

# 程序代写代做 CS编程辅导

Introduction



cs2001

Theory of Computation

Lecture 10

WeChat: cstutorcs

Ruth Urner  
Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

Febuary 8, 2023

<https://tutorcs.com>

程序代写代做 CS编程辅导



# Non-deterministic finite automata

WeChat: cstutorcs

Reading: ITC Section 1.2  
Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

# DFA versus NFAs

程序代写代做 CS编程辅导

To show that regular languages are also closed under the concatenation and star operations, it is convenient to employ the concept of a Non-deterministic Finite Automaton (NFA for short).



In the automata we have seen so far, every state has exactly one outgoing edge for every symbol. That is, for every state, the "reaction" to any input symbol is uniquely determined. Such an automaton is also called a Deterministic Finite Automaton (DFA for short).

QQ: 749389476

We will generalize this to also allow non-unique (think randomized or parallelized) computations. NFAs are a model for such computations.

<https://tutorcs.com>

# Example: the NFA $N_1$

程序代写代做 CS编程辅导



WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

For the machine  $N_1$  we get

QQ: 749389476

$L(N_1) = \{w \mid w \text{ contains } 11 \text{ or } 101 \text{ as a substring}\}$

<https://tutorcs.com>

# 程序代写代做 CS 编程辅导

Example: the NFA  $N_1$

Differences



ε can be "read"

same states  
have several  
instructions  
for the same  
letter

WeChat: cstutorcs

source states  
Assignment Project Exam Help  
for all letters

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

Talking: Ruth Umer

# 程序代写代做 CS 编程辅导

Example: the NFA  $N_1$

Differences



E can be "read"

{ Examples or words

10110 ✓  
accepted

1000 X  
not accepted

WeChat: cstutorcs

source states  
for all letters  
Assignment Project Exam Help

Email: tutorcs@163.com

A word is accepted by an NFA if  
there is at least one path reading  
the word that leads to an accept state.

QQ: 749389476

<https://tutorcs.com>

# 程序代写代做 CS 编程辅导

Example: the NFA  $N_1$

Differences



$\epsilon$  can be "read"

$\epsilon$



{ Examples or words

10110 ✓  
accepted

1000 X  
not accepted

WeChat: cstutorcs

source states  
for all letters

Email: tutorcs@163.com

$$(\mid \mid \mid \mid = \mid \epsilon \mid \mid)$$

same states  
have several  
instructions  
for the same  
letter

A word is accepted by an NFA if  
there is at least one path reading  
the word that leads to an accept state.

QQ: 749389476

<https://tutorcs.com>

程序代写代做 CS 编程辅导

Example: the NFA / M

The diagram shows a Non-deterministic Finite Automaton (NFA) with four states:  $q_1$ ,  $q_2$ ,  $q_3$ , and  $q_4$ . State  $q_1$  is the start state, indicated by an incoming arrow. Transitions are as follows:  $q_1 \xrightarrow{0,1} q_2$ ,  $q_2 \xrightarrow{\epsilon} q_3$ ,  $q_2 \xrightarrow{1} q_4$ ,  $q_3 \xrightarrow{0,1} q_4$ , and  $q_4 \xrightarrow{1} q_3$ .

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Consider word: leave

the word is accepted since leave exist branches leading to accept states

A computation stops

# 程序代写代做 CS 编程辅导

Example: the NFA  $N_1$

10001  
not accepted



WeChat: cstutorcs

## Assignment Project Exam Help

For the machine  $N_1$  we get

Email: [tutorcs@163.com](mailto:tutorcs@163.com)  
(In this case, 10001 contains 101 or 11 as a substring)

QQ: 749389476

<https://tutorcs.com>

Reminder:  
For alphabet  
 $\Sigma$ , the set  
of all words  
over  $\Sigma$  is  
denoted by  
 $\Sigma^*$ .

Talking: Ruth Urner

# Formal definition of a non-deterministic finite automaton

程序代写代做 CS编程辅导

For an alphabet  $\Sigma$ , we have  $\Sigma_\epsilon = \Sigma \cup \{\epsilon\}$ .



