Principles of Compiler

Chapter 3

Lexical Analysis

Example

int x=10,y=20,sum;

Token Class	Lexeme
Keyword	int
Identifier	x
Operator	=
Number	10
Separator	,
Identifier	y
Operator	=
Number	20
Separator	,
Identifier	sum
Separator	;

Example

优先级

*

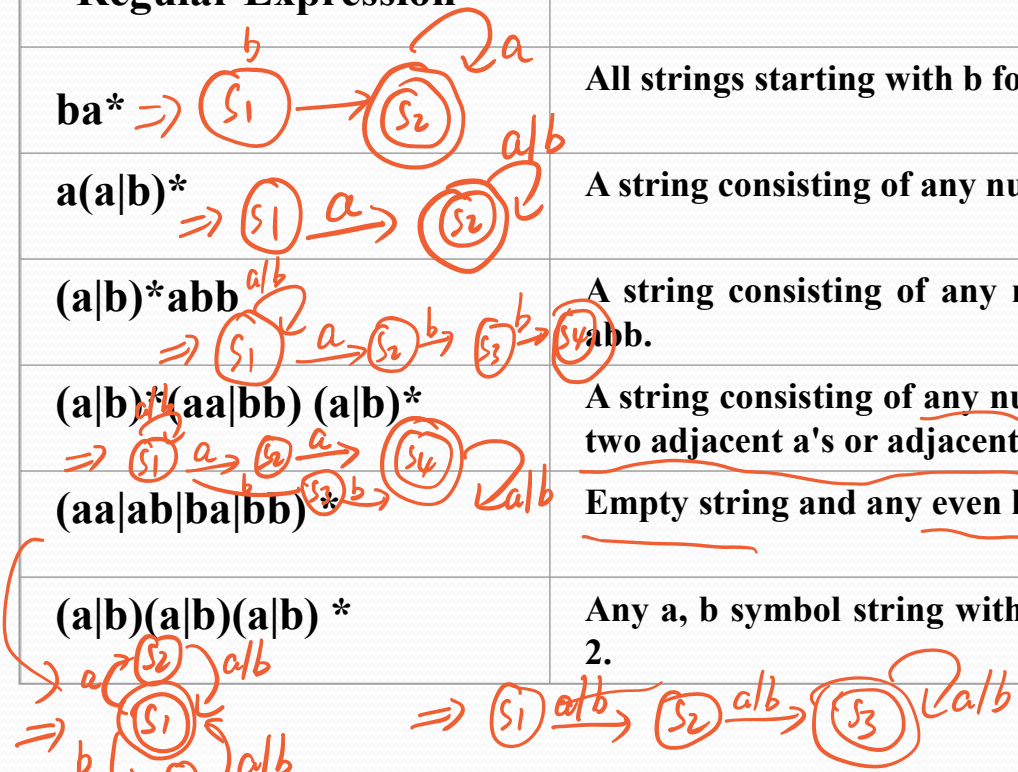
|

↓

描述

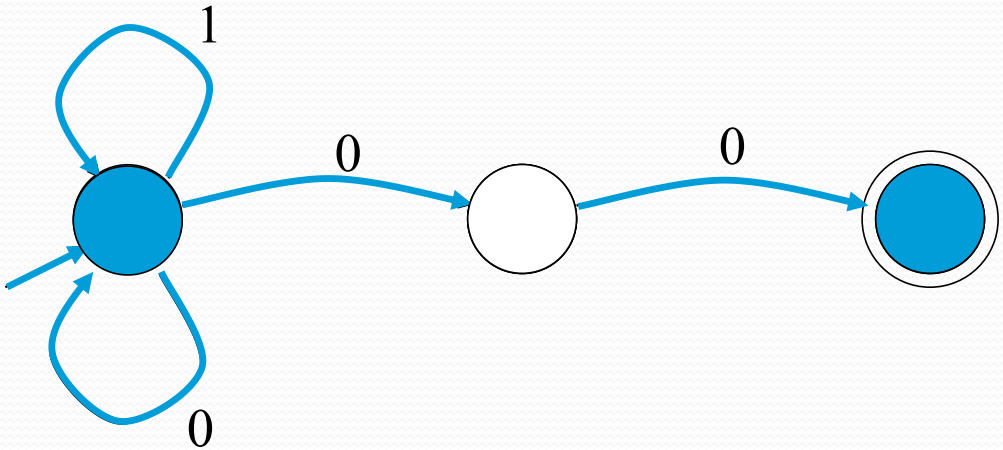
正规式转换自动机并确定化

Regular Expression	Formal Language
ba*	All strings starting with b followed by any number of a.
a(a|b)*	A string consisting of any number of a and b starting with a.
