

第二章：词法分析

自动机与正则表达式
词法分析器的设计

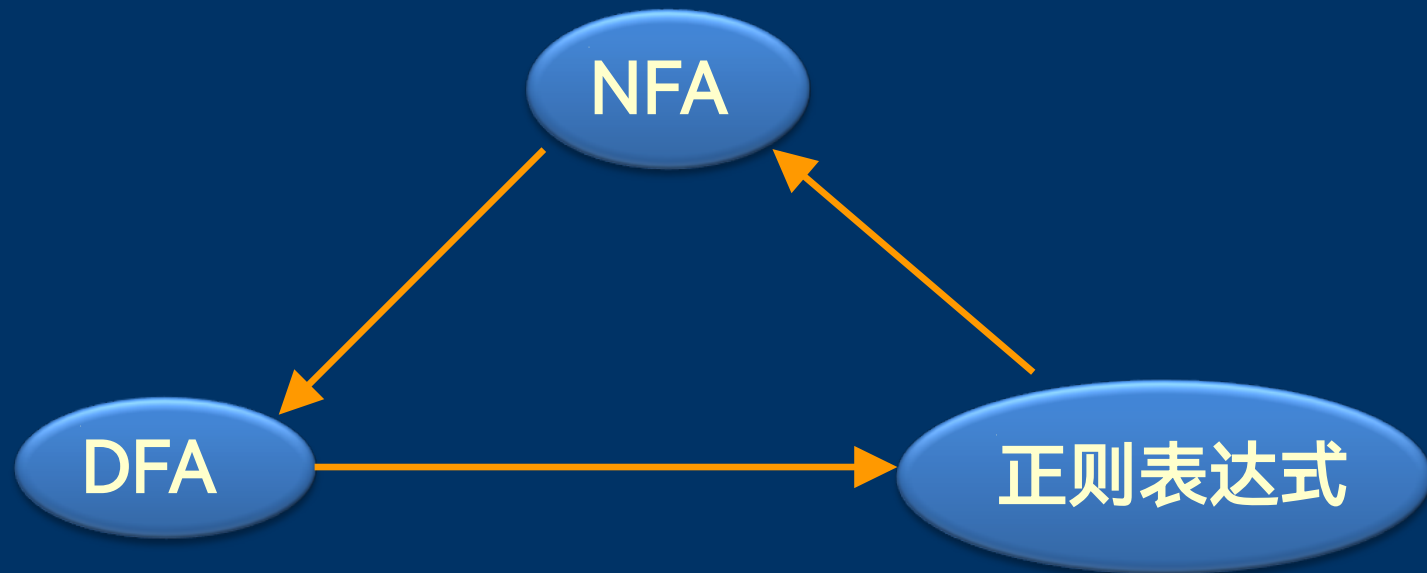


内容介绍

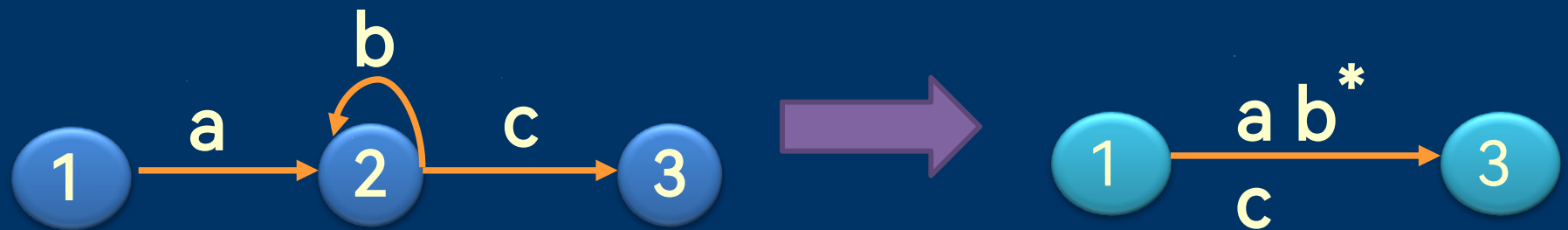
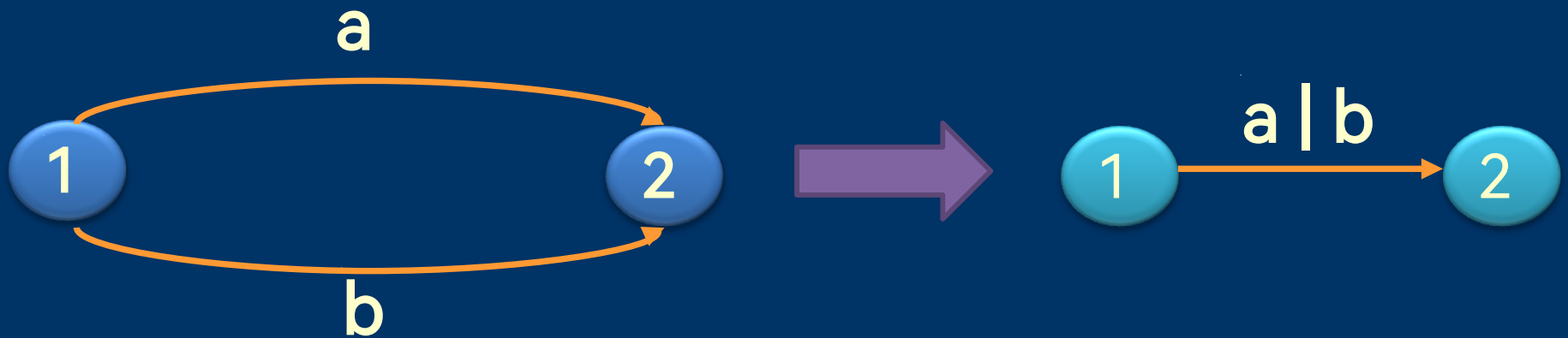
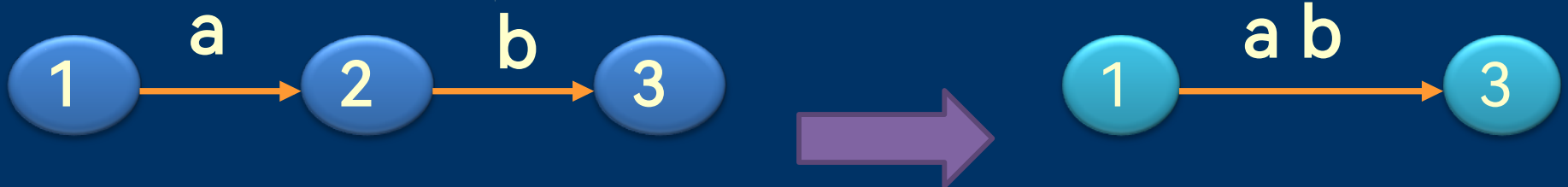
- w 自动机与正则表达式的关系
- w 词法分析器的具体设计