## Definition

A non-deterministic finite automaton is a 5-tuple  $(Q, \Sigma, \delta, q_0, F)$ , where

WeChat: cstutorcs

Assignment Project Exam Help

1.  $Q$  is a finite set called the set of states
2.  $\Sigma$  is a finite set called the alphabet,
3.  $\delta : Q \times \Sigma_\epsilon \rightarrow \mathcal{P}(Q)$  is the transition function,
4.  $q_0 \in Q$  is the start state, and
5.  $F \subseteq Q$  is the set of accept states.

Email: tutorcs@163.com

QQ: 749389476

<https://tutors.com>

# 程序代写代做 CS 编程辅导

## Formal description of the NFA $N_1$



Formal description of  $N_1 = (Q, \Sigma, \delta, q_0, F)$

Q = {q<sub>1</sub>, q<sub>2</sub>, ..., q<sub>n</sub>}, WeChat: cstutorcs, q<sub>0</sub> = q<sub>1</sub>, F = {q<sub>d</sub>}.

$$\Sigma_2 = \{0, 1, \varepsilon\}$$

~~S | O | I~~ Assignment Project Exam Help

$\overline{q_1}$  | {q,} {q, q} 

9.  $\{9, 3\}$  Email  $\{9, 3\}$

**Eiffel**

$$q_3 \mid \emptyset \text{ } (14) \text{ } \emptyset$$

$$q_4 \mid \{q_4\} \quad \{14\} : \textcolor{red}{0} : \textcolor{blue}{7}$$

Email: tutorcs@163.com

$\{9_1\} \quad \emptyset \quad \{1, 2, 3, 4\} \quad \{5\}$

00-749389476

com These are the same word!!

<https://tutorcs.com>

# Formal description of the NFA $N_1$



WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

For the machine  $N_1$  we get

$$L(N_1) = \{w \mid w \text{ contains } 11 \text{ or } 101 \text{ as a substring}\}$$

# Formal definition of acceptance for NFAs

程序代写代做 CS编程辅导

For a string  $w = w_1 w_2 \dots w_n$  where  $w_i \in \Sigma$  for some alphabet  $\Sigma$ , we let strings  $z_0 w_1 z_1 w_2 z_2 \dots z_{n-1} w_n z_n$  where each  $z_i$  is a sequence of 0 or more symbols  $\epsilon$  represent the string  $w$  as a string over  $\Sigma$ .



## Definition

Let  $N = (Q, \Sigma, \delta, q_0, F)$  a non-deterministic finite automaton (NFA) and let  $w$  be a string over  $\Sigma$ . We say that  $N$  accepts  $w$  if we can write  $w = y_1 y_2 \dots y_m$  with  $y_i \in \Sigma_\epsilon$  and there exists a sequence  $s_0, s_1, \dots, s_m$  of states such that

1.  $s_0 = q_0$ , Email: [tutorcs@163.com](mailto:tutorcs@163.com)
2.  $s_{i+1} \in \delta(s_i, y_{i+1})$  for  $i = 0, 1, \dots, m$ , and QQ: 749389476
3.  $s_m \in F$ .

<https://tutorcs.com>

We then define the language  $L(N)$  recognized by NFA  $N$  in the same way as for DFAs, namely as the set of all words that are accepted by  $N$ .

# 程序代写代做 CS编程辅导

Formal definition of acceptance for NFAs

$|w| = \omega$   
 $|\Sigma| = \epsilon$   
 $\{\epsilon\} \subset \Sigma$   
 $w \in \Sigma^*$

For a string  $w = z_0 w_1 z_1 w_2 z_2 \dots z_n w_n z_n$  where each  $z_i$  is a sequence of 0 or more symbols  $\epsilon$  representing some word in  $\Sigma^*$ .