(a|b)*abb	A string consisting of any number of a and b ending with abb.
(a|b)*(aa|bb) (a|b)*	A string consisting of any number of a and b and containing two adjacent a's or adjacent b's.
(aa|ab|ba|bb)*	Empty string and any even length a, b symbol string.
(a|b)(a|b)(a|b) *	Any a, b symbol string with length greater than or equal to 2.



Acceptance of NFAs

- An NFA can get into multiple states



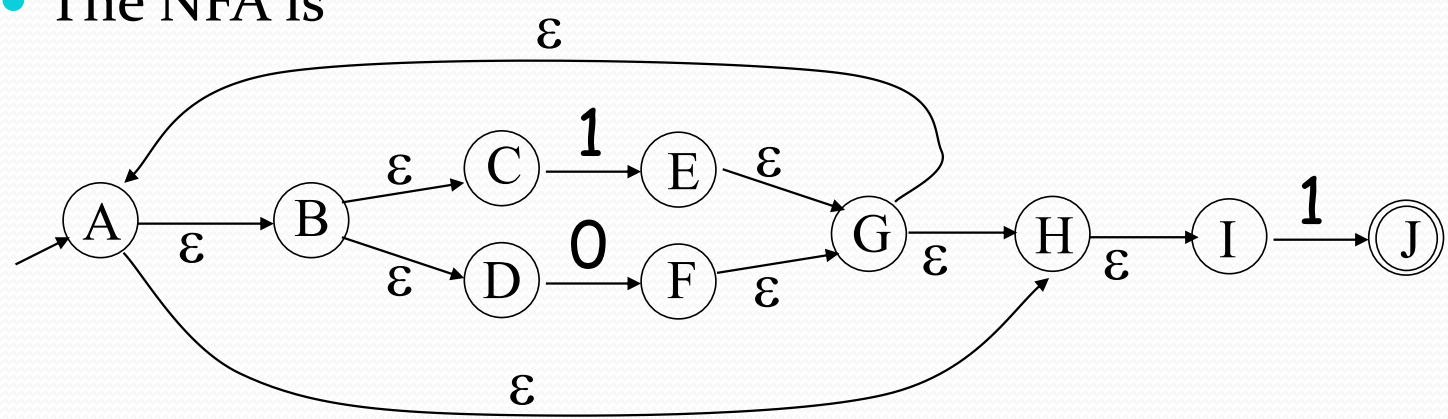
- Input: 1 0 0
- State: {A} {A,B} {A,B,C}
- Rule: NFA accepts if it <u>can</u> get in a final state