1. 自动机与正则表达式的关系

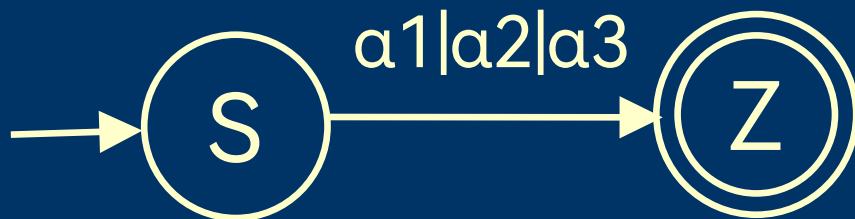
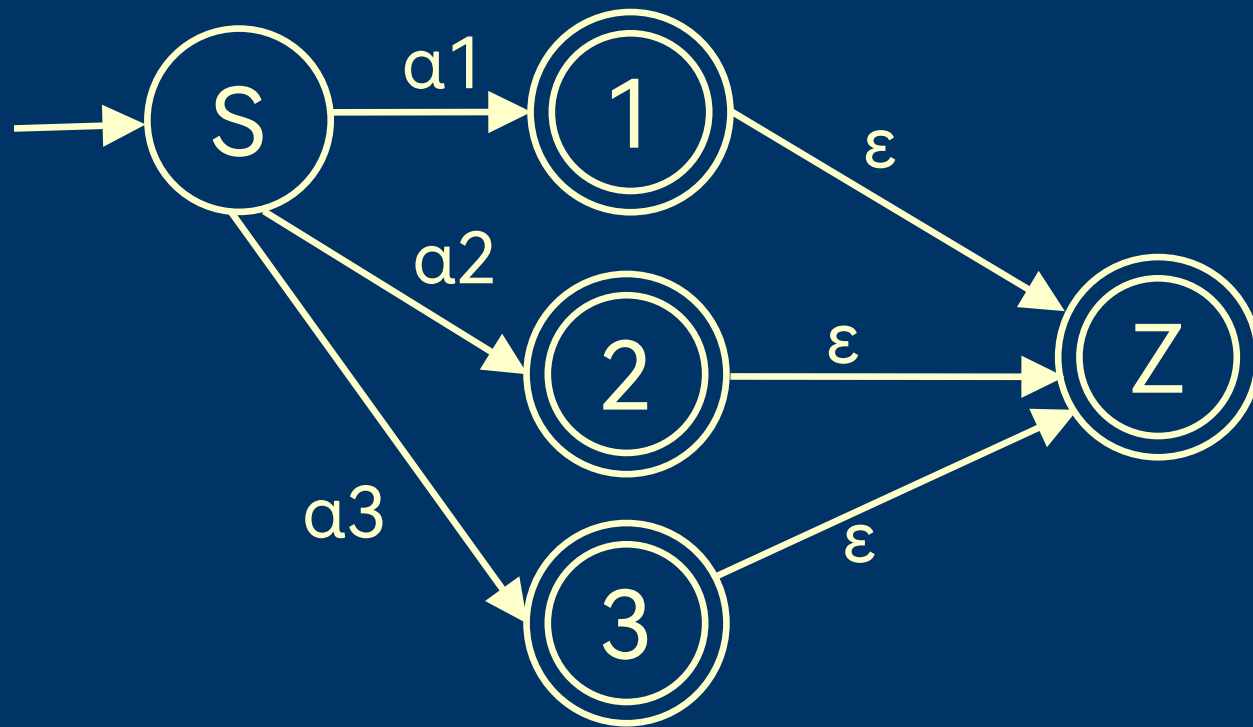
- 定理：对任一确定有限自动机A，存在一正则表达式e，使得 $L(A)=L(e)$ ，反之亦然。