## Definition

Let  $N = (Q, \Sigma, \delta, q_0, F)$  a non-deterministic finite automaton (NFA) and let  $w$  be a string over  $\Sigma$ . We say that  $N$  accepts  $w$  if we can write  $w = y_1 y_2 \dots y_m$  with  $y_i \in \Sigma^*$  and there exists a sequence  $s_0 s_1 s_2 \dots s_m$  of states such that

1.  $s_0 = q_0$ ,
2.  $s_{i+1} \in \delta(s_i, y_{i+1})$  for  $i = 0, 1, \dots, m$ , and
3.  $s_m \in F$ .

**Assignment Project Exam Help**

**Email: tutorcs@163.com**

**QQ: 749389476**

**<https://tutorcs.com>**

# Example: the NFA $N_2$

程序代写代做 CS编程辅导



WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

For the machine  $N_2$  we get

QQ: 749389476

$L(N_2) = \{w \mid w \text{ contains } 1 \text{ at the third position from the end}\}$

<https://tutorcs.com>

# 程序代写代做 CS编程辅导

Example: the NFA  $N_2$



WeChat: cstutorcs  
0101 ✓ 0001 ✗  
1 1 1 1 ✗ 0 0 0 0 1 ✗  
Assignment Project Exam Help

For the machine  $N_2$  we get

Email: tutorcs@163.com

QQ: 749389476  
 $= \{ w = w_1 w_2 \dots w_n \in \Sigma^* \mid w_{n-2} = 1 \}$  at the third position from the end }

<https://tutorcs.com>

# Example: the NFA $N_3$

Let's consider a unary alphabet  $\Sigma = \{0\}$ .

程序代写代做 CS编程辅导



For the machine  $N_3$  we get

$$L(N_3) = \{w \mid \text{The number of 0 symbols in } w \text{ is divisible by 2 or by 3}\}$$

# 程序代写代做 CS编程辅导

Example: the NFA

Let's consider



abet  $\Sigma = \{0\}$ .



$\Sigma$  ✓

00000 X

00 ✓

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

$L(N_1) = \{\omega \in \{0\}^* \mid \text{the length of } \omega \text{ is divisible by 2 or by 3}\}$

<https://tutorcs.com>

Example: 程序代写代做CS编程辅导

Let's consider



abet  $\Sigma = \{0\}$ .

$\epsilon$  ✓

00000 X

00 ✓

is the same word  
as ε00

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

$L(N_1) = \{w \in \{0\}^* \mid \text{the length of } w \text{ is divisible by 2 or by 3}\}$   
 $= \{w \in \{0\}^* \mid \text{length of } w \text{ even}\} \cup \{w \in \{0\}^* \mid \text{length of } w \text{ divisible by 3}\}$

QQ: 749389476

<https://tutorcs.com>

# Equivalence of NFAs and DFAs

程序代写代做 CS编程辅导

## Definition

We call two finite state machines equivalent if they recognize the same language.



WeChat: cstutorcs

## Theorem

Assignment Project Exam Help  
For every NFA, there exists an equivalent DFA.

Email: tutorcs@163.com

## Corrolary

QQ: 749389476

A language  $L$  is regular if and only if there exists an NFA that recognizes it.  
<https://tutorcs.com>

程序代写代做CS编程辅导

Equival

House Select Text Draw Stamp Spotlight Eraser Format Undo Redo Clear Save



### Definition

We call two machines **equivalent** if they recognize the same language.

WeChat: cstutorcs

### Theorem

For every NFA, there exists an equivalent DFA.

Assignment Project Exam Help

For every DFA  $N$ , there exists a DFA  
 $M$  with  $L(M) = L(N)$ .

Corollary QQ: 749389476

A language  $L$  is regular, if and only if there  
is a DFA that recognizes  $L$ .

<https://tutorcs.com>

# Transforming an NFA into a DFA—example

程序代写代做 CS编程辅导



To prove the theorem from the previous slide, we need to show that for every NFA  $N$ , there exists a DFA  $M$  with  $L(M) = L(N)$ .

