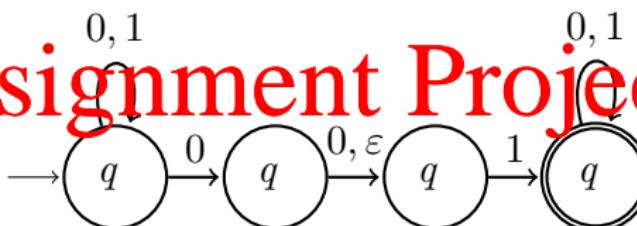
► <https://eduassistpro.github.io/>

\emptyset

► Add WeChat edu_assist_pro

► Note: when following epsilon transitions,
in the same state (as if $q \in \delta(q, \varepsilon)$ for all $q \in Q$)

EXAMPLE



Assignment Project Exam Help

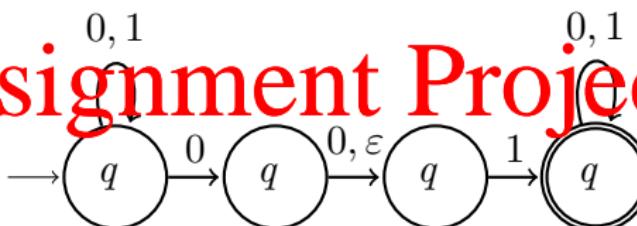
<https://eduassistpro.github.io> by.

- $Q = \{q_1, q_2, q_3, q_4\}$
- $\Sigma = ?$
- The start state is q_1
- The set of accept states is $\{q_4\}$

			ε
q_1			
q_2			
q_3			
q_4	$\{q_4\}$	$\{q_4\}$	\emptyset

Some strings where N reaches the accept state: 01, 0000001

EXAMPLE



Assignment Project Exam Help

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

- $Q = \{1, 2, 3, 4\}$
- $\Sigma = \{0, 1\}$
- The start state is q_1
- The set of accept states is $\{q_4\}$

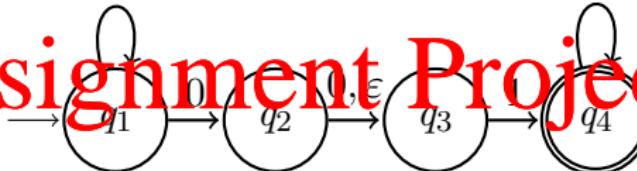
			ε
q_1			
q_2			
q_3			
q_4	$\{q_4\}$	$\{q_4\}$	\emptyset

Some strings where N reaches the accept state: 01, 0000001

NFA COMPUTATION : PARALLELISM

0, 1

0, 1

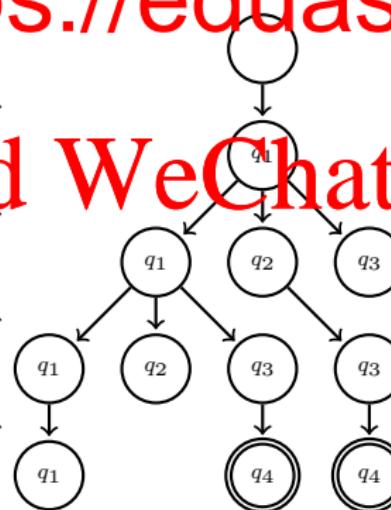
Seve
a tree<https://eduassistpro.github.io>

Read 1 -----

Read 0 -----

Read 0 -----

Read 1 -----



LANGUAGE OF AN NFA

Assignment Project Exam Help

- ▶ A string w is accepted by an NFA if at least one sequence of

<https://eduassistpro.github.io>

- ▶ The language recognised by an NFA (i.e. the accepted) is called regular

Add WeChat edu_assist_pro

EXAMPLES OF NFA (1)

Assignment Project Exam Help

L_1 : Strings over $\{a, b, c\}$ ending with an a

L_1 : Strings over $\{a, b, c\}$ ending with an a

<https://eduassistpro.github.io>



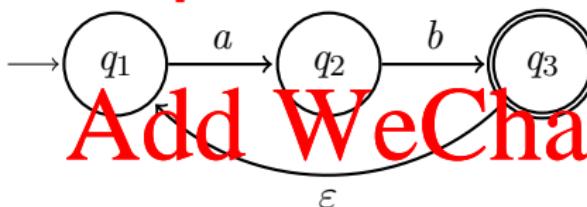
→ Add WeChat edu_assist_pr

EXAMPLES OF NFA (2)

Assignment Project Exam Help

L_2 : Strings over $\{a, b, c\}$ composed of at least one repetition of the su

<https://eduassistpro.github.io>



Add WeChat edu_assist_pro

EXAMPLES OF NFA (3)

Assignment Project Exam Help

L_3 : Strings over $\{0, 1\}$ with a 1 in the third last position

<https://eduassistpro.github.io>



OUTLINE

- # Assignment Project Exam Help
- ▶ Non-Deterministic Finite Automata (NFA)
 - ▶ Non-determinism
 - ▶ ϵ -transitions
 - ▶ <https://eduassistpro.github.io>
 - ▶ Minimal DFA
 - ▶ Add WeChat edu_assist_pro
 - ▶ Regular Languages and Closure properties
 - ▶ Regular Expressions (introduction)

EQUIVALENCE OF NFAs AND DFAS

- # Assignment Project Exam Help
- ▶ DFA: exactly one transition per input from each state
 - ▶ NFA: zero, one or several possible transitions, ϵ -transitions
 - ▶

<https://eduassistpro.github.io>

But ar

by a DFA?

Add WeChat edu_assist_pro

No. If a language is recognised by a NFA then we can always devise a DFA which recognises it.

Theorem: Every NFA has an equivalent DFA

REGULAR LANGUAGES AND FINITE AUTOMATA

Theorem (Kleene 56):

The class of regular languages is exactly the same as the class of languages accepted by DFAs

The

The cl

languages accepted by NFAs

⇒ Add WeChat `edu_assist_pro`
and at least one NFA

Can we transform NFAs into DFAs?

TRANSFORMING A NFA INTO A DFA: INTRO

Key idea: Read the NFA as if it was a DFA and keep in memory the possible set of states it could be in for each given input (i.e. similar to the levels of the decision tree example.)

Each state
consists of

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

When we transition from a state X , with input i , to a state Y , we must also consider all the ways in which we could have had transitions before and after reading i .

- i.e. We transition to the set of states reachable via paths using any number of ϵ transitions before and/or after i

EPSILON-CLOSURE

The *Epsilon-closure* of a state q , denoted $E(q)$, is the set of all states which can be reached from q by 0 or more ϵ -transitions.

► $q \in E(q)$

►

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

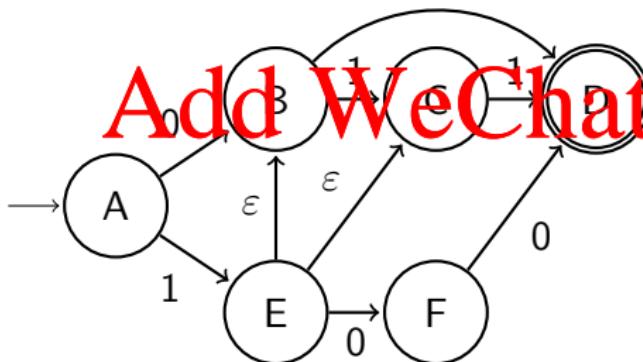
EPSILON-CLOSURE

The *Epsilon-closure* of a state q , denoted $E(q)$, is the set of all states which can be reached from q by 0 or more ϵ -transitions.

► $q \in E(q)$

►

<https://eduassistpro.github.io>



Add WeChat edu_assist_pro

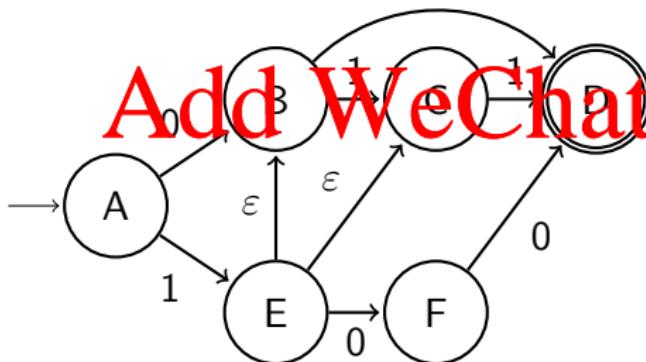
EPSILON-CLOSURE

The *Epsilon-closure* of a state q , denoted $E(q)$, is the set of all states which can be reached from q by 0 or more ϵ -transitions.

► $q \in E(q)$

►

<https://eduassistpro.github.io>



Add WeChat edu_assist_pro

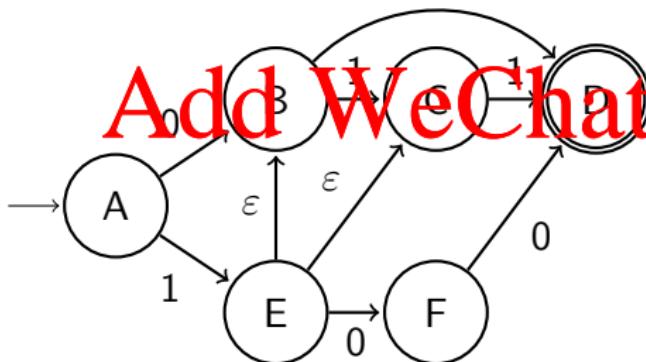
EPSILON-CLOSURE

The *Epsilon-closure* of a state q , denoted $E(q)$, is the set of all states which can be reached from q by 0 or more ϵ -transitions.