Example of RegExp -> NFA conversion

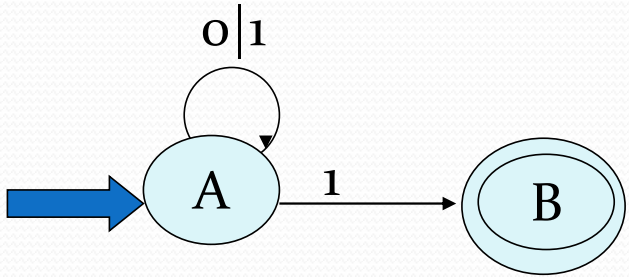
- Consider the regular expression

$$(1 \mid 0)^*1$$

- The NFA is



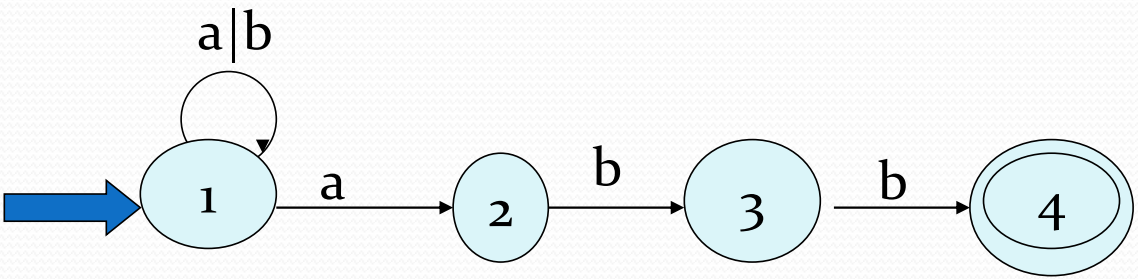
- Simple NFA



Example:

RegExp R=(a|b)* abb -> NFA N

L(N)=L(R)

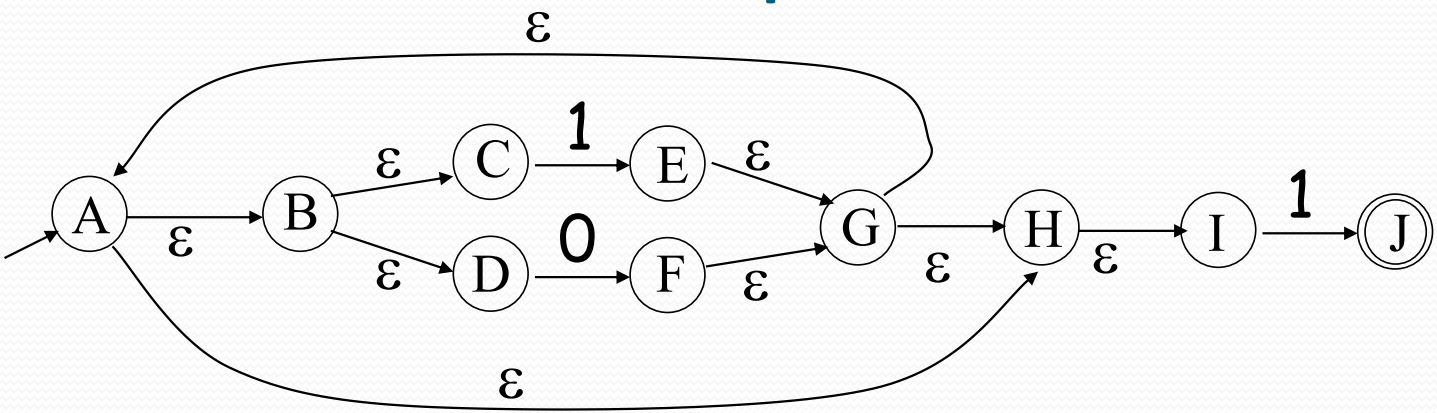


Practice: RegExp R -> NFA N

- Correct decomposition according to priority

  - RegExp R=(0|1)* 101
  - RegExp R=a((a|b)*|ab*a) *b
  - RegExp R=b((ab) *|bb) *ab