WeChat: cstutorcs

Assignment Project Exam Help  
Before we see the general proof for this, we go through one concrete example. We transform NFA  $N_4$  into an equivalent DFA.  
Email: tutorcs@163.com

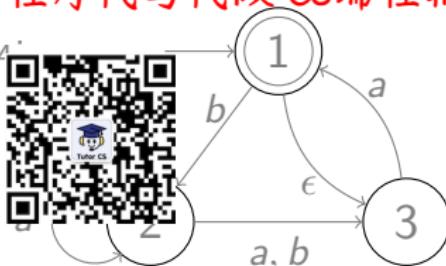
QQ: 749389476

<https://tutorcs.com>

# Transforming an NFA into a DFA—example

程序代写代做 CS编程辅导

Consider the NFA  $N_1$ :



WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

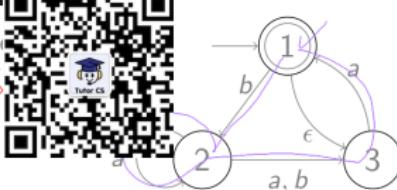
<https://tutorcs.com>

# 程序代写代做CS编程辅导



Consider the

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



Examples of  
accepted words

$\epsilon$  ✓  
a ✓  
" " ✓  
 $\Sigma a$  ✓

baba ✓

Examples of  
not accepted:

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com bbb X

QQ: 749389476 bb X

<https://tutorcs.com>

Talking: Ruth Umer

# 程序代写代做CS编程辅导

Transforming an NFA into a DFA-example

Consider the NFA  $\Sigma = \{a, b\}$

QR code: TutorCS

$Q = \{1, 2, 3\}$  Examples of accepted words

Diagram of NFA:

```
graph LR; 1((1)) -- a --> 2((2)); 1 -- b --> 3((3)); 2 -- a --> 3; 2 -- b --> 1; 3 -- a --> 1; 3 -- b --> 2; 3 -- ε --> 1;
```

Equivalent DFA:

WeChat: cstutorcs

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Diagram of DFA:

```
graph LR; S((∅)) -- "a,b" --> 11((1,2)); S -- b --> 33((3)); 11 -- a --> 13((1,3)); 11 -- b --> 22((2,3)); 13 -- a --> 11; 13 -- b --> 22; 22 -- a --> 13; 22 -- b --> 33; 33 -- a --> 11; 33 -- b --> 22;
```

Examples of accepted words: ε, a, " ", εa, baba

Examples of not accepted words: bbb, bb

# 程序代写代做 CS 编程辅导

Transforming an NFA into a DFA-example

Consider the NFA  $\Sigma = \{a, b\}$

$Q = \{1, 2, 3\}$  Examples of accepted words

Equivalent DFA

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Diagram of the NFA:

```
graph LR; S(( )) -- "a,b" --> 1((1)); 1 -- "a" --> 1; 1 -- "b" --> 2((2)); 2 -- "a,b" --> 3((3)); 3 -- "a" --> 1; 3 -- "epsilon" --> 2;
```

Diagram of the DFA:

```
graph LR; S(( )) -- "a,b" --> 1(( )); 1 -- "a" --> 1; 1 -- "b" --> 2(( )); 2 -- "a" --> 3(( )); 2 -- "b" --> 2; 3 -- "a" --> 1; 3 -- "b" --> 2;
```

Accepted words (green): ε, a, " ", εa, baba

Not accepted words (red): bbb, bb

# General proof of Theorem

程序代写代做 CS编程辅导

Proof

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



A.

We need to construct a DFA  $M = (Q', \Sigma, \delta', q'_0, F')$  with  $L(M) = L(N)$ .

Assuming that  $N$  does not contain any  $\epsilon$ -transitions, we construct  $M$  as follows:

1.  $Q' = \mathcal{P}(Q)$

WeChat: cstutorcs

2. For  $R \in Q'$  and  $a \in \Sigma$ , we let

Assignment Project Exam Help

$$\delta'(R, a) = \{q \in Q \mid q \in \delta(r, a) \text{ for some } r \in R\}$$

QQ: 749389476

3.  $q'_0 = \{q_0\}$

Email: tutorcs@163.com

4.  $F' = \{R \in Q' \mid \text{there exists an } r \in F \text{ with } r \in R\}.$

This completes the construction for NFAs that don't contain  $\epsilon$ -transitions.

