

<https://github.com/lazyparser/becoming-a-compiler-engineer>

方舟·编译技术入门与实战

第五课：习题课（1）

吴伟 (@lazyparser)

2019-12-28

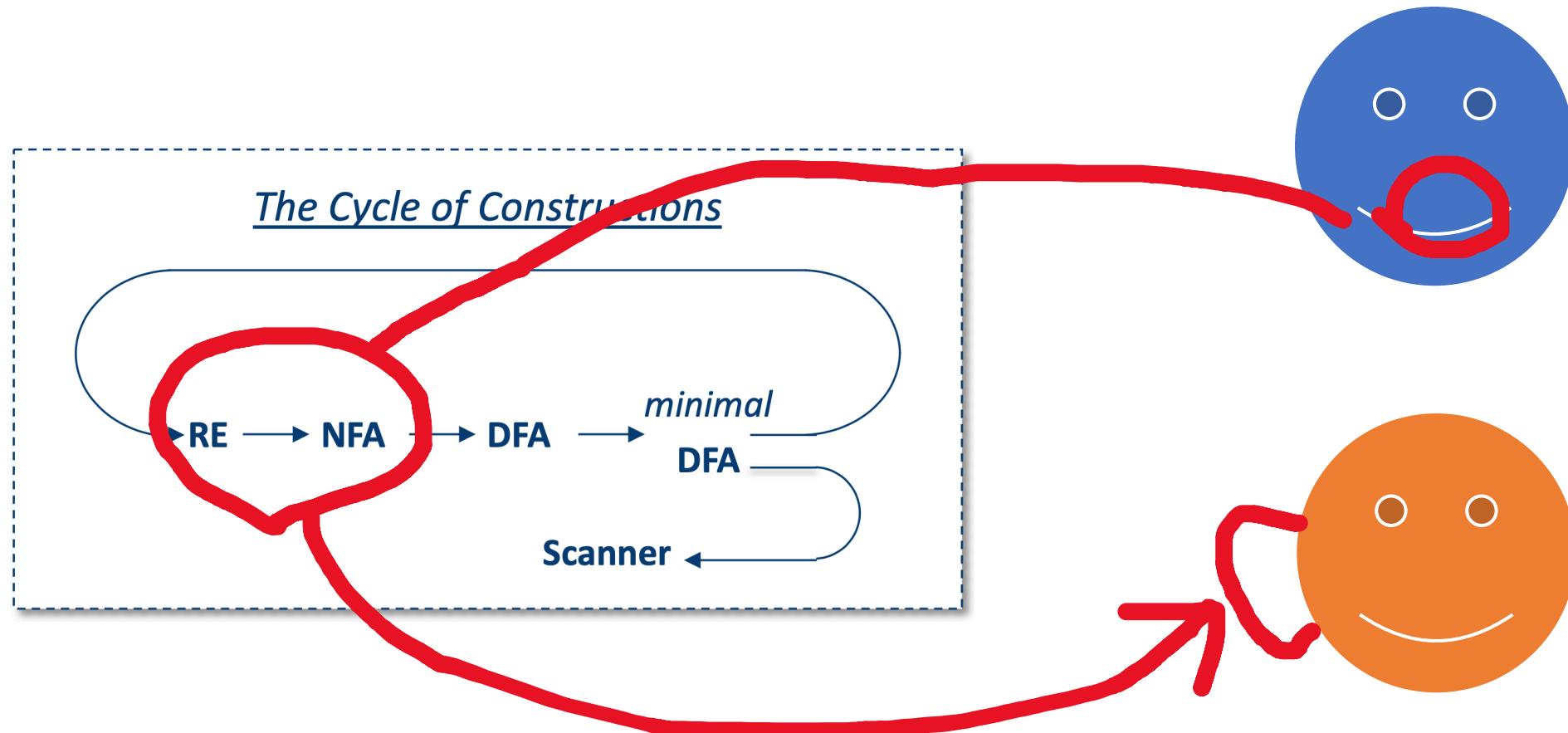
本课程所有资料都是开源免费的（更新中）

- 课程配套代码及幻灯片地址（也是提问的地方）
 - <https://github.com/lazyparser/becoming-a-compiler-engineer>
 - <https://github.com/lazyparser/becoming-a-compiler-engineer-codes>
- 课程视频回看（包含所有直播及录播视频）
 - <https://space.bilibili.com/296494084>
- 课程直播地址（可以弹幕或评论区互动）
 - <https://live.bilibili.com/10339607>