► $q \in E(q)$

►

<https://eduassistpro.github.io>



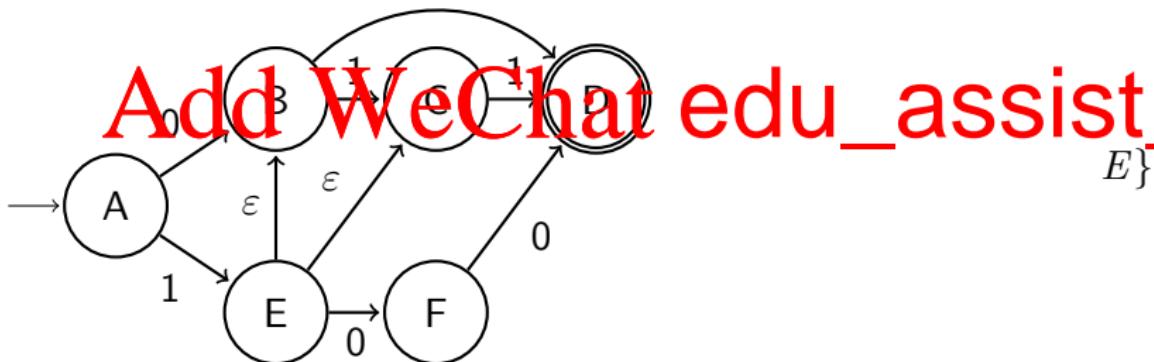
Add WeChat edu_assist_pro

EPSILON-CLOSURE

The *Epsilon-closure* of a state q , denoted $E(q)$, is the set of all states which can be reached from q by 0 or more ε -transitions.

- $q \in E(q)$
 -

<https://eduassistpro.github.io>



EPSILON-CLOSURE FOR SETS

Assignment Project Exam Help

The E

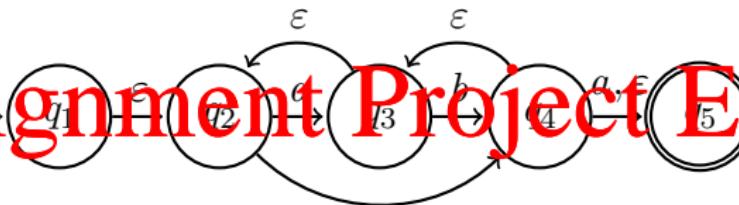
unio

<https://eduassistpro.github.io>

$$E(S \cup T) = E(S) \cup E(T)$$

Add WeChat edu_assist_pro

CONSTRUCTING EPSILON-CLOSURES



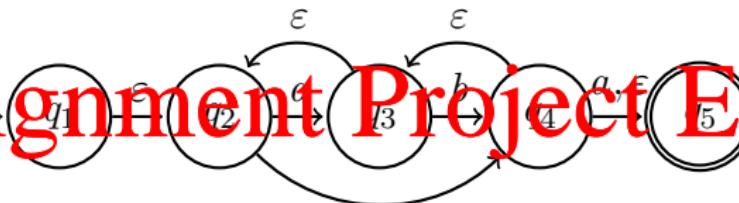
<https://eduassistpro.github.io>

q_1			2
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset
q_3	\emptyset	q_4	$\{q_4\}$
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$
q_5	\emptyset	\emptyset	\emptyset

$$E(\{q_1, q_3, q_5\}) =$$

Add WeChat `edu_assist_pro`

CONSTRUCTING EPSILON-CLOSURES



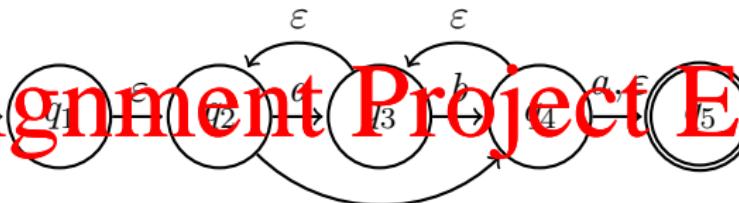
<https://eduassistpro.github.io>

q_1			2
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset
q_3	\emptyset	q_4	$\{q_4\}$
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$
q_5	\emptyset	\emptyset	\emptyset

$$E(\{q_1, q_3, q_5\}) =$$

Add WeChat `edu_assist_pro`

CONSTRUCTING EPSILON-CLOSURES



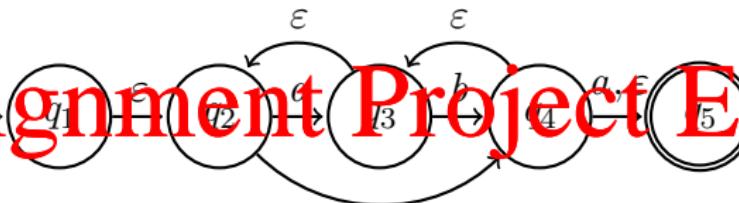
Assignment Project Exam Help

q_1			2
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset
q_3	\emptyset	q_4	$\{q_3\}$
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$
q_5	\emptyset	\emptyset	\emptyset

$$E(\{q_1, q_3, q_5\}) =$$

Add WeChat [edu_assist_pro](https://eduassistpro.github.io)

CONSTRUCTING EPSILON-CLOSURES



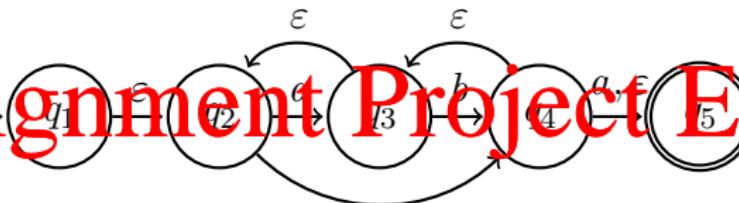
Assignment Project Exam Help

q_1			2
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset
q_3	\emptyset	q_4	$\{q_3\}$
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$
q_5	\emptyset	\emptyset	\emptyset

$$E(\{q_1, q_3, q_5\}) =$$

Add WeChat [edu_assist_pro](https://eduassistpro.github.io)

CONSTRUCTING EPSILON-CLOSURES



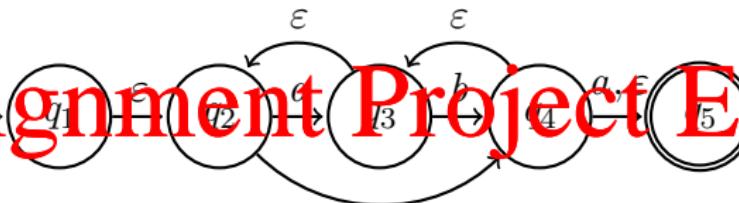
Assignment Project Exam Help

q_1			2
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset
q_3	\emptyset	q_4	$\{q_3\}$
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$
q_5	\emptyset	\emptyset	\emptyset

$$E(\{q_1, q_3, q_5\}) =$$

Add WeChat [edu_assist_pro](https://eduassistpro.github.io)

CONSTRUCTING EPSILON-CLOSURES



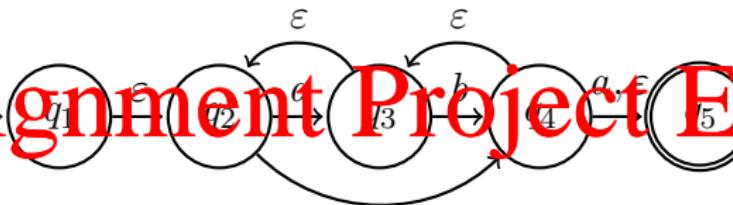
Assignment Project Exam Help

q_1			2
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset
q_3	\emptyset	q_4	$\{q_3\}$
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$
q_5	\emptyset	\emptyset	\emptyset

$$E(\{q_1, q_3, q_5\}) =$$

Add WeChat [edu_assist_pro](https://eduassistpro.github.io)

CONSTRUCTING EPSILON-CLOSURES

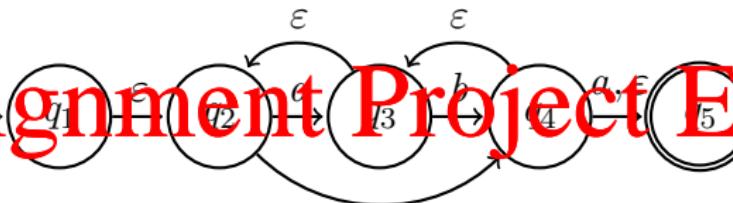


<https://eduassistpro.github.io>

q_1			2		2	2
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset	E		
q_3	\emptyset	q_4	$\{q_4\}$	E		
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$	E		
q_5	\emptyset	\emptyset	\emptyset			

$E(\{q_1, q_3, q_5\})$
 $= E(\{q_1\}) \cup E(\{q_3\}) \cup E(\{q_5\})$
 $=$

CONSTRUCTING EPSILON-CLOSURES



<https://eduassistpro.github.io>

q_1			2	
q_2	$\{q_3, q_4\}$	\emptyset	\emptyset	E
q_3	\emptyset	q_4	$\{q_4\}$	E
q_4	$\{q_5\}$	\emptyset	$\{q_3, q_5\}$	E
q_5	\emptyset	\emptyset	\emptyset	

Add WeChat edu_assist_pro

$$\begin{aligned}E(\{q_1, q_3, q_5\}) \\= E(\{q_1\}) \cup E(\{q_3\}) \cup E(\{q_5\}) \\= \{q_1, q_2, q_3, q_5\}\end{aligned}$$

NFA TO DFA ALGORITHM

Assignment Project Exam Help

We want to convert a NFA $N = (Q, \Sigma, \delta, q_0, F)$ into an equivalent DFA $M = (Q', \Sigma, \delta', q'_0, F')$ which recognises $L(N)$

- ▶ <https://eduassistpro.github.io/>
- ▶ Σ does not change
- ▶ $\delta'(R, a) = \bigcup_{r \in R} E(\delta(r, a))$
- ▶ $q'_0 = E(q_0)$
- ▶ $F' = \{R \in Q' | R \text{ contains an accept state of } N\}$

Add WeChat edu_assist_pro

NFA TO DFA ALGORITHM

We want to convert a NFA $N = (Q, \Sigma, \delta, q_0, F)$ into an equivalent DFA $M = (Q', \Sigma, \delta', q'_0, F')$ which recognises $L(N)$.