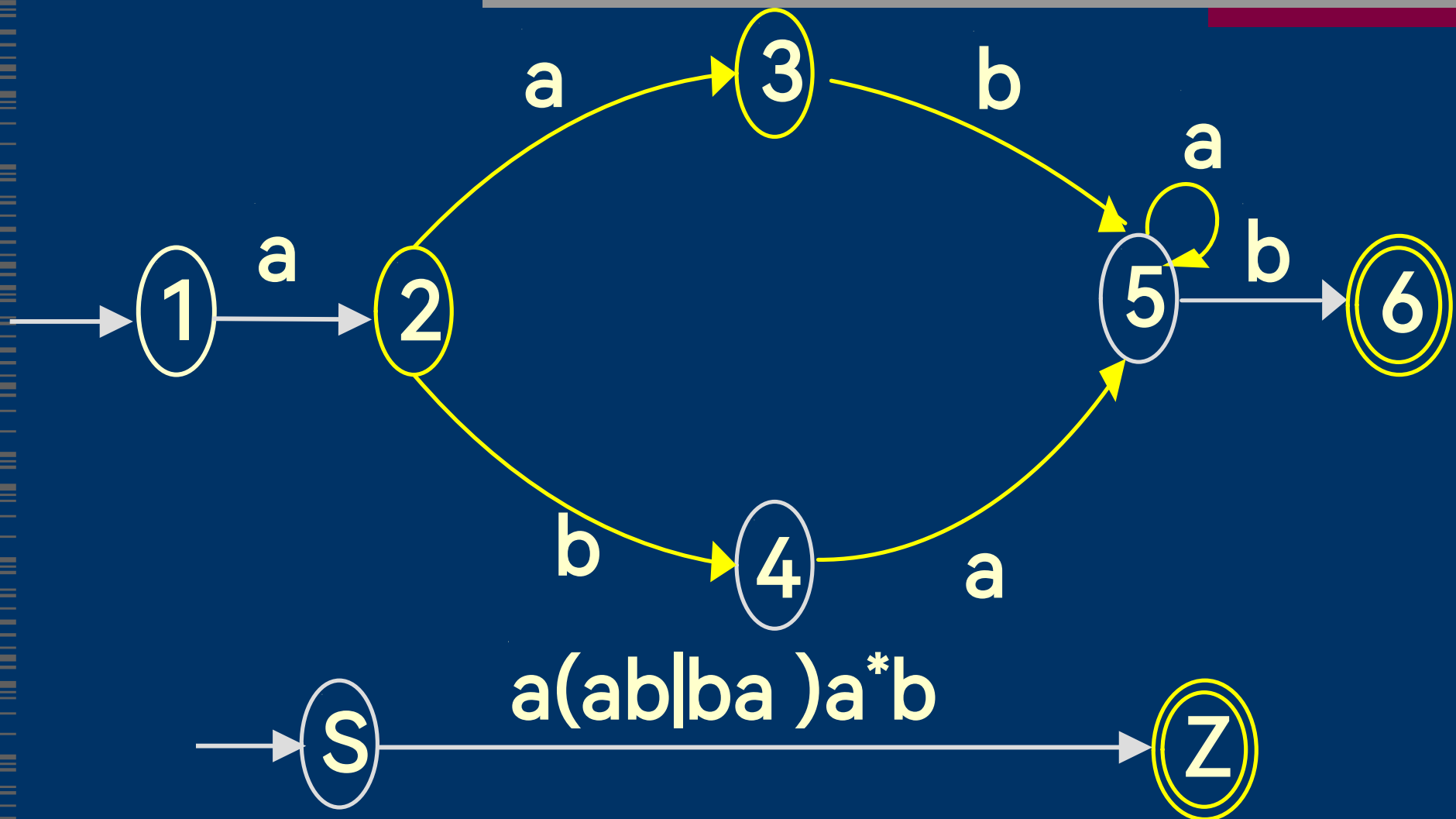
1.1 DFA到正则表达式



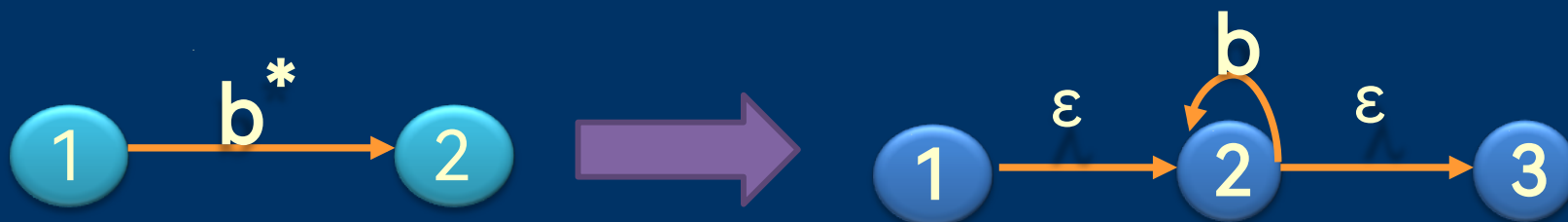
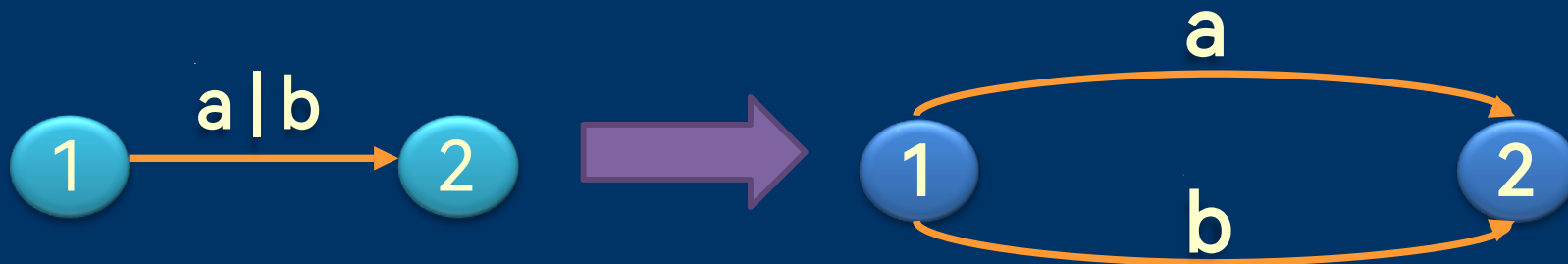