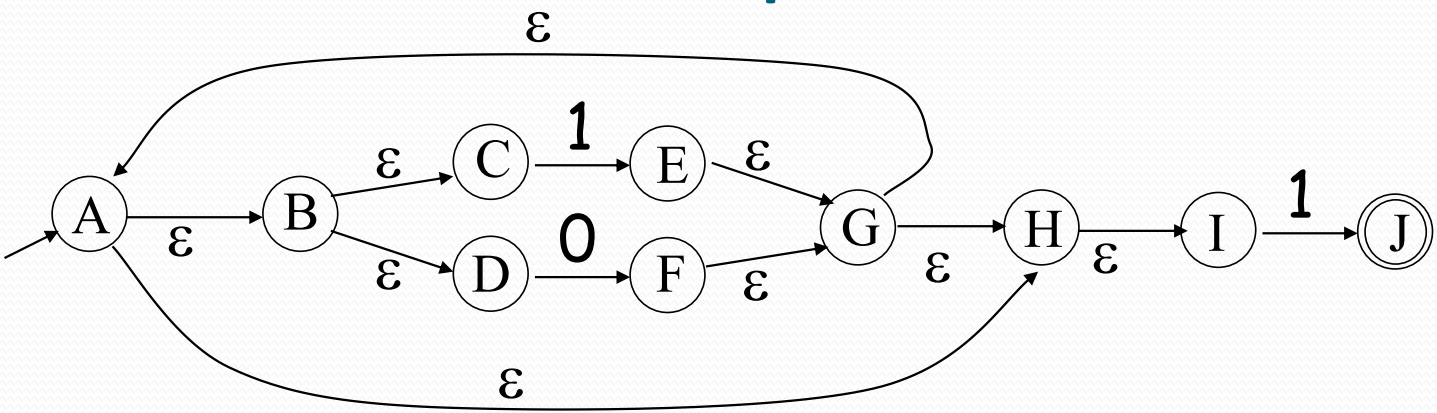
NFA -> DFA Example



ε-closure(B)={B,C,D}

ε-closure(G)={A,B,C,D,G,H,I}

NFA -> DFA Example



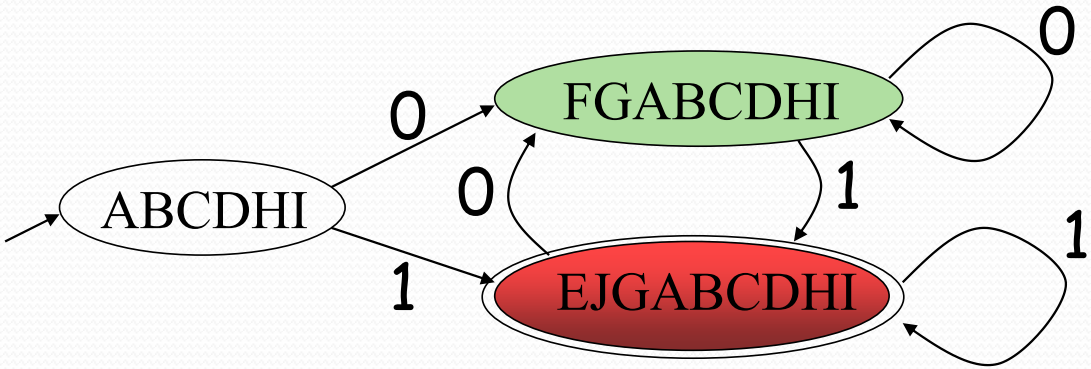
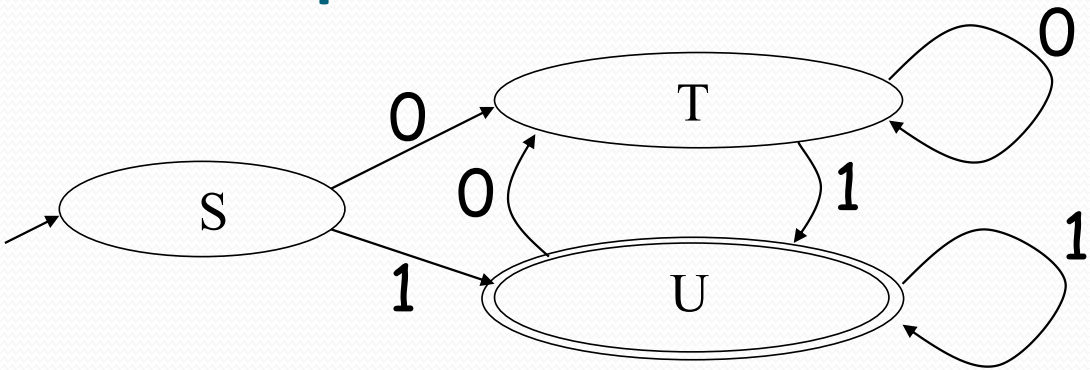