Algo

1.

$$2. \quad R \subseteq \mathcal{P}(Q)$$

$$a_i \in \Sigma$$

e

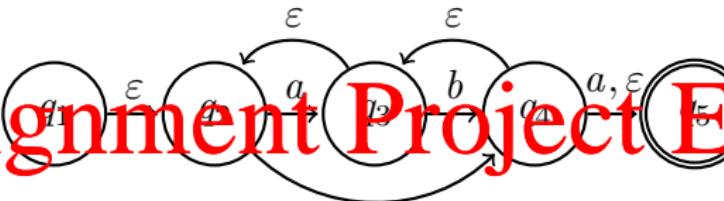
following DFA state as a DFA table entry in either

Add WeChat

► $\delta'(R, a) = \bigcup_{r \in R} E(\delta(r, a))$ un

3. A DFA state is accepting if any of its elements are an NFA accept state.

FROM NFA TO DFA: EXAMPLE



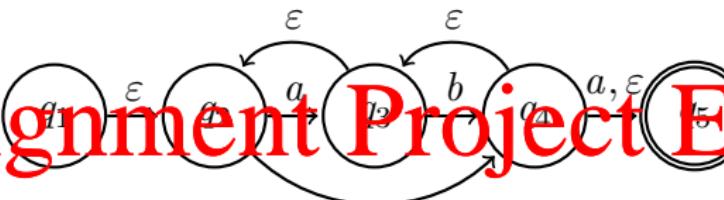
Trans

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



Assignment Project Exam Help

Tran

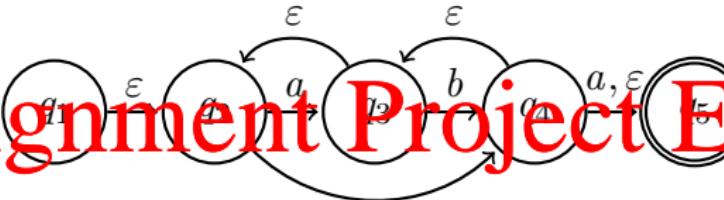
<https://eduassistpro.github.io>

$E(\text{start}) =$

Add WeChat edu_assist_pro

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



Trans

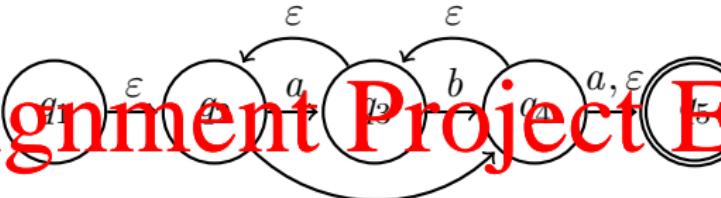
<https://eduassistpro.github.io>

$$E(start) = \{q_1, q_2\}$$

Add WeChat edu_assist_pr

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



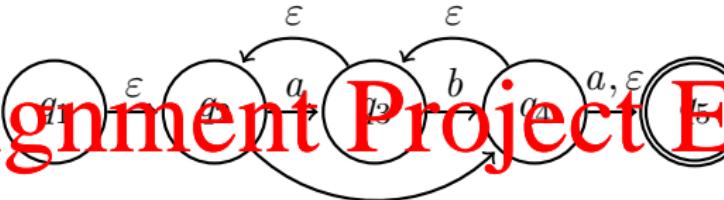
Trans

<https://eduassistpro.github.io>

<https://eduassispro.g>
 $E(start) = \{q_1, q_2\}$ $\{q_2, q_3, q$
Add WeChat edu_assis

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



Trans

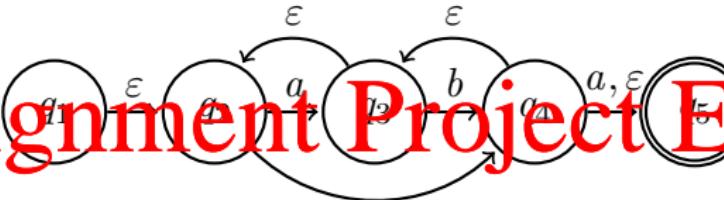
<https://eduassistpro.github.io>

<https://eduassispro.g>
 $E(start) = \{q_1, q_2\}$ $\{q_2, q_3, q$
 $\{q_2, q_3, q_4, q_5\}$

Add WeChat edu_assis

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



Trans

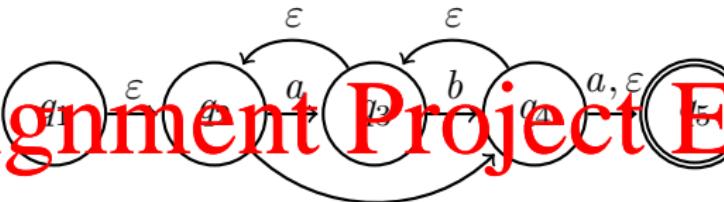
<https://eduassistpro.github.io>

<https://eduassispro.g>
 $E(start) = \{q_1, q_2\}$ $\{q_2, q_3, q$
 $\{q_2, q_3, q_4, q_5\}$

Add WeChat edu_assis

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



Trans

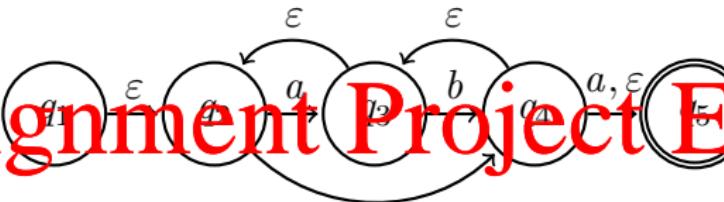
<https://eduassistpro.github.io>

$E(start) = \{q_1, q_2\}$	$\{q_2, q_3, q_4\}$
$\{q_2, q_3, q_4, q_5\}$	\emptyset

Add WeChat edu_assist_pro

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



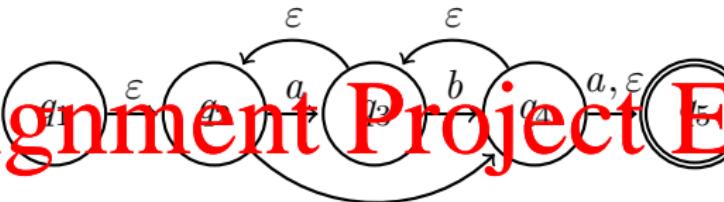
Trans

<https://eduassistpro.github.io>

$E(start) = \{q_1, q_2\}$	$\{q_2, q_3, q_4, q_5\}$
$\{q_2, q_3, q_4, q_5\}$	$\{q_2, q_3, q_4, q_5\}$
\emptyset	\emptyset

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



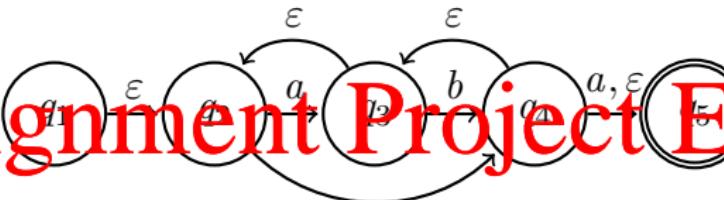
Trans

<https://eduassistpro.github.io>

$E(start) = \{q_1, q_2\}$	$\{q_2, q_3, q_4, q_5\}$
$\{q_2, q_3, q_4, q_5\}$	$\{q_2, q_3, q_4, q_5\}$
\emptyset	\emptyset

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



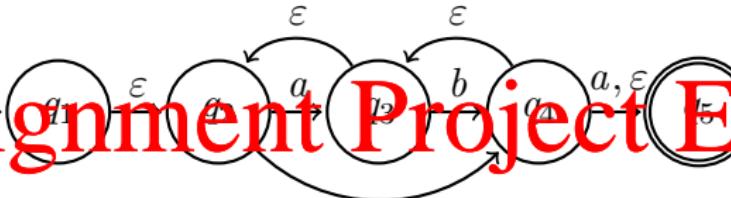
Trans

<https://eduassistpro.github.io>

$E(start) = \{q_1, q_2\}$	$\{q_2, q_3, q_4\}$
$\{q_2, q_3, q_4, q_5\}$	$\{q_2, q_3, q_4\}$
\emptyset	\emptyset

Resulting DFA:

FROM NFA TO DFA: EXAMPLE



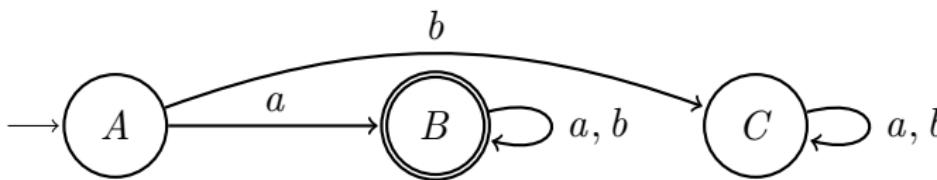
Trans

<https://eduassistpro.github.io>

<i>A</i>	$E(start) = \{q_1, q_2\}$	$\{q_2, q_3, q$	
<i>B</i>	$\{q_2, q_3, q_4, q_5\}$	$\{q_2, q_3, q$	
<i>C</i>	\emptyset	\emptyset	

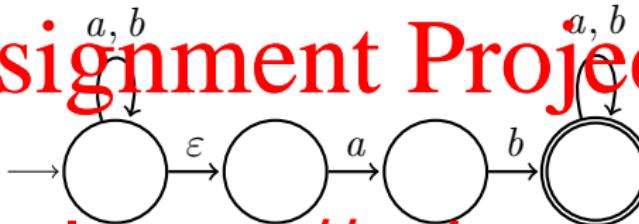
Add WeChat edu_assi

Resulting DFA:



FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help



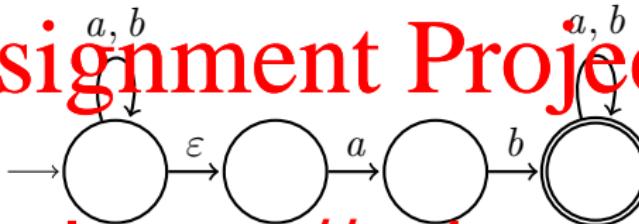
<https://eduassistpro.github.io>

Tran

Add WeChat edu_assist_pro	q	

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help



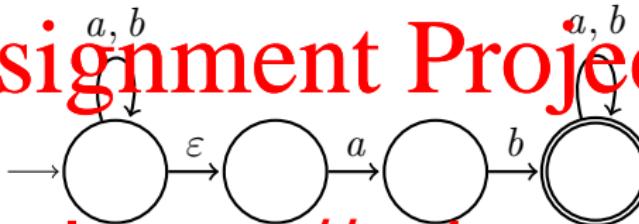
<https://eduassistpro.github.io>

Tran

$E(\text{start}) =$	q_1	

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help



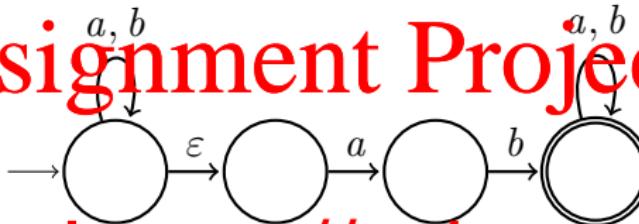
<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	q_1	

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help

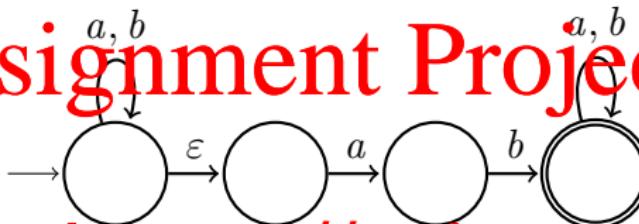


<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2, q_3\}$	

FROM NFA TO DFA: EXAMPLE

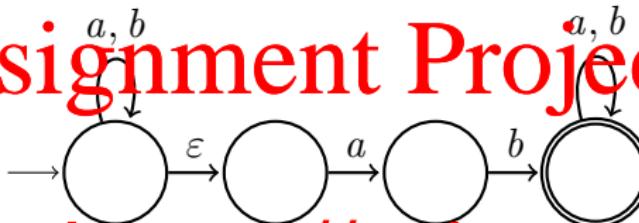


<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2,$	
$\{q_1, q_2, q_3\}$		

FROM NFA TO DFA: EXAMPLE



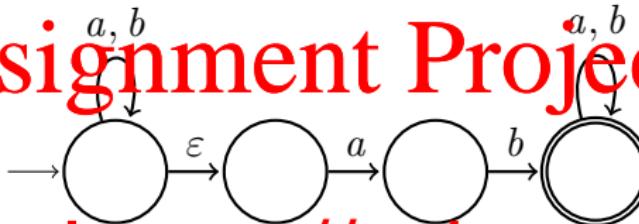
Assignment Project Exam Help
<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2,$	
$\{q_1, q_2, q_3\}$		

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help



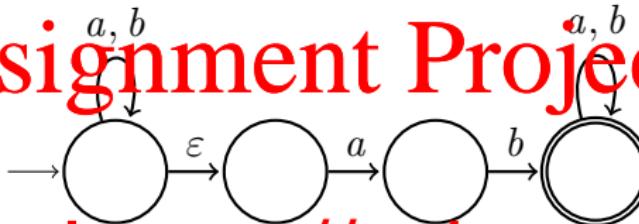
<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2, q_3\}$	$\{q_1, q_2, q_3\}$

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help



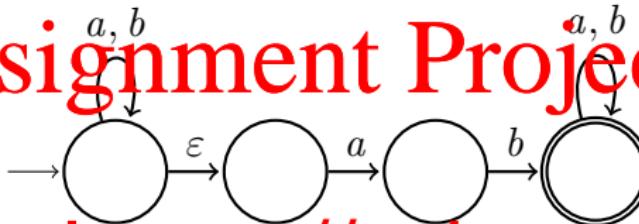
<https://eduassistpro.github.io>

Trans

$E(\text{start}) = \{q_1, q_2\}$ $\{q_1, q_2, q_3\}$	$\overset{a}{\{q_1, q_2, 3\}}$	1 2 4
---	--------------------------------	-------------

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help

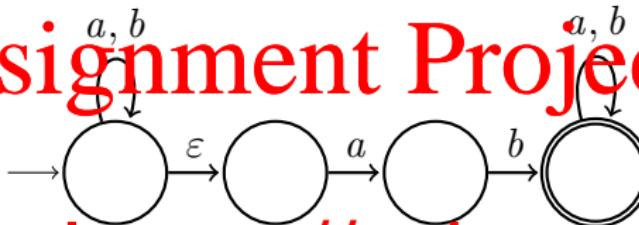


<https://eduassistpro.github.io>

Trans

Add WeChat edu_as

FROM NFA TO DFA: EXAMPLE

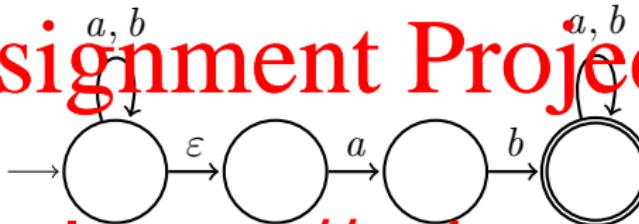


<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2, q_3\}$ $\{q_1, q_2, q_4\}$	$\{q_1, q_2, q_3, q_4\}$	1 2 4

FROM NFA TO DFA: EXAMPLE

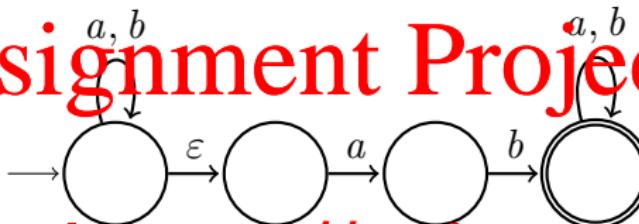


<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2,$ $\{q_1, q_2, q_3\}$ $\{q_1, q_2, q_4\}$ <u>$\{q_1, q_2, q_3, q_4\}$</u>	q $\{q_1, q_2, 3$ $\{q_1, q_2, q_3, q_4\}$	1 2 4

FROM NFA TO DFA: EXAMPLE

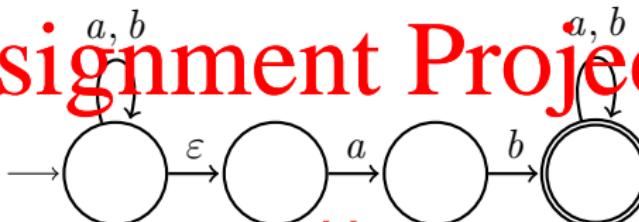


<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2, q_3\}$ $\{q_1, q_2, q_4\}$ <u>$\{q_1, q_2, q_3, q_4\}$</u>	$\{q_1, q_2, q_3\}$ $\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_3, q_4\}$
$\{q_1, q_2, q_3\}$ $\{q_1, q_2, q_4\}$ <u>$\{q_1, q_2, q_3, q_4\}$</u>	$\{q_1, q_2, q_3\}$ $\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_3, q_4\}$

FROM NFA TO DFA: EXAMPLE



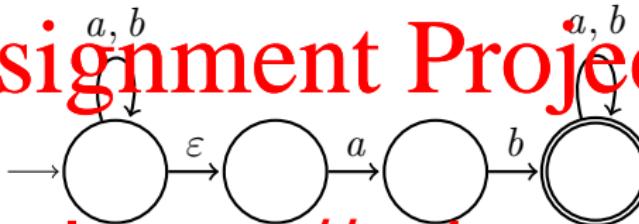
<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2,$ $\{q_1, q_2, q_3\}$ $\{q_1, q_2, q_4\}$ <u>$\{q_1, q_2, q_3, q_4\}$</u>	q $\{q_1, q_2, 3$ $\{q_1, q_2, q_3, q_4\}$ $\{q_1, q_2, q_3, q_4\}$	1 2 4 $\{q_1, q_2, q_4\}$

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help



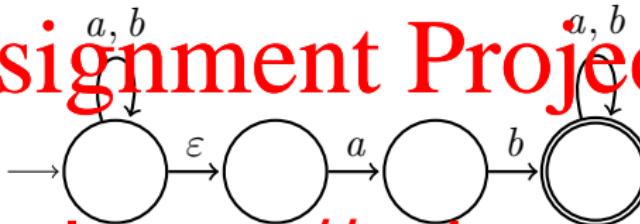
<https://eduassistpro.github.io>

Tran

$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2,$	$q_1, q_2, q_3,$
$\{q_1, q_2, q_3\}$	$\{q_1, q_2, q_3\}$	$1 \quad 2 \quad 4$
<u>$\{q_1, q_2, q_4\}$</u>	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_4\}$
$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_4\}$

FROM NFA TO DFA: EXAMPLE

Assignment Project Exam Help



<https://eduassistpro.github.io>

Tran

A	$E(\text{start}) = \{q_1, q_2\}$	$\{q_1, q_2,$	q
B	$\{q_1, q_2, q_3\}$	$\{q_1, q_2, 3$	1 2 4
C	$\{q_1, q_2, q_4\}$	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_4\}$
D	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_4\}$

OUTLINE

- # Assignment Project Exam Help
- ▶ Non-Deterministic Finite Automata (NFA)
 - ▶ Non-determinism
 - ▶ ϵ -transitions
 - ▶ <https://eduassistpro.github.io/>
 - ▶ Minimal DFA
 - ▶ Add WeChat edu_assist_pro
 - ▶ Regular Languages and Closure properties
 - ▶ Regular Expressions (introduction)

MINIMISATION OF DFA (IDEA)

Every DFA has a unique equivalent *minimal DFA*

Assignment Project Exam Help

Definition: Two states s and t are equivalent if: for any string, the path from s to t exists.

<https://eduassistpro.github.io>

To re

- Add WeChat edu_assist**

 1. If s is accepting and t non accepting
equivalent
 2. If, with input x , there is a transition from s to t' and we know that s' and t' are not equivalent, then s and t are not equivalent

Then merge equivalent states.