# 程序代写代做 CS编程辅导



## Proof

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



We need to construct an NFA  $M = (Q', \Sigma, \delta', q'_0, F')$  with  $L(M) = L(N)$ .

Assuming that  $N$  does not contain any  $\epsilon$ -transitions, we construct  $M$  as follows:

1.  $Q' = \mathcal{P}(Q)$

**WeChat: cstutorcs**  $\in Q' = \mathcal{P}(Q)$

2. For  $R \in Q'$  and  $a \in \Sigma$ , we let

$\delta'(R, a) = \{q \mid \exists p \in R, \delta(p, a) = q\}$  for every  $a \in \Sigma$

**Assignment Project Exam Help**

3.  $q'_0 = \{q_0\}$  **Email: tutorcs@163.com**

4.  $F' = \{R \in Q' \mid \text{there exists an } r \in F \text{ with } r \in R\}$ .

**QQ: 749389476**

This completes the construction for NFAs that don't contain  $\epsilon$ -transitions.

<https://tutorcs.com>

# Proof of Theorem

程序代写代做 CS编程辅导

Proof continued

If  $N$  does contain  $\epsilon$ -transi-



xtend the above construction as follows:

For a state  $R \in Q'$ , we let  $E(R)$  denote the set of states that can be reached via 0 or more  $\epsilon$ -transitions from some state  $r \in R$  (in  $N$ ).

WeChat: cstutorcs

Then we modify the transition function of  $M$  to be

Assignment Project Exam Help

$$\delta'(R, a) = \{q \in Q \mid q \in E(\delta(r, a)) \text{ for some } r \in R\}$$

Email: tutorcs@163.com

and the start state of  $M$  to be

QQ: 749389476

$$q'_0 = E(q_0).$$

https://tutorcs.com

# Regular languages are closed under regular operations

程序代写代做 CS编程辅导

## Theorem

Regular languages are closed under unions.

"Proof" by illustration



Given A and B are NFA's  $N_A$  and  $N_B$  with  
 $L(N_A) = A$  and  $L(N_B) = B$ . We need to construct an  
NFA  $N$  that recognizes  $A \cup B$ .

WeChat: cstutorcs

Assignment Project Exam Help

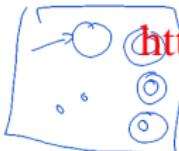
$N_A$



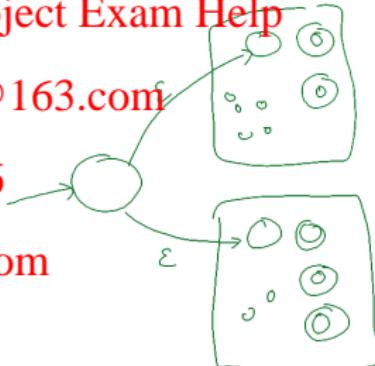
Email: tutorcs@163.com

QQ: 749389476

$N_B$



<https://tutorcs.com>



# Regular languages are closed under regular operations

程序代写代做 CS编程辅导

## Theorem

Regular languages are  under unions.

## Proof (formal version)



We need to show that if two languages  $A$  and  $B$  are regular, then so is  $A \cup B$ .

If  $A$  and  $B$  are regular, then there exist NFAs  $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$  and  $N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$  such that  $L(N_1) = A$  and  $L(N_2) = B$ . We need to show that there exists an NFA  $M$  with  $L(M) = A \cup B$ . The picture on the previous slide illustrates a construction for  $M = (Q, \Sigma, \delta, q_0, F)$ . Formally, we set:

WeChat: cstutorcs  
Assignment Project Exam Help  
Email: tutorcs@163.com

1.  $Q = Q_1 \cup Q_2 \cup \{q_0\}$ , where  $q_0$  is a new state, that is  $q_0 \notin Q_1 \cup Q_2$  and this state  $q_0$  will then also be the start state of  $M$ .

