

# 第 5 次作业

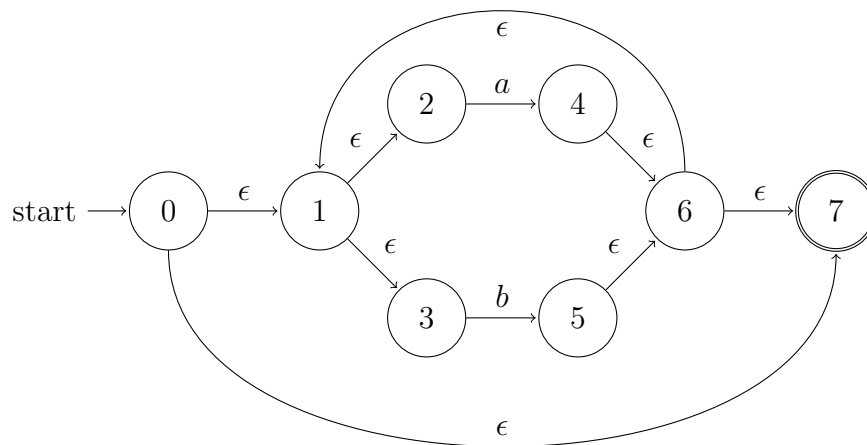
Log Creative

2021 年 6 月 28 日

1 将下列正则表达式转换成DFA:

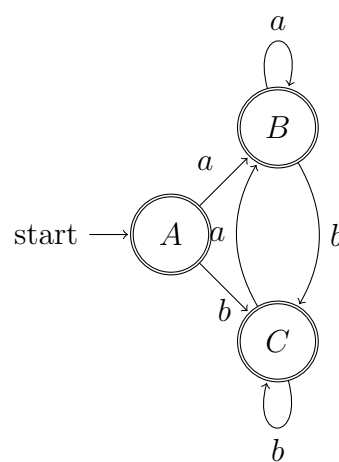
a)  $(a|b)^*$

解. 先转换为 NFA,



DFA 转换表如下:

NFA 状态	DFA 状态	a	b
$\{0,1,2,3,7\}$	A	B	C
$\{1,2,3,4,6,7\}$	B	B	C
$\{1,2,3,5,6,7\}$	C	B	C

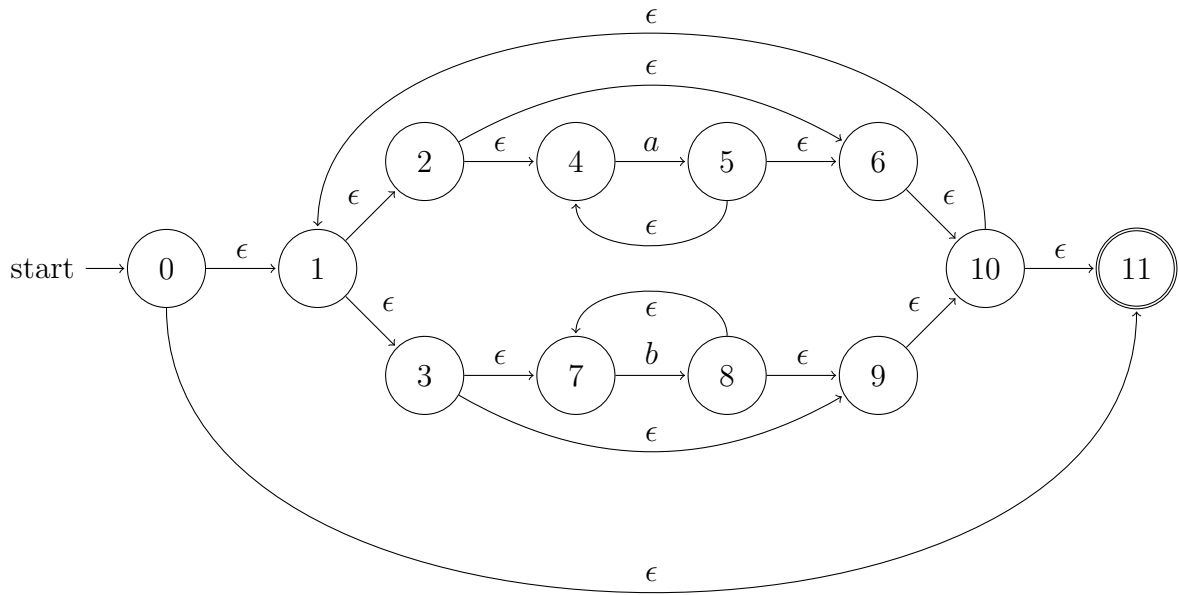


最终 DFA 如右图所示。

b)  $(a^*|b^*)^*$

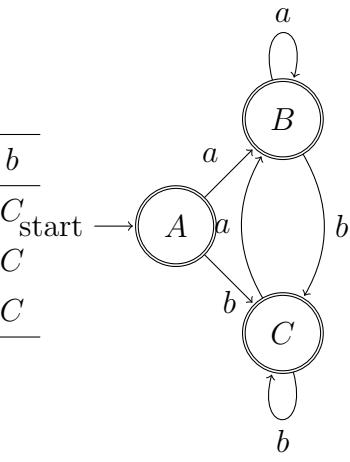
解.