Add WeChat edu assist

MINIMISATION OF DFA: TABLE-FILLING ALGORITHM

Assignment Project Exam Help

1. Examine all pairs of states and find all pairs s, t that are NOT equivalent i.e. satisfying either of:

1.1 s is accepting and t is not or vice versa

ate t'
ivalent

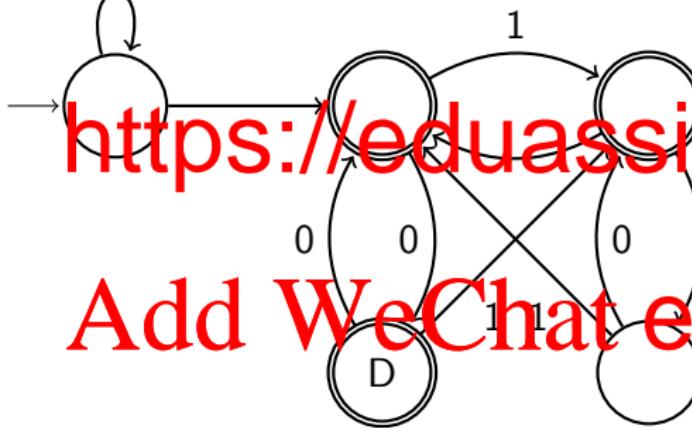
2. <https://eduassistpro.github.io>

Theorems Add WeChat edu_assist_pro

1. If two states are not distinguished by the table algorithm, then the states are equivalent
2. The equivalence of states is transitive

EXAMPLE

Assignment Project Exam Help



Add WeChat edu_assist_pro

THE TABLE OF STATE NON-EQUIVALENCES

A	-	-	-	-	-
B		-	-	-	
C				-	
D				-	-
E					

<https://eduassistpro.github.io>

1. $, E$ are

non-accepting and all the others are.

Add WeChat edu_assist_pro

THE TABLE OF STATE NON-EQUIVALENCES

A	-	-	-	-	-
B	x	-	-	-	-
C	x	-	-	-	-
D	x	-	-	-	-
E					

<https://eduassistpro.github.io>

1. $, E$ are

non-accepting and all the others are.

2. Look at each remaining pair

Add WeChat edu_assist_pro

THE TABLE OF STATE NON-EQUIVALENCES

A	-	-	-	-	-
B	x	-	-	-	-
C	x	-	-	-	-
D	x	-	-	-	-
E					

<https://eduassistpro.github.io>

1. non-accepting and all the others are.

2. Look at each remaining pair

 - $\delta(A, 0) = A$ and $\delta(E, 0) = C$, but

THE TABLE OF STATE NON-EQUIVALENCES

A	-	-	-	-	-
B	x	-	-	-	-
C	x	x	-	-	-
D	x			-	-
E					

<https://eduassistpro.github.io>

1. A, B, C are accepting states, D, E are non-accepting and all the others are.

2. Look at each remaining pair:

► $\delta(A, 0) = A$ and $\delta(E, 0) = C$, but $A \neq E$

► $\delta(B, 0) = D$ and $\delta(C, 0) = E$, but $D \neq E$

$B \neq C$

THE TABLE OF STATE NON-EQUIVALENCES

A	-	-	-	-	-
B	x	-	-	-	-
C	x	x	-	-	-
D	x			-	-
E					

<https://eduassistpro.github.io>

1. $, E$ are

non-accepting and all the others are.

2. Look at each remaining pair

- $\delta(A, 0) = A$ and $\delta(E, 0) = C$, but $/ E$
- $\delta(B, 0) = D$ and $\delta(C, 0) = E$, but $D \not\equiv E$ $B \not\equiv C$
- We can't find a reason to claim $B \not\equiv D$ yet

Add WeChat edu_assist_pro

THE TABLE OF STATE NON-EQUIVALENCES

A	-	-	-	-	-
B	x	-	-	-	-
C	x	x	-	-	-
D	x		x	-	-
E					

<https://eduassistpro.github.io>

1. $\{A, B, C\}$ and $\{D, E\}$ are non-accepting and all the others are.

2. Look at each remaining pair:

- $\delta(A, 0) = A$ and $\delta(E, 0) = C$, but $A \neq C$ / $E \neq C$
- $\delta(B, 0) = D$ and $\delta(C, 0) = E$, but $D \neq E$ / $B \neq C$
- We can't find a reason to claim $B \neq D$ yet
- $\delta(C, 0) = E$ and $\delta(D, 0) = B$, and $E \neq B$, therefore $C \neq D$

THE TABLE OF STATE NON-EQUIVALENCES

A	-	-	-	-	-
B	x	-	-	-	-
C	x	x	-	-	-
D	x		x	-	-
E					

<https://eduassistpro.github.io>

1. $\delta(A, 0) = A$, $\delta(B, 0) = D$, $\delta(C, 0) = C$, $\delta(D, 0) = B$, $\delta(E, 0) = E$, $A \neq B$, $A \neq C$, $A \neq D$, $A \neq E$, $B \neq C$, $B \neq D$, $B \neq E$, $C \neq D$, $C \neq E$, $D \neq E$, E are non-accepting and all the others are.

2. Look at each remaining pair:

- $\delta(A, 0) = A$ and $\delta(E, 0) = C$, but $A \neq E$, $A \neq C$
- $\delta(B, 0) = D$ and $\delta(C, 0) = E$, but $D \neq E$, $B \neq C$
- We can't find a reason to claim $B \neq D$ yet
- $\delta(C, 0) = E$ and $\delta(D, 0) = B$, and $E \neq B$, therefore $C \neq D$

3. Repeat step 2 until no changes are found

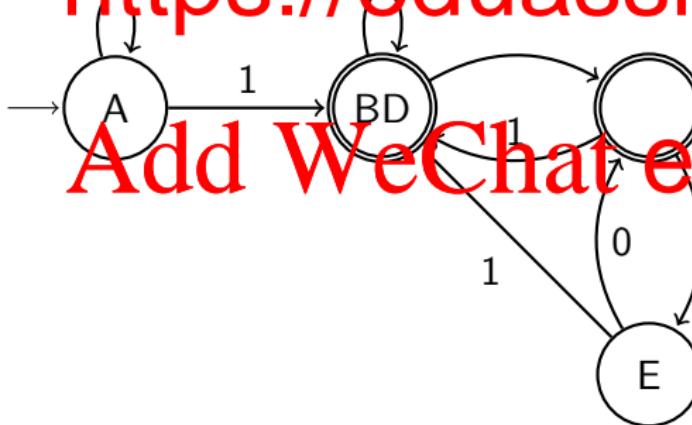
MERGING EQUIVALENT STATES

Theorem: If two states are not distinguished by the table-filling algorithm, then that pair of states are equivalent.

Assignment Project Exam Help

We can merge together equivalent states. The minimal DFA for the ex

<https://eduassistpro.github.io>



Add WeChat edu_assist_pro

RESULTING DFA

Assignment Project Exam Help

► <https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OUTLINE

Assignment Project Exam Help

- ▶ Non-Deterministic Finite Automata (NFA)
 - ▶ Non-determinism
 - ▶ ϵ -transitions

- ▶ <https://eduassistpro.github.io>
 - ▶ Minimal DFA
 - ▶ Add WeChat edu_assist_pr
 - ▶ Regular Languages and Closure prop
 - ▶ Regular Expressions (introduction)

Add WeChat `edu_assist_pr`

REVISION

Definition:

A language is regular if and only if there exists a finite automaton which recognises it.

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

- ▶ \emptyset
 - ▶ $\{\varepsilon\}$ is a regular language.
 - ▶ For all $a \in \Sigma$, the set $\{a\}$ is a regular language.

Think about how to make a DFA for each of these languages

CLOSURE PROPERTIES OF REGULAR LANGUAGES

Let L_1 and L_2 be regular languages.

Assignment Project Exam Help

The *regular operations* are defined as follows:



$\text{https://eduassistpro.github.io}$

1 2 k

i

k

Theorems

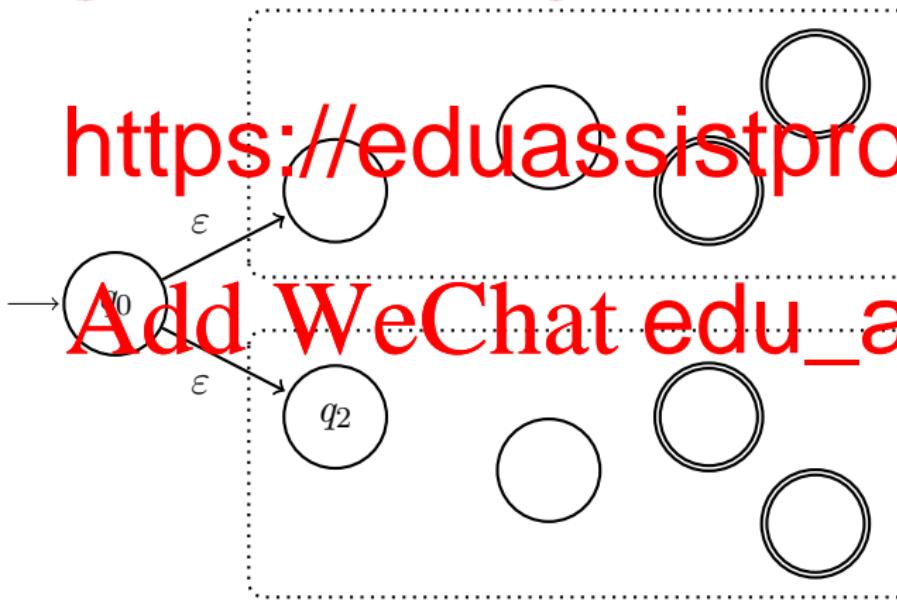
Add WeChat edu_assist_pro

- The union of two regular languages is regular
- The concatenation of two regular languages is regular
- The star closure of a regular language is regular

UNION OF TWO REGULAR LANGUAGES

If L_1 and L_2 are regular, then finite automata M_1 and M_2 exist which recognise them. we can construct an automaton M which recognises $L_1 \cup L_2 = \{w \mid w \in L_1 \text{ or } w \in L_2\}$.