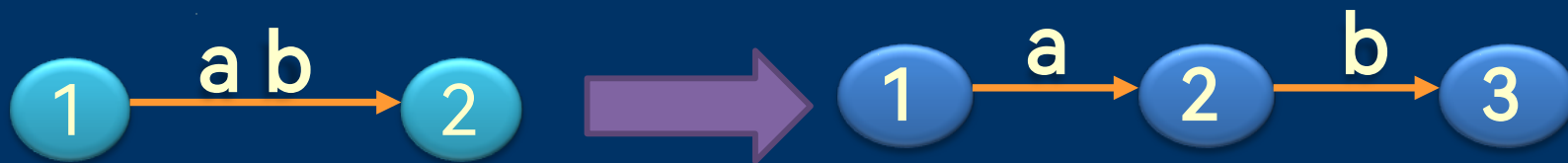
1.1 DFA到正则表达式



1.1 DFA到正则表达式

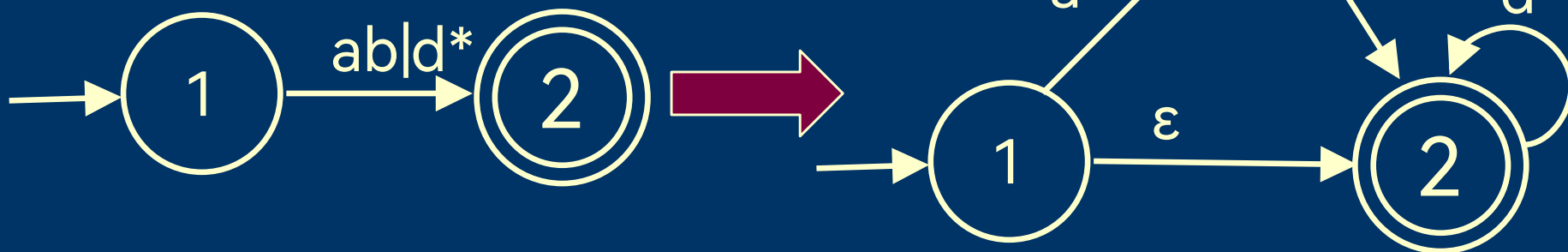
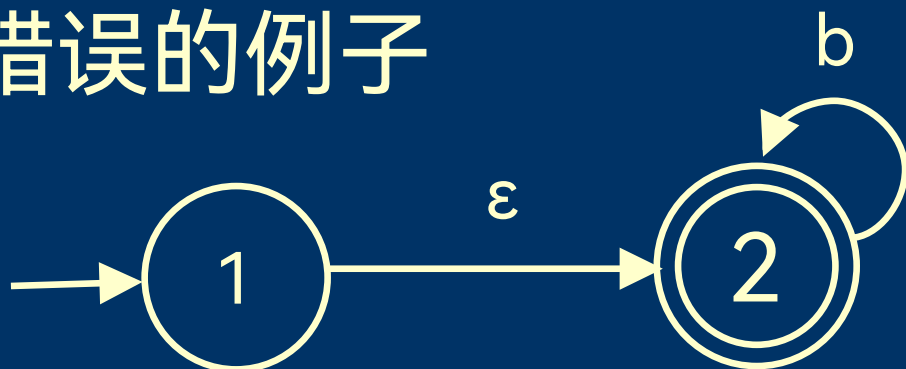