先转换为 NFA，



DFA 转换表如下：

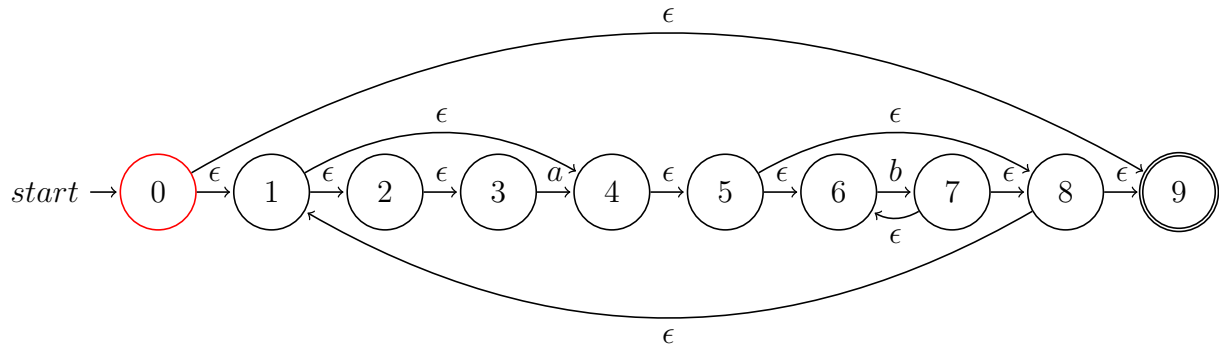
NFA 状态	DFA 状态	$a$	$b$
$\{0,1,2,3,4,6,7,9,10,11\}$	$A$	$B$	$C$
$\{1,2,3,4,5,6,7,9,10,11\}$	$B$	$B$	$C$
$\{1,2,3,4,6,7,8,9,10,11\}$	$C$	$B$	$C$



最终 DFA 如右图所示。

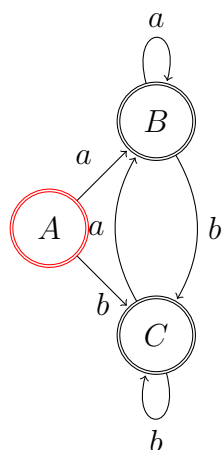
c)  $((\epsilon|a)b^*)^*$

解. 先转换为 NFA，



DFA 转换表如下:

NFA 状态	DFA 状态	$a$	$b$
$\{0,1,2,3,4,5,6,8,9\}$	$A$	$B$	$C$
$\{1,2,3,4,5,6,8,9\}$	$B$	$B$	$C$
$\{1,2,3,4,5,6,7,8,9\}$	$C$	$B$	$C$



最终 DFA 如右图所示。

- 2** 对于深度学习编译器来说，表达一个深度神经网络的基本语言是什么？如果让你设计一种语言，并构造一种词法分析器，你会怎么做？你也可以替换成任何一种感兴趣的语言，如数据库查找SQL等。

答: 目前, 大部分深度学习编译器表达深度神经网络的基本语言是 Python, 由于 Python 较慢的速度也造成了一定的性能瓶颈。<sup>[1]</sup>

建立对于  $\{\text{L}\text{T}_{\text{E}}\text{X}_{\epsilon} \text{ 命令}\} - \{\text{T}_{\text{E}}\text{X} \text{ 命令}\}$  子集的词法分析器 (lexer)。这些命令用于 AutoBeamer 识别文档分块。

## 词法单元

AutoBeamer 主要识别两种基本的词法单元，正则描述如表 1 所示：

- 命令（比如：\title{Slide Title}，命令参数是可选的）
- 英文词（比如：beamer）
- 其他语言的单字（比如：幻，灯，片）

表 1: AutoBeamer 所识别的基本词法单元

词法单元	非正式描述	正则描述
command	命令	<code>\\w+(\[.*\] \{.*\})*</code>
word	词	<code>\\w+</code>
character	字	<code>[^(\ \\w\\s)]</code>

此外，对于命令语法单元，AutoBeamer 需要进行扩展识别，其相关正则描述如表 2 所示：

- 环境开始（比如：`\begin{document}`）
- 环境结束（比如：`\end{document}`）
- 文档类（比如：`\documentclass{article}`）

表 2: AutoBeamer 所识别的命令扩展词法单元

词法单元	非正式描述	正则描述
<b>begin</b>	环境开始	<code>\\begin{\w+}</code>
<b>end</b>	环境结束	<code>\\end{\w+}</code>
<b>class</b>	文档类	<code>\\documentclass(\[ \w*\])?\{ \w+\}</code>