Assignment Project Exam Help



<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

UNION OF TWO REGULAR LANGUAGES

Let $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$, $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$

Let $\bullet M = (Q, \Sigma, \delta, q_0, F)$ where:

Assignment 1

- q

<https://eduassistpro.github.io>

Add WeChat

- Accepting set $F = F_1 \cup F_2$

M accepts if and only if M_1 or M_2 does

i.e. $L(M) = \{w \mid w \in L_1 \text{ or } w \in L_2\} = L_1 \cup L_2$

CONCATENATION OF TWO REGULAR LANGUAGES

If L_1 and L_2 are regular, then finite automata M_1 and M_2 exist

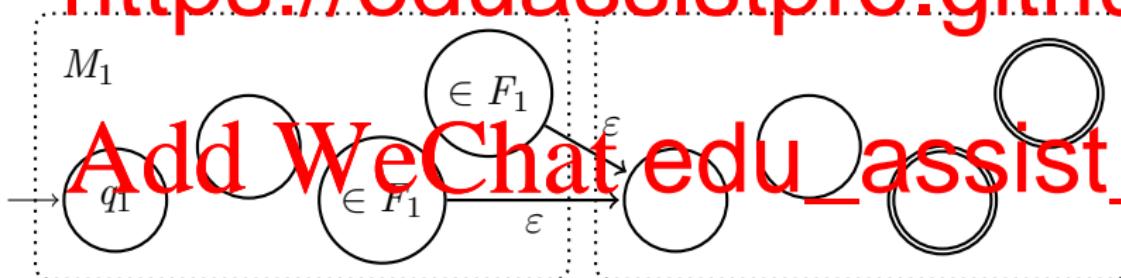
which recognise them.

Assignment Project Exam Help

we ca

$L_1 L_2$

<https://eduassistpro.github.io>



Note that the accept states of M_1 are *not* accept states in M

CONCATENATION OF TWO REGULAR LANGUAGES

Let $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$, $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$

Let $M = (Q, \Sigma, \delta, q_0, F)$ where

- $Q = Q_1 \cup Q_2$

▶

► <https://eduassistpro.github.io/>

$$\text{AddWeChat edu_assi}$$

- Accepting set $F = F_2$ (same as M_2)

$$L(M) = \{w_1 w_2 \mid w_1 \in L_1 \text{ and } w_2 \in L_2\} = L_1 L_2$$

STAR CLOSURE OF A REGULAR LANGUAGE

If L is regular, then a finite automaton M_1 exists which recognises

Assignment Project Exam Help

we can construct an automaton M which recognises

$L^* =$

<https://eduassistpro.github.io>



e.g. if $L = \{a, b\}$ then this automaton recognises
 $\{\epsilon, a, b, aa, ab, bb, aaa, \dots\} = L^*$

STAR CLOSURE OF A REGULAR LANGUAGE

Let $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$

Let $\bullet M = (Q, \Sigma, \delta, q_0, F)$ where:

- # signmen

- ▶ q_0 is the new start state

<https://eduassistpro.github.io>

Add WeChat edu_assist_pr

- Accepting set $F = F_1 \cup \{q_0\}$

$$L(M) = \{w_1 w_2 \dots w_k \mid k \geq 0 \text{ and } w_i \in L \text{ for } i = 1, \dots, k\} = L^*$$

EXAMPLE

Assignment Project Exam Help

Cons
patt

<https://eduassistpro.github.io>

Add WeChat edu_assist_pr

EXAMPLE FROM CONSTRUCTION TO MINIMAL DFA

Assignment Project Exam Help

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

Let

0's

Let

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

COMPLEMENT OF A REGULAR LANGUAGE

Assignment Project Exam Help

The complement of a regular language is regular.
The complement of a language L is $\{x|x \notin L\}$

Exa
Let

<https://eduassistpro.github.io>

Then the complement of L_1 is $L_2 = \{ \quad / \quad \text{ns}$
an even number of a's}

Add WeChat edu_assist_pro

You will explore how to construct a suitable automata in this week's tutorial

INTERSECTION OF A REGULAR LANGUAGE

If L_1 and L_2 are regular, then $L_1 \cap L_2$ is regular.

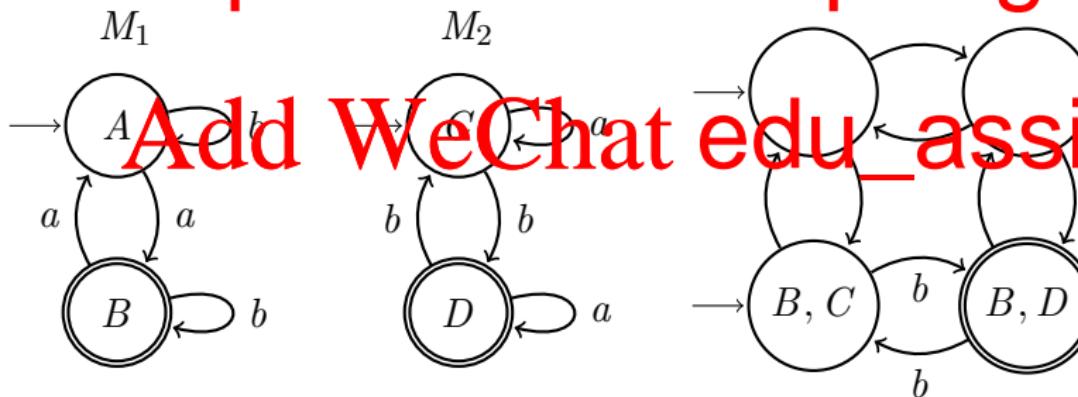
Assignment Project Exam Help

Example:

$$L_1 =$$

$$L_2 =$$

<https://eduassistpro.github.io>



Add WeChat edu_assist_pro

CROSS-PRODUCT OF DFAs

Let $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$, $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ be two DFA

Assignment Project Exam Help

Let $M = (Q, \Sigma, \delta_1, q_0, F)$ where



$\in Q_1$

► <https://eduassistpro.github.io/>

Q_1 and

2

► The transition function δ is defined as

Add WeChat edu_assist_pro

$$\delta((u, v), x) = (\delta_1(u, x), \delta_2(v, x))$$

► Accepting set $F = \{(u, v) | u \in F_1 \text{ and } v \in F_2\}$

Note that M is also deterministic

EXAMPLE CROSS-PRODUCT FOR INTERSECTION

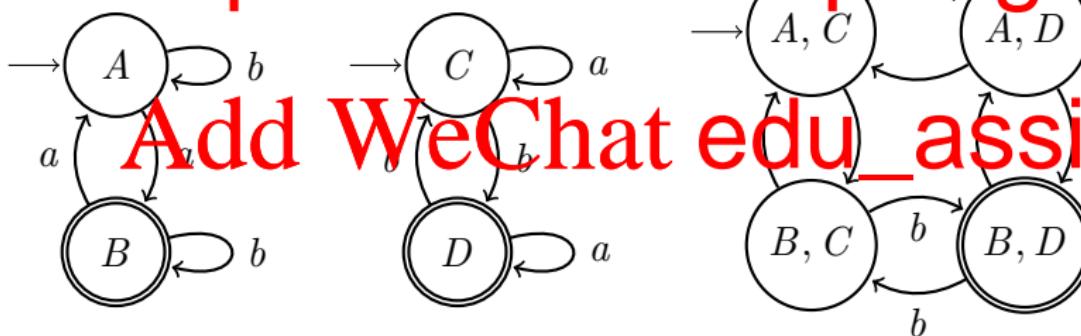
Example:

Assignment Project Exam Help

$L_1 = \{x \mid x \in \{a, b\}^* \text{ and } x \text{ contains an odd number of } a's\}$

$L_2 = \{x \mid x \in \{a, b\}^* \text{ and } x \text{ contains an odd number of } b's\}$

<https://eduassistpro.github.io>



CROSS-PRODUCT CAN BE USED FOR UNION

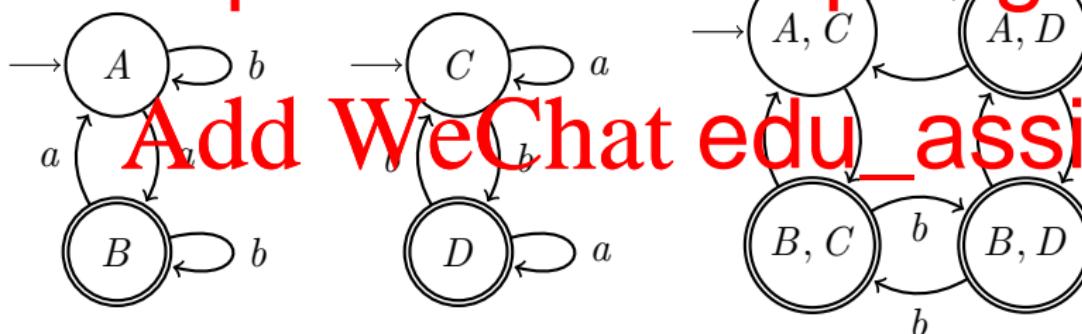
Example:

Assignment Project Exam Help

$L_1 = \{x | x \in \{a, b\}^* \text{ and } x \text{ contains an odd number of } a's\}$

$L_2 = \{x | x \in \{a, b\}^* \text{ and } x \text{ contains an odd number of } b's\}$

<https://eduassistpro.github.io>



OUTLINE

- # Assignment Project Exam Help
- ▶ Non-Deterministic Finite Automata (NFA)
 - ▶ Non-determinism
 - ▶ ϵ -transitions
 - ▶ <https://eduassistpro.github.io>
 - ▶ Minimal DFA
 - ▶ Add WeChat edu_assist_pro
 - ▶ Regular Languages and Closure properties
 - ▶ Regular Expressions (introduction)

REGULAR EXPRESSIONS

In arithmetic we use operations to build up expressions, describing numbers such as

Assignment Project Exam Help

<https://eduassistpro.github.io>

With regular languages, we use regular operations such as regular expressions describing languages such as

Add WeChat edu_assist_pr

(0 | 1)0* (strings starting with 0 or 1, then any

$(ab)^*$ a (any number of ab 's, followed by a)

REGULAR EXPRESSIONS VS FINITE AUTOMATA

- ▶ RegEx: algebraic description of the strings in a language

- ▶ FA: a machine-like description



<https://eduassistpro.github.io>

- ▶ Web browsers
- ▶ Text editors
- ▶ Text formatting systems
- ▶ ...