2.  $\delta(q, a) = \begin{cases} \{q_1, q_2\} & \text{if } a = \epsilon \\ \delta_i(q, a) & \text{if } q \in Q_i \\ \emptyset & \text{else} \end{cases}$

<https://tutorcs.com>

3.  $F = F_1 \cup F_2$ .

# Regular languages are closed under regular operations

## Theorem

程序代写代做 CS编程辅导

Regular languages are closed under concatenations.

“Proof” by illustration

Intuition for



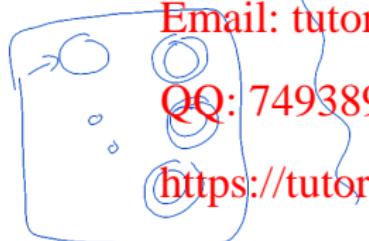
$$L(\mathcal{D}_A) = A$$



WeChat: cstutorcs

Assignment Project Exam Help

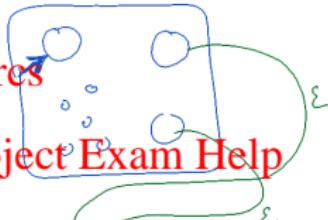
$$L(\mathcal{D}_B) = B$$



Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



$$L(\mathcal{D}) = A \circ B$$

# Regular languages are closed under regular operations

## Theorem

程序代写代做 CS编程辅导

Regular languages are closed under concatenations.



## Proof (formal version)

We need to show that if two languages  $A$  and  $B$  are regular, then so is  $A \circ B$ .

If  $A$  and  $B$  are regular, then there exist NFAs  $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$  and  $N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$  such that  $L(N_1) = A$  and  $L(N_2) = B$ . We need to show that there exists an NFA  $M$  with  $L(M) = A \circ B$ . The picture in the previous slide illustrates a construction for  $M = (Q, \Sigma, \delta, q_0, F)$ . Formally, we set:

Email: tutorcs@163.com

1.  $Q = Q_1 \cup Q_2$

QQ: 749389476

2.  $q_0 = q_1$

3.  $\delta(q, a) = \begin{cases} \delta_1(q, a) & \text{if } q \in Q_1 \text{ and } (q \notin F_1 \text{ or } a = \epsilon) \\ \delta_2(q, a) & \text{if } q \in Q_2 \end{cases}$

4.  $F = F_2$ .

# Regular languages are closed under regular operations

程序代写代做 CS编程辅导

## Theorem

Regular languages are closed under the star-operation.

“Proof” by illustration



# Regular languages are closed under regular operations

程序代写代做 CS编程辅导

## Theorem

Regular languages are closed under the star-operation.



## Proof (formal version)

We need to show that if a language  $A$  is regular, then so is  $A^*$ . If  $A$  is regular, then there exist an NFA  $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$  such that  $L(N_1) = A$ . We need to show that there exists an NFA  $M$  with  $L(M) = A^*$ . The picture on the previous slide illustrates a construction for  $M = (Q, \Sigma, \delta, q_0, F)$ . We set:

Email: tutorcs@163.com

QQ: 749389476

1.  $Q = Q_1 \cup \{q_0\}$ , where  $q_0$  is a new state, that is  $q_0 \notin Q_1 \cup Q_2$  and this state  $q_0$  will then also be the start state of  $M$ .
2.  $\delta(q, a) = \begin{cases} \{q_1\} & \text{if } q = q_0 \text{ and } a = \epsilon \\ \delta_1(q, a) & \text{if } q \in Q_1 \text{ and } (q \notin F_1 \text{ or } a \neq \epsilon) \\ \delta_1(q, a) \cup \{q_1\} & \text{if } q \in F_1 \text{ and } a = \epsilon \end{cases}$
3.  $F = F_1 \cup \{q_0\}$ .