Table Implementation of a DFA

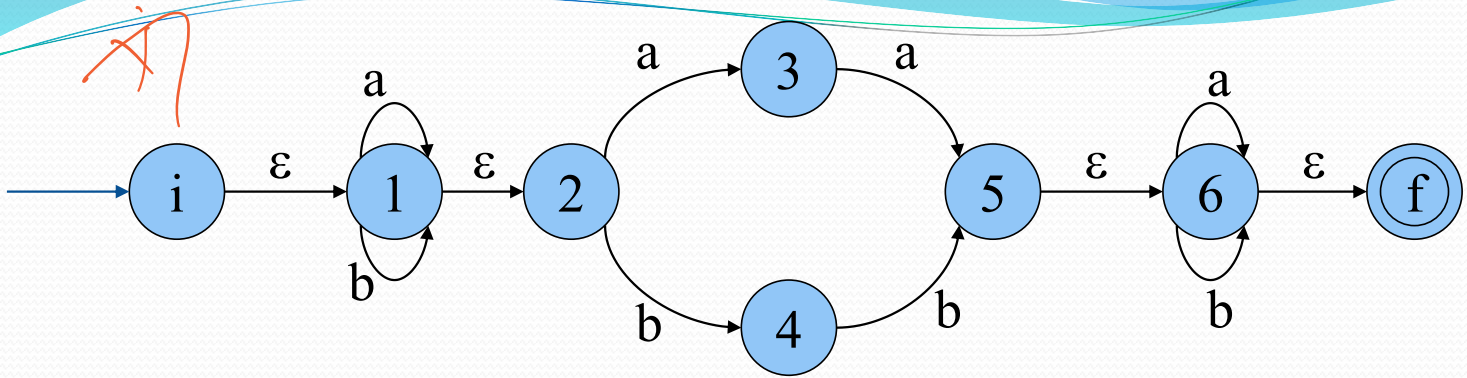


Implementation

```
i=0;
state=0;
while(input[i]){
 state=A[state,input[i++]];
}
```

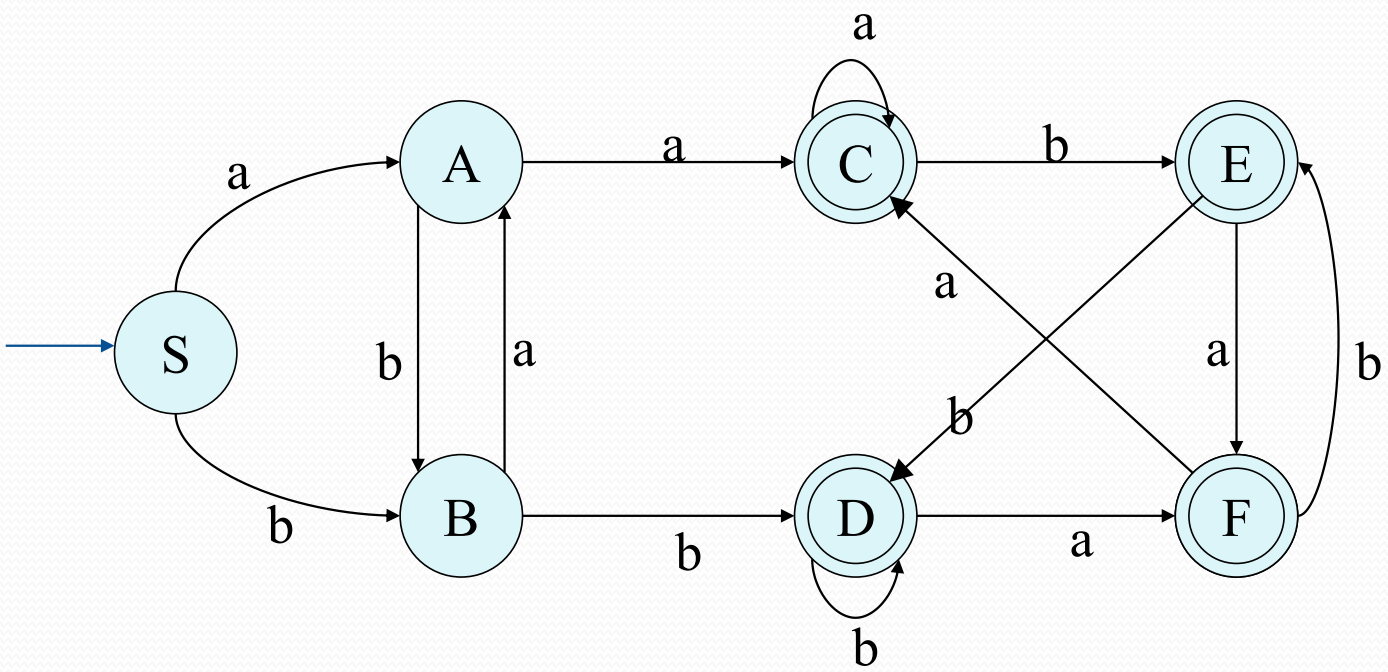
Array A:

	0	1
S	T	U
T	T	U
U	T	U

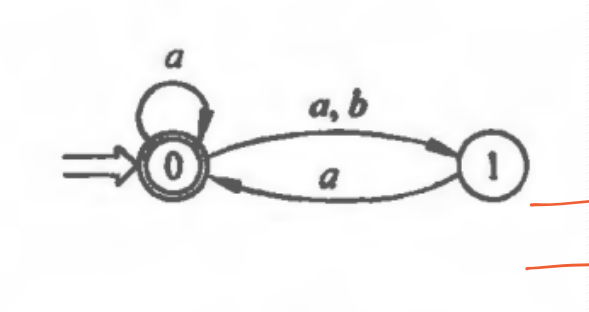


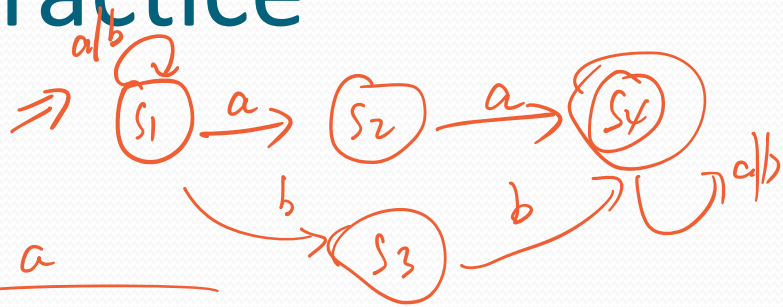
	Ia	Ib
{i,1,2} S	{1,2,3} A	{1,2,4} B
{1,2,3} A	{1,2,3,5,6,f} C	{1,2,4} B
{1,2,4} B	{1,2,3} A	{1,2,4,5,6,f} D
{1,2,3,5,6,f} C	{1,2,3,5,6,f} C	{1,2,4,6,f} E
{1,2,4,5,6,f} D	{1,2,3,6,f} F	{1,2,4,5,6,f} D
{1,2,4,6,f} E	{1,2,3,6,f} F	{1,2,4,5,6,f} D
{1,2,3,6,f} F	{1,2,3,5,6,f} C	{1,2,4,6,f} E