Add WeChat edu_assist_pro

- ▶ Lexical-analyser generators (Lex, Flex, etc.)
- ▶ Like FA, RegEx define exactly the class of *regular languages*

DEFINITION OF A REGULAR EXPRESSION (REGEx)

- Basic expressions and the language they describe

► If a is any symbol of the input alphabet, then a is a RegEx
and its language $L(a) = \{a\}$

- ε is a RegEx and $L(\varepsilon) = \varepsilon$

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

DEFINITION OF A REGULAR EXPRESSION (REGEx)

- Basic expressions and the language they describe

► If a is any symbol of the input alphabet, then a is a RegEx
and its language $L(a) = \{a\}$

► ε is a RegEx and $L(\varepsilon) = \varepsilon$

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

$R | S$

$R + S$

► Concatenation: RS (sometimes denoted RS)

► Star Closure: R^* is a RegEx

Add WeChat edu_assist_pro

DEFINITION OF A REGULAR EXPRESSION (REGEx)

- Basic expressions and the language they describe

► If a is any symbol of the input alphabet, then a is a RegEx and its language $L(a) = \{a\}$

► ε is a RegEx and $L(\varepsilon) = \varepsilon$

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

$R | S$

$R + S$

► Concatenation: RS (sometimes denoted RS^*)

► Star Closure: R^* is a RegEx

Add WeChat edu_assist_pro

- Precedence of operators: * , \circ , $|$ to a

► e.g. $R | ST^* = (R | (S(T^*)))$, similar to how
 $8 + 21/3^2 = (8 + (21/(3^2)))$

► Use parentheses to change the order of operations

LANGUAGE OF A REGULAR EXPRESSION

Assignment Project Exam Help

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

$$L(\emptyset) = \{\emptyset\}$$

$$L(x \mid y) = \{x, y\}$$

$$L(x^*) = \{\epsilon, x, xx, \dots\}$$

$$L(xy) = \{xy\}$$

Add WeChat `edu_assist_pro`

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

$$L(a \mid b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$$

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

$L(a \mid b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$

$a \mid b)$

$L($ <https://eduassistpro.github.io> $)$

Add WeChat edu_assist_pro

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

$L(a \mid b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$

$a \mid b)$

$L($ <https://eduassistpro.github.io> $)$

Add WeChat edu_assist_pro

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

$$L(a \mid b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$$

$a \mid b$)

$L($ <https://eduassistpro.github.io> $)$

=

Add WeChat edu_assist_pro

OPERATORS ON REGEX: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

a | b)

L(<https://eduassistpro.github.io>*)*

$$= \{a, b, c\}$$

$L(a_1(b_1c_1)) = \text{WeChat edu_assist_pr}$

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

$$L(a \mid b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$$

$a \mid b$)

$L(\text{https://eduassistpro.github.io})$

$$= \{a, b, c\}$$

$$L(a \mid (b \mid c)) = L(a) \cup L(b \mid c)$$

$=$

Add WeChat edu_assist_pro

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project

$$L(a \mid b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$$

$$a \mid b)$$

L(<https://eduassistpro.github.io>*)*

$$= \{a, b, c\}$$

$$L(a \Delta (b \sqcup c)) = L(a) \sqcup L(b \sqcup c)$$

$$= \{a\} \cup \{b, c\}$$

三

OPERATORS ON REGEx: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project

$$L(a \mid b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$$

$$a \mid b)$$

L(<https://eduassistpro.github.io>*)*

$$= \{a, b, c\}$$

$$L(a \Delta (b \sqcup c)) = L(a) \cup L(b) \sqcup c$$

$$= \{a\} \cup \{b, c\}$$

$$= \{a, b, c\}$$

OPERATORS ON REGEX: UNION $R \mid S$

Definition: $L(R \mid S) = L(R) \cup L(S)$

Assignment Project Exam Help

$$a \mid b)$$

L(<https://eduassistpro.github.io>*)*

$$= \{a, b, c\}$$

$$L(a \Delta (b \sqcup c)) = L(a) \cup L(b \sqcup c)$$

$$= \{a\} \cup \{b, c\}$$

$$= \{a, b, c\}$$

Because it is associative we can write $L(a \mid b \mid c)$

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

=

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

$$= \{abc\}$$

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

$$= \{abc\}$$

Add WeChat edu_assist_pro

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

$$= \{abc\}$$

Add WeChat edu_assist_pro

$$L(a(bc)) = \{rs \mid r \in L(a) \text{ and } s \in L(bc)\}$$

$$= \{rs \mid r \in \{a\} \text{ and } s \in L(bc)\}$$

=

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

$$= \{abc\}$$

Add WeChat edu_assist_pro

$$L(a(bc)) = \{rs \mid r \in L(a) \text{ and } s \in L(bc)\}$$

$$= \{rs \mid r \in \{a\} \text{ and } s \in L(bc)\}$$

$$= \{abc\}$$

OPERATORS ON REGEx: CONCATENATION RS

Definition: $L(RS) = \{rs \mid r \in L(R) \text{ and } s \in L(S)\}$

Assignment Project Exam Help

$$L(ab) = \{rs \mid r \in L(a) \text{ and } s \in L(b)\} = \{ab\}$$

<https://eduassistpro.github.io>

$$= \{abc\}$$

Add WeChat edu_assist_pro

$$L(a(bc)) = \{rs \mid r \in L(a) \text{ and } s \in L(bc)\}$$

$$= \{rs \mid r \in \{a\} \text{ and } s \in L(bc)\}$$

$$= \{abc\}$$

Because it is associative we can write $L(abc)$

UNION AND CONCATENATION

$L((a \mid b)^*) =$

Assignment Project Exam Help

<https://eduassistpro.github.io>

$L((ab) \mid cd) =$

Add WeChat edu_assist_pro

UNION AND CONCATENATION

Assignment Project Exam Help

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

UNION AND CONCATENATION

Assignment Project Exam Help

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

Add WeChat edu_assist_pr

UNION AND CONCATENATION

Assignment Project Exam Help

$$\begin{aligned} L((a+b)c) &= \{rs \mid r \in L(a+b) \text{ and } s \in L(c)\} \\ &= \{rs \mid r \in (\{a\} \cup \{b\}) \text{ and } s \in \{c\}\} \end{aligned}$$

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

UNION AND CONCATENATION

Assignment Project Exam Help

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

Add WeChat edu_assist_pr

UNION AND CONCATENATION

Assignment Project Exam Help

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

Add WeChat `edu_assist_pr`

UNION AND CONCATENATION

Assignment Project Exam Help

$$\begin{aligned} L((a \mid b) \cdot c) &= \{rs \mid r \in L(a \mid b) \text{ and } s \in L(c)\} \\ &= \{rs \mid r \in (\{a\} \cup \{b\}) \text{ and } s \in \{c\}\} \end{aligned}$$

<https://eduassistpro.github.io>

Add WeChat edu_assist_pro

=

UNION AND CONCATENATION

Assignment Project Exam Help

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

$$\begin{aligned}
 L((ab) | c) &= L(ab) \cup L(c) \\
 &= \{rs \mid r \in L(a) \text{ and } s \in L(b)\} \cup \{c\} \\
 &\equiv
 \end{aligned}$$

UNION AND CONCATENATION

Assignment Project Exam Help

$$\begin{aligned} L((a \mid b)c) &= \{rs \mid r \in L(a \mid b) \text{ and } s \in L(c)\} \\ &= \{rs \mid r \in (\{a\} \cup \{b\}) \text{ and } s \in \{c\}\} \end{aligned}$$

<https://eduassistpro.github.io>

Add WeChat `edu_assist_pro`

$$\begin{aligned} L((ab) \mid c) &= L(ab) \cup L(c) \\ &= \{rs \mid r \in L(a) \text{ and } s \in L(b)\} \cup \{c\} \\ &= \{ab\} \cup \{c\} \\ &= \{ab, c\} \end{aligned}$$

OPERATORS ON REGEx: STAR CLOSURE R^*

R^* means 0 or more occurrences of R
 $L(R^*) = \{\epsilon\} \cup L(R) \cup L(RR) \cup L(RR) \cup \dots$

<https://eduassistpro.github.io>

$$L(\emptyset^*) =$$

Add WeChat edu_assist_pro

OPERATORS ON REGEx: STAR CLOSURE R^*

R^* means 0 or more occurrences of R
 $L(R^*) = \{\varepsilon\} \cup L(R) \cup L(RR) \cup L(RRR) \cup \dots$

<https://eduassistpro.github.io>

$$L(\emptyset^*) = \{\varepsilon\} \cup L(\emptyset) \cup \dots$$

$L(a^* b c^*) =$
Add WeChat edu_assist_pro

OPERATORS ON REGEx: STAR CLOSURE R^*

R^* means 0 or more occurrences of R
 $L(R^*) = \{\varepsilon\} \cup L(R) \cup L(RR) \cup L(RR) \cup \dots$

<https://eduassistpro.github.io>

$$L(\emptyset^*) = \{\varepsilon\} \cup L(\emptyset) \cup \dots$$

$L(a \mid bc^*) = \{a, b, b, bc, b\}$
 $L(a \mid (bc)^*) =$

OPERATORS ON REGEx: STAR CLOSURE R^*

R^* means 0 or more occurrences of R
 $L(R^*) = \{\varepsilon\} \cup L(R) \cup L(RR) \cup L(RR) \cup \dots$