## 构建 DFA

确定的有穷自动机 (Deterministic Finite Automata, DFA) 是构建词法分析器的重要依据。

基本的词法单元需要先识别命令，再识别词/字，其 DFA 如图 1 所示。注意，输入的文件应该是已经可以通过 L<sup>A</sup>T<sub>E</sub>X 编译的普通文档（不会出现 Error 异常，能输出 PDF 文档），所以 \ 后面应该就是字母，否则会报错。

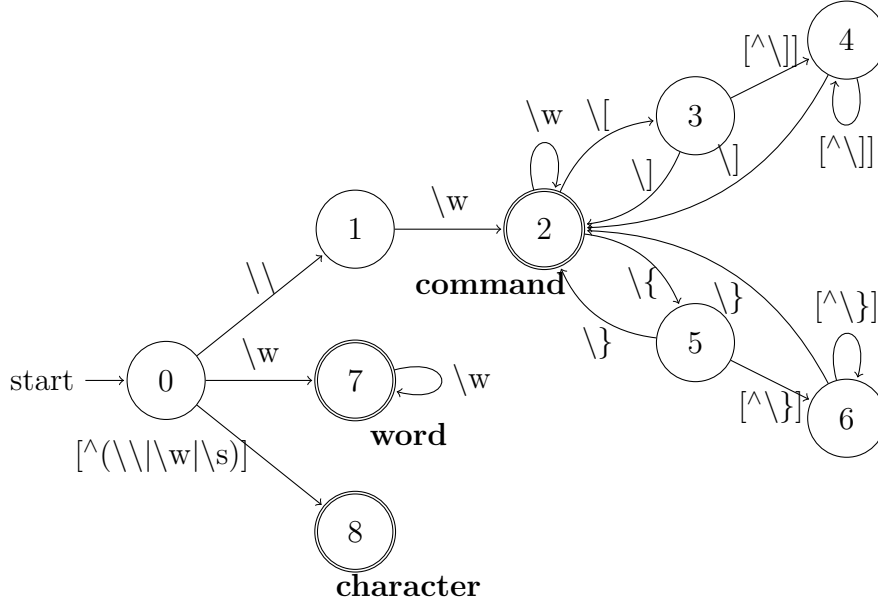


图 1: 基本词法单元的 DFA

对于命令扩展词法单元而言，需要有其已经是命令的先决条件，其 DFA 如图 2 所示。仍然是在文档是编译正确的前提下进行的，如果没有匹配到扩展命令，将仍然作为 **command** 词法单元。

DFA 构建完毕后，就可以编写对应的词法分析器了，后续的代码进展请进入：

<https://github.com/LogCreative/AutoBeamer/tree/main/src/lexer>。

## 参考文献

- [1] LI M, LIU Y, LIU X, et al. The deep learning compiler: A comprehensive survey[J/OL]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(3): 708-727. DOI: [10.1109/TPDS.2021.3054444](https://doi.org/10.1109/TPDS.2021.3054444).

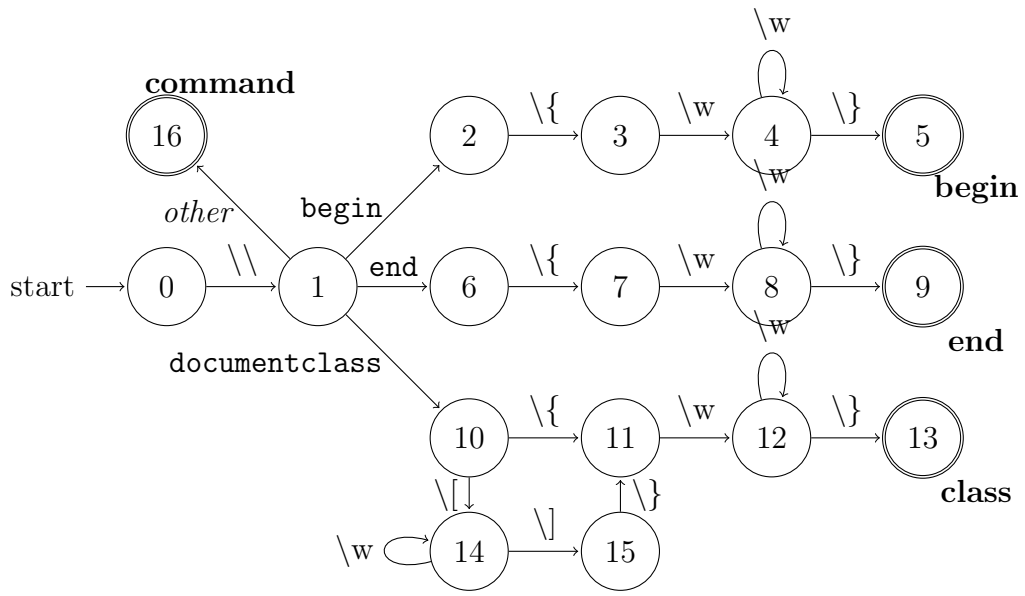


图 2: 命令扩展词法单元的 DFA

109/tpds.2020.3030548.