1.2 正则表达式到NFA



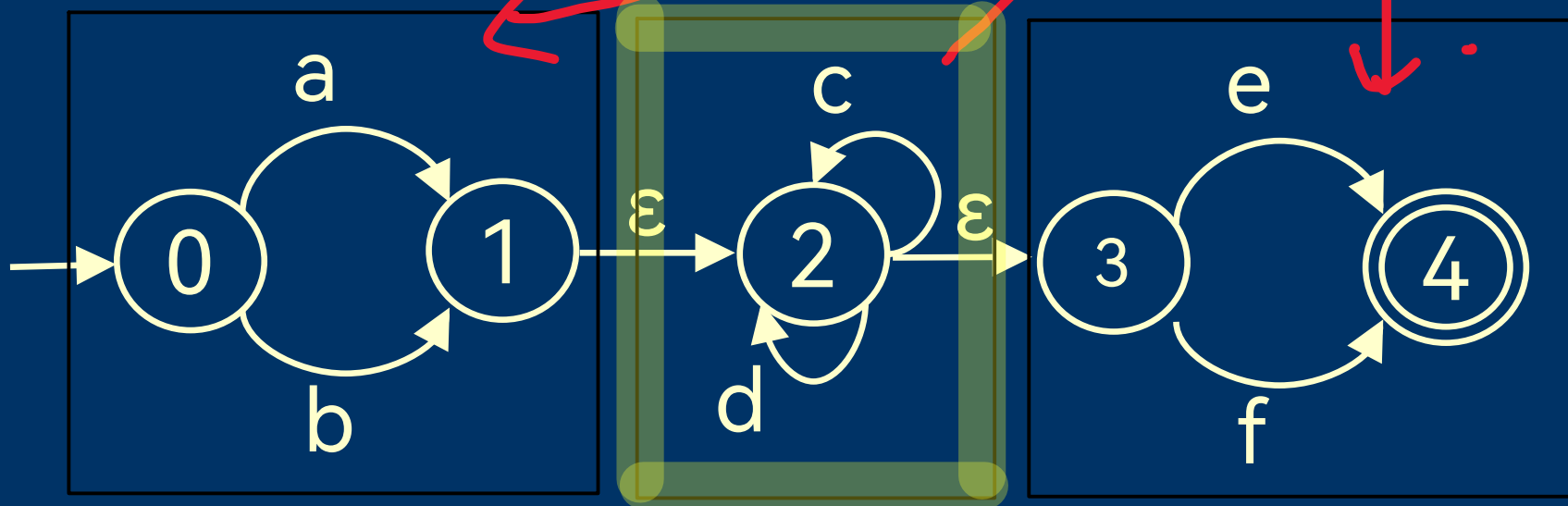
1.2 正则表达式到NFA

错误的例子



1.3 相关的例子

□ 给出一个正则表达式 $(a|b)(c|d)^*(e|f)$



1.3 相关的例子

□ 转化成DFA

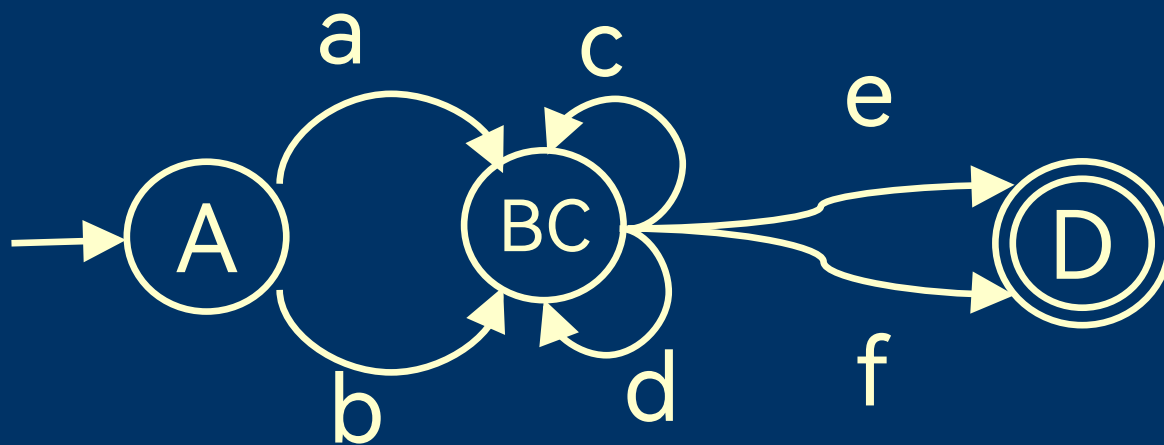
		a	b	c	d	e	f
A	$\{0\}^+$	$\{123\}$	$\{123\}$				
B	$\{123\}$			$\{23\}$	$\{23\}$	$\{4\}$	$\{4\}$
C	$\{23\}$			$\{23\}$	$\{23\}$	$\{4\}$	$\{4\}$
D	$\{4\}^*$						