<https://eduassistpro.github.io>

$$L(\emptyset^*) = \{\varepsilon\} \cup L(\emptyset) \cup \dots$$

Add WeChat edu_assist_pro

$$L(a \mid bc^*) = \{a, b, \varepsilon, bc, bcbc, \dots\}$$

$$L((a \mid b)c^*) = \{a, b, \varepsilon, ac, ac, \dots\}$$

OPERATORS ON REGEx: STAR CLOSURE R^*

R^* means 0 or more occurrences of R
 $L(R^*) = \{\varepsilon\} \cup L(R) \cup L(RR) \cup L(RRR) \cup \dots$

<https://eduassistpro.github.io>

$$L(\emptyset^*) = \{\varepsilon\} \cup L(\emptyset) \cup \dots$$

Add WeChat edu_assist_pro

$$L(a \mid bc^*) = \{a, b, \varepsilon, bc, bcc, \dots\}$$

$$L((a \mid b)c^*) = \{a, b, ac, bc, acc, bcc, accc, bccc, \dots\}$$

SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions

Assignment Project Exam Help

► Union properties:

► $R | S = S | R$ (commutative)

<https://eduassistpro.github.io>

► Concatenation properties:

Add WeChat edu_assist_pro

► Union and concatenation are *distributive* when combined:

SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions

Assignment Project Exam Help

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

► Concatenation properties:

$$\blacktriangleright R\emptyset = \emptyset R = \emptyset$$

$$\text{Add} = \varepsilon R = W$$

► $(RS)T = R(ST)$ (associative)

- Union and concatenation are *distributive* when combined:

SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions

Assignment Project Exam Help

► Union properties:

► $R | S = S | R$ (commutative)

<https://eduassistpro.github.io>

► Concatenation properties:

► $R\emptyset = \emptyset R = \emptyset$

► $R\varepsilon = \varepsilon R = R$

► $(RS)T = R(ST)$ (associative)

Add WeChat edu_assist_pro

► Union and concatenation are *distributive* when combined:

► $R(S | T) = RS | RT$

► $(S | T)R = SR | TR$

SOME PROPERTIES OF REGEx

Assignment Project Exam Help

► <https://eduassistpro.github.io>

Add WeChat edu_assist_pr

SOME PROPERTIES OF REGEx

Assignment Project Exam Help

► <https://eduassistpro.github.io>

Add WeChat edu_assist_pr

SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions. Then

Assignment Project Exam Help

-

<https://eduassistpro.github.io>

$$\blacktriangleright \quad R^\star = \varepsilon \mid R^\star = (\varepsilon \mid R)^\star = (\varepsilon \mid \quad \quad \quad \star$$

Add WeChat edu_assist_pr

SOME PROPERTIES OF REGEx

Assignment Project Exam Help

- ▶

<https://eduassistpro.github.io>

- $R^* = \varepsilon \mid R^* = (\varepsilon \mid R)^* = (\varepsilon \mid \dots)^*$
 - ~~$RR^* = R^*R$~~

Add WeChat edu_assist_pr

SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions. Then

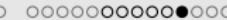
Assignment Project Exam Help

- ▶

<https://eduassistpro.github.io>

- $R^\star = \varepsilon \mid R^\star = (\varepsilon \mid R)^\star = (\varepsilon \mid \quad \quad \quad)^\star$
 - $R R^\star = R^\star R$

Add WeChat[®] edu_assist_pr



SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions

Assignment Project Exam Help



<https://eduassistpro.github.io>

- ▶ $R^* = \varepsilon \mid R^* = (\varepsilon \mid R)^* = (\varepsilon \mid \dots)^*$
- ▶ $RR^* = R^*R$
- ▶ $(T \mid S)^* = (R^* \mid S^*)^* = (RS)^*$
- ▶ $R(SR)^* = (RS)^*R$

Add WeChat `edu_assist_pro`

SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions.

Assignment Project Exam Help

-

<https://eduassistpro.github.io>

- $R^* = \varepsilon \mid R^* = (\varepsilon \mid R)^* = (\varepsilon \mid R)^*$
 - $RR^* = R^*R$
 - $(R \mid S)^* = R^* \mid (S^*)^* = (RS)^*$
 - $R(SR)^* = (RS)^*R$
 - $(R^*S)^* = \varepsilon \mid (R \mid S)^*S$

► Add WeChat edu_assist_pr

SOME PROPERTIES OF REGEx

Let R, S, T be regular expressions. Then

Assignment Project Exam Help

- ▶

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

- $R^* = \varepsilon \mid R^* = (\varepsilon \mid R)^* = (\varepsilon \mid R)^*$
 - $RR^* = R^*R$
 - $(T^+)^* = (R^+ + S^+)^* = (RS)^*$
 - $R(SR)^* = (RS)^*R$
 - $(R^*S)^* = \varepsilon \mid (R \mid S)^*S$
 - $(RS^*)^* = \varepsilon \mid R(R \mid S)^*$

► $R^H = R$, $R^{\star} = (R^H)^{\star} = R^{\star H}$
► $(F^{-1}S)^{\star} = (R^{\star H} + S^{\star})^{\star} = (R^{\star}S^{\star})^{\star}$
► $B(SB)^{\star} = (BS)^{\star}B$

EXAMPLES OF REGEx

All possible strings which can be formed with a , b , or c

► <https://eduassistpro.github.io/>

► Traffic lights?

Add WeChat edu_assist_pro

► Identifiers for Java programs?

EXAMPLES OF REGEx

Assignment Project Exam Help

- ▶ All possible strings which can be formed with a , b , or c
 $(a|b|c)^*$

- ▶ <https://eduassistpro.github.io>

- ▶ Traffic lights?

Add WeChat edu_assist_pro

- ▶ Identifiers for Java programs?

EXAMPLES OF REGEx

Assignment Project Exam Help

- ▶ All possible strings which can be formed with a , b , or c
 $(a|b|c)^*$

- ▶ <https://eduassistpro.github.io>

- ▶ Traffic lights?

Add WeChat edu_assist_pro

- ▶ Identifiers for Java programs?

EXAMPLES OF REGEx

Assignment Project Exam Help

- ▶ All possible strings which can be formed with a , b , or c
 $(a|b|c)^*$

- ▶ <https://eduassistpro.github.io>

- ▶ Traffic lights?
 $(red|yellow|green)^*$
- ▶ Add WeChat edu_assist_pro
- ▶ Identifiers for Java programs?

EXAMPLES OF REGEx

Assignment Project Exam Help

- ▶ All possible strings which can be formed with a , b , or c
 $(a|b|c)^*$

▶ <https://eduassistpro.github.io>

- ▶ Traffic lights?

(red | yellow | amber | green)

Add WeChat edu_assist_pro

- ▶ Identifiers for Java programs?

$(letter \mid \$ \mid _) (letter \mid digit \mid \$ \mid _)^*$

EXAMPLES OF REGEx

► a^*ba^* represents Assignment Project Exam Help



<https://eduassistpro.github.io>

► $ab^*a \mid ba^*b \mid a \mid b$ represents

Add WeChat edu_assist_pro

► $a^*\emptyset$ represents

EXAMPLES OF REGEx

► a^*ba^* represents strings over $\{a, b\}$ with exactly one b



<https://eduassistpro.github.io>

► $ab^*a \mid ba^*b \mid a \mid b$ represents

Add WeChat edu_assist_pro

► $a^*\emptyset$ represents

EXAMPLES OF REGEx

► a^*ba^* represents strings over $\{a, b\}$ with exactly one b



<https://eduassistpro.github.io>

► $ab^*a \mid ba^*b \mid a \mid b$ represents

Add WeChat edu_assist_pro

► $a^*\emptyset$ represents

EXAMPLES OF REGEx

► a^*ba^* represents
Strings over $\{a, b\}$ with exactly one b



<https://eduassistpro.github.io>

► $ab^*a \mid ba^*b \mid a \mid b$ represents

Strings of a 's starting and ending with
starting and ending with a , or a single

Add WeChat `edu_assist_pro`

► $a^*\emptyset$ represents

EXAMPLES OF REGEx

► a^*ba^* represents
Strings over $\{a, b\}$ with exactly one b



<https://eduassistpro.github.io>

► $ab^*a \mid ba^*b \mid a \mid b$ represents

Strings of a 's starting and ending with
starting and ending with a , or a single

Add WeChat edu_assist_pro

► $a^*\emptyset$ represents

The empty language, \emptyset

EQUIVALENCE OF REGEx AND FA

Theorem:

A language is regular if and only if a regular expression describes it.

Proof

Sho

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

recognises the same language

- ## 2. FA \Rightarrow RegEx

Show that for each NFA, there exists a RegEx

recognises the same language

EQUIVALENCE OF REGEx AND FA

Theorem:

A language is regular if and only if a regular expression describes it.

Proof

Show

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

recognises the same language

2. $FA \Rightarrow RegEx$.

Show that for each NFA, there exists a RegEx

recognises the same language

Add WeChat edu_assist_pro

Next week: proof

APPLICATION: PATTERN MATCHING

Assignment Project Exam Help

- ▶ Provides a technique for finding occurrences of patterns in
- ▶ <https://eduassistpro.github.io/>
- ▶ pattern is trivial.
- ▶ Then we convert the NFA to DFA, then mini pattern can be matched efficiently

DFAs AND NFAs

Assignment Project Exam Help

- ▶ NFAs and DFAs define the class of regular languages
 - ▶ Each NFA can be transformed into a unique minimal DFA
 - ▶
 - ▶ <https://eduassistpro.github.io/>
 - ▶ despite the fact they often have far more states than DFAs
 - ▶ Regular languages are closed under the union, intersection, and complement operations
 - ▶ We use these to build the NFA, before converting it to a DFA
- Add WeChat edu_assist_pro

REGULAR EXPRESSIONS

Assignment Project Exam Help

- ▶ <https://eduassistpro.github.io>
- ▶ Equivalence with DFA/NFA

Add WeChat edu_assist_pro