NFA -> DFA Example

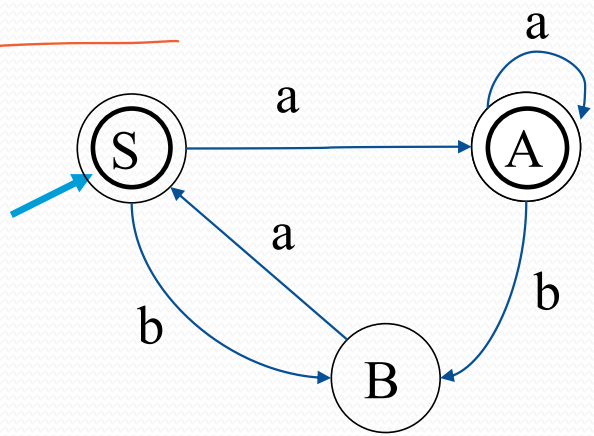


NFA -> DFA Practice

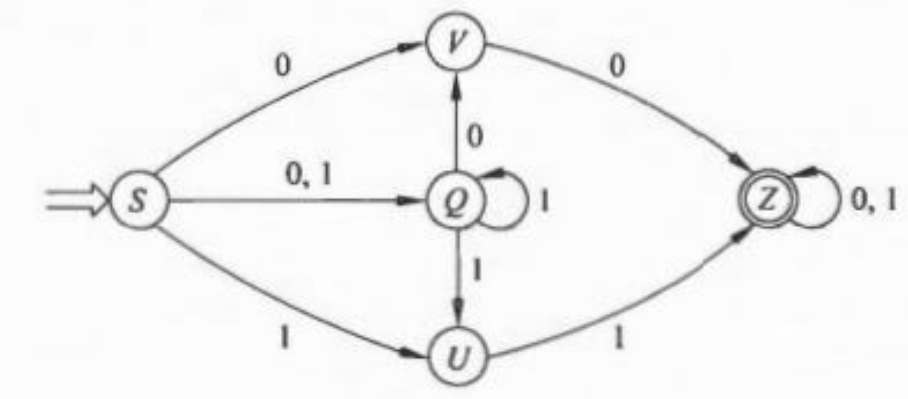




	a	b
S{0}	A{0,1}	B{1}
A{0,1}	A{0,1}	B{1}
B{1}	S{0}	--

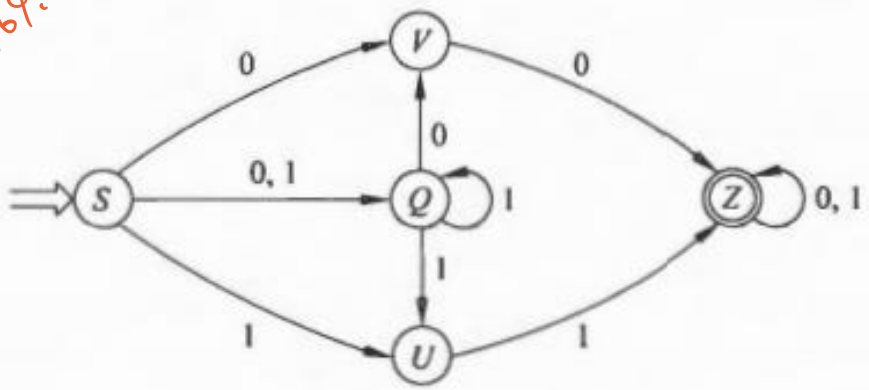


NFA -> DFA Practice



NFA -> DFA Practice

P64.3



	0	1
$S_0\{S\}$	$S_1\{V,Q\}$	$S_2\{Q,U\}$
$S_1\{V,Q\}$	$S_3\{V,Z\}$	$S_2\{Q,U\}$
$S_2\{Q,U\}$	$S_4\{V\}$	$S_5\{Q,U,Z\}$
$S_3\{V,Z\}$	$S_6\{Z\}$	$S_6\{Z\}$
$S_4\{V\}$	$S_6\{Z\}$	--
$S_5\{Q,U,Z\}$	$S_3\{V,Z\}$	$S_5\{Q,U,Z\}$
$S_6\{Z\}$	$S_6\{Z\}$	$S_6\{Z\}$