1.3 相关的例子

□ 最小化

$\{A, B, C\} \{D\}$

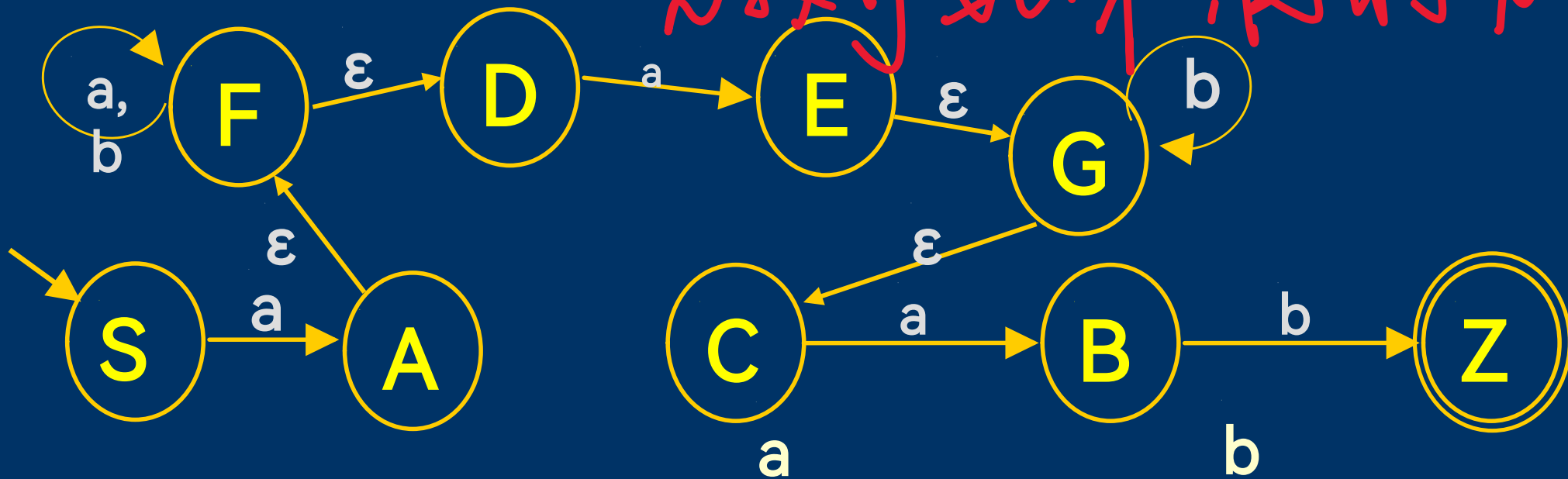
$\{A\} \{B, C\} \{D\}$



类型题

- 构造与正则表达式 $a((a|b)^*ab^*a)b$ 等价的最简DFA，要求给出中间转换步骤和结果

先按图写出来，再划分



{S}⁺

{A,F,D}

{A,F,D}

{F,E,D,G,C}

{F,D}

{F,E,D,G,C}

{F,E,B,D,G,C}

{F,G,D,C}

{F,D}

{F,E,D,G,C}

{F,D}

{F,E,B,D,G,C}

{F,E,B,D,G,C}

{F,G,Z,D,C}

{F,G,D,C}

{F,E,B,D,G,C}

{F,G,D,C}

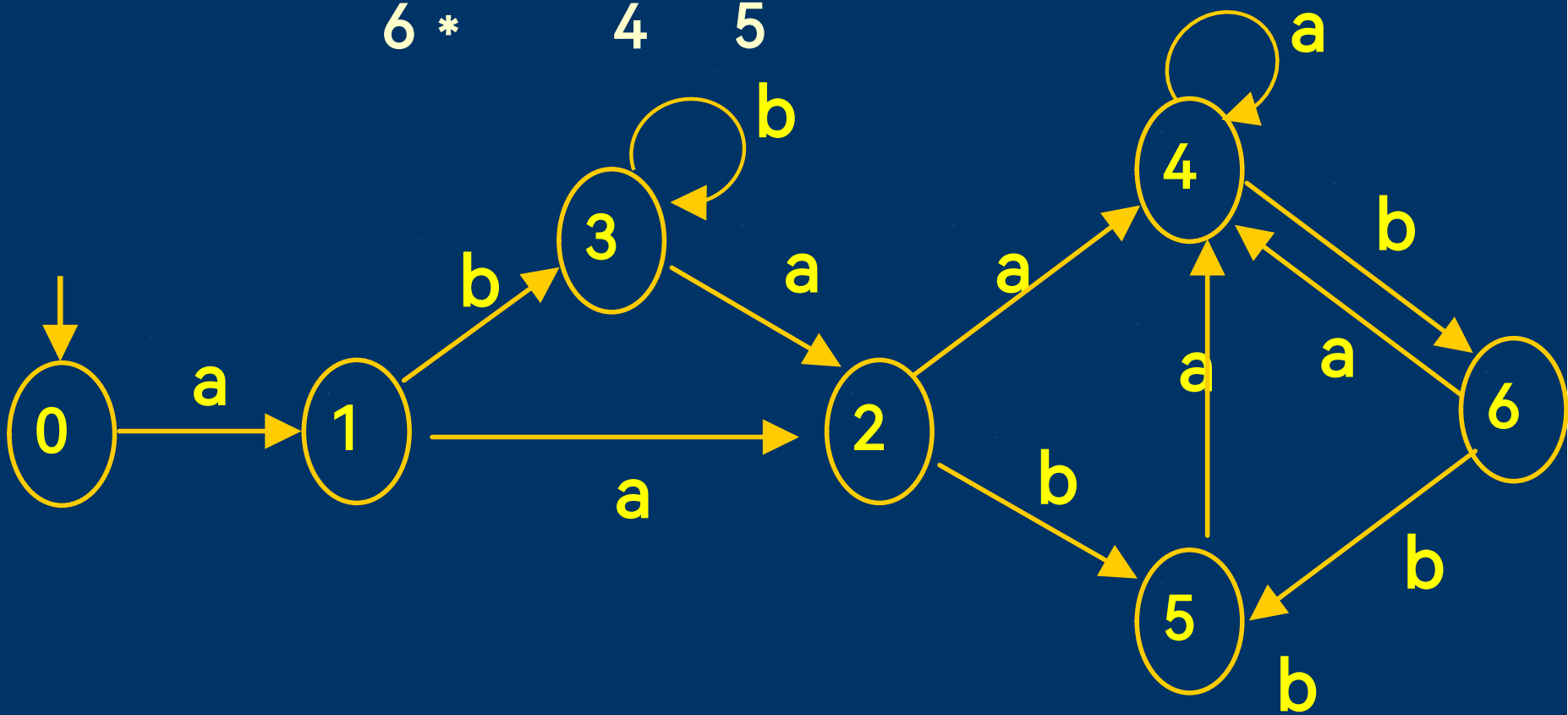
{F,G,Z,D,C} *

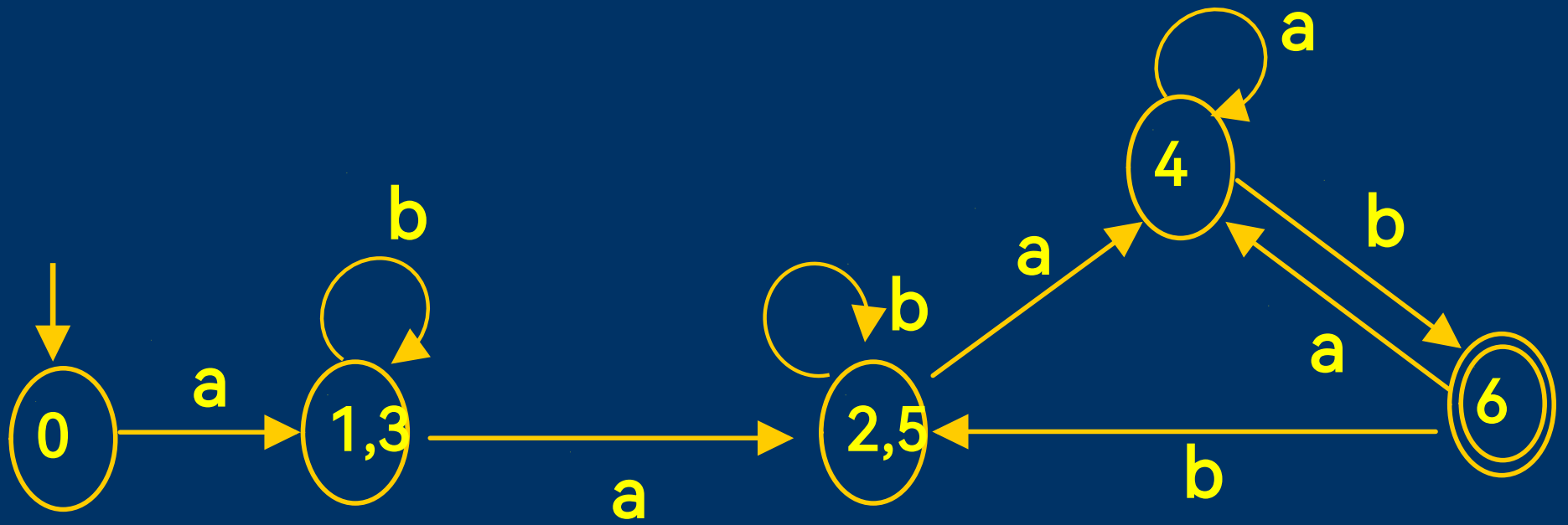
{F,E,B,D,G,C}

{F,G,D,C}

确定化结果

0^+	1	
1	2	3
2	4	5
3	2	3
4	4	6
5	4	5
6 *	4	5



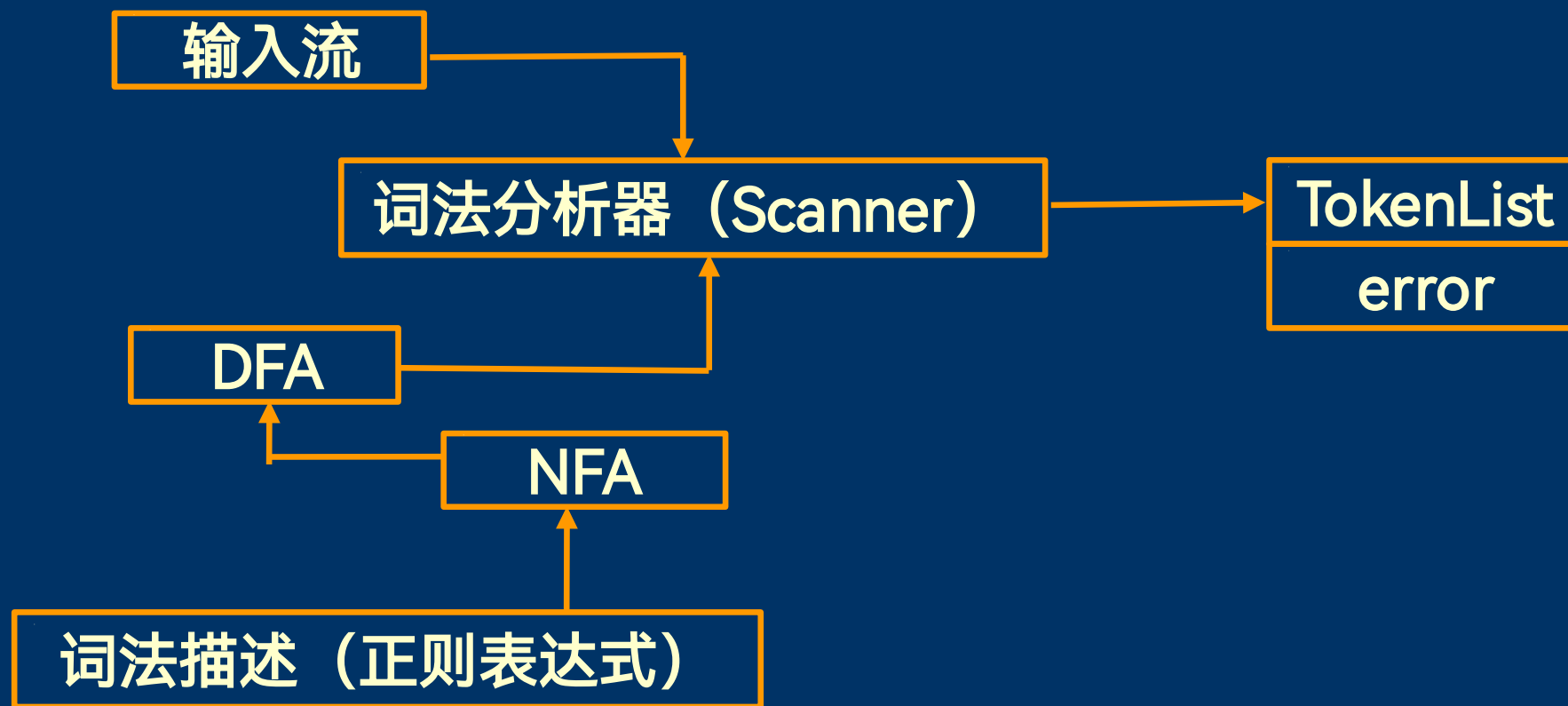


2. 词法分析器的设计

□ 设计步骤

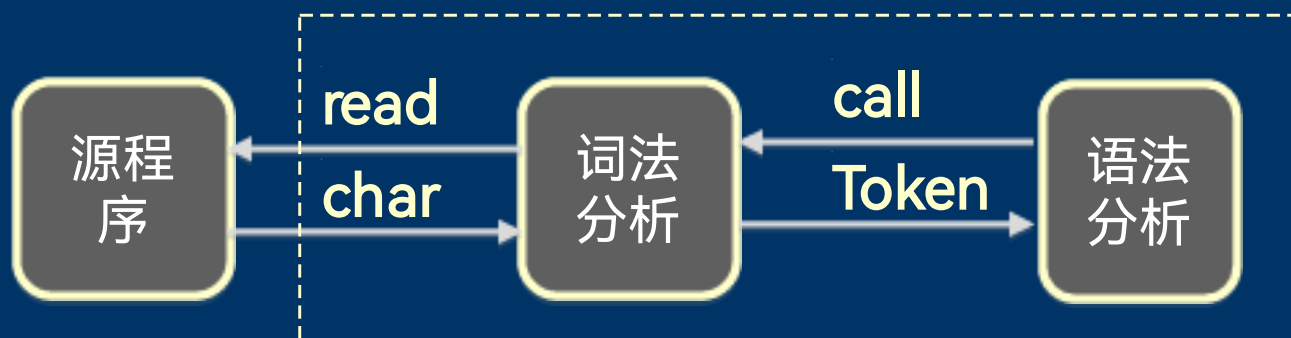
- ❖ 确定词法分析的接口
- ❖ 确定单词的结构
- ❖ 给出单词的描述
- ❖ 设计算法

2.1 词法分析器的工作过程

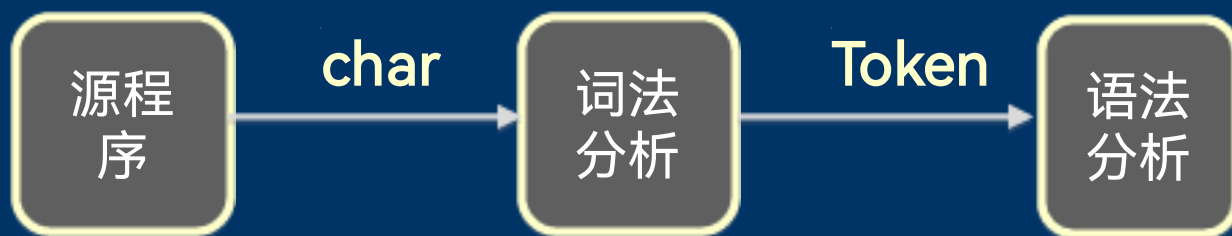


2.2 词法分析器的接口

- 词法分析器有两类，一类是仅作为语法分析的子程序：



- 另一类是作为编译器的独立一遍处理器：



2.3 单词的结构

- 源程序的处理过程是，首先从源程序文件一个字符一个字符进行读取，并逐个分离出单词，然后构造它们的机内表示Token。
- 关于Token的结构没有统一的规定，但至少包括两部分内容：
 - 单词的类型（语法信息）
 - 单词的内容（语义信息）

2.4 一种可行的Token样例

Token结构

类型	内容
----	----

标识符索引表	
1	
2	
3	
...	...
n	

常量索引表	
1	
2	
3	
...	...
n	

保留字、特殊符号表	
3	while
4	if
5	%
...	...
56	}

※如果单词为标识符，则类型填1

※如果单词为常量，则类型填2

※如果类型为保留字或特殊符号，则类型填该保留字或特殊符号所对应的数字

2.4 一种可行的Token样例

□ 例子 程序段:

```
{ int x,y;
```

```
x=10;
```

```
y=11;
```

```
x=x+y;
```

```
....
```

```
}
```

2.5 程序的实现

□ 标识符的实现

拼接标识符，停留在某状态，确定是标识符状态，查找保留字表，查找标识符表，确定其在标识符表中位置，生成对应Token

w 整数的实现

w 特殊符号的实现

2.6 注意的问题

- 名字的的实现方式
- 复合单词的识别
- 数的转换
- 向前看若干个字符
- 控制字符的处理
- 注释的处理

词法分析总结

□ 这一章的内容需要掌握的是以下几点：

- ❖ 一个核心：词法分析器的设计
- ❖ 两个工具：正则表达式和自动机
- ❖ 三个转换算法：NFA到DFA，自动机的极小化，正则表达式和自动机的互相转换