本次课程将分四次视频介绍

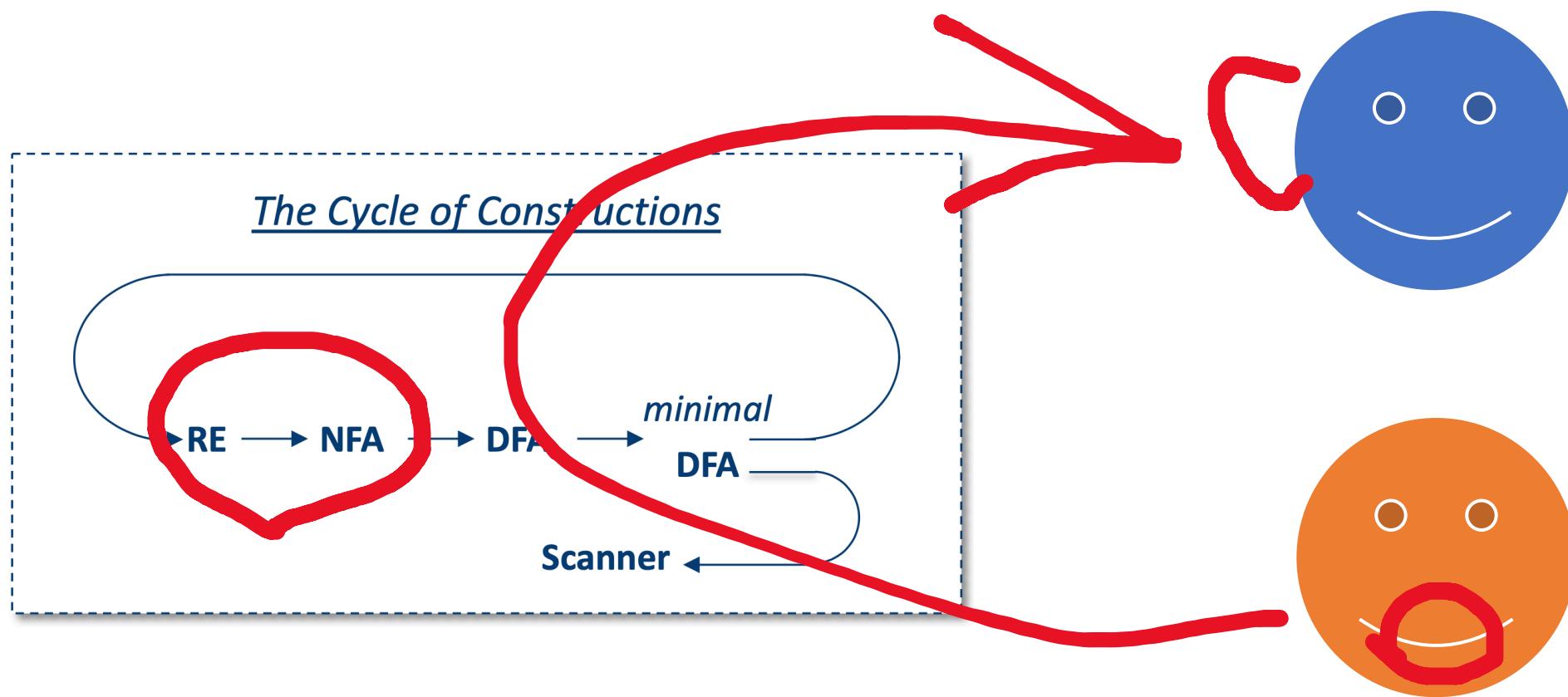
- 已经学习的思维模式的回顾
- 讲解《现代编译原理》第二章词法分析课后习题（今天）
- 讲解《编译器设计》第二章词法分析课后习题（明天）
- 讲解《现代编译原理》第三章语法分析课后习题（后天）
- 讲解《编译器设计》第三章语法分析课后习题（大后天）

习题课是知识的理解表达：一定要讨论



<https://www.clear.rice.edu/comp412/Lectures/>

习题课是知识的理解表达：一定要讨论



<https://www.clear.rice.edu/comp412/Lectures/>

复习：思考算法问题的几个思考模式

- 问题是如何定义的？
- 解空间规整么？
- 解的存在性证明了么？
- 构造性的方法来找到解？
- 穷举法？贪婪法？
- 是要找验证算法么？
- 是否可以构造等价映射？
- 能Google到答案么？

Google First. && Must Google Harder.

不断的变换搜索词，
同义词，近义词
中文，英文
特定网站名字

modern compiler implementation book exercise reference answer solution

All Images News Shopping Videos More Settings Tools

About 20,800,000 results (0.57 seconds)

[dvirtz/GreenTiger: Solutions to exercises of Modern ... - GitHub](#)
<https://github.com/dvirtz/GreenTiger> ▾
GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together. ... Each chapter contains a program which is a step towards building a **compiler** for the Tiger programming language as well as some **solutions** to other ...
Missing: **reference** | Must include: **reference**

[Modern compiler implementation c | Programming languages ...](#)
[www.cambridge.org](http://www.cambridge.org/core/title) ▾ core_title
Detailed descriptions of the interfaces between modules of a **compiler** are illustrated with actual C header files. The first part of the **book**, Fundamentals of ...

为什么强调 Google First 和 Must Try Harder.

- 你已经不是中小学生了 : 解决技术问题和业务需要
 - 怀念考卷题目做对了就可以得到表扬的校园时代 ☺
- 代码这种事, 即使是复制粘贴, 也太难了
 - 分析清楚, 要解决什么问题 ; 定位出来, 缺失什么, 要补什么
 - 发挥专业经验寻找可能的方案, 搜索, 尝试, 搜索, 尝试
 - 最终产品里, 日均20行代码 ; 可能背后尝试且丢弃的代码有2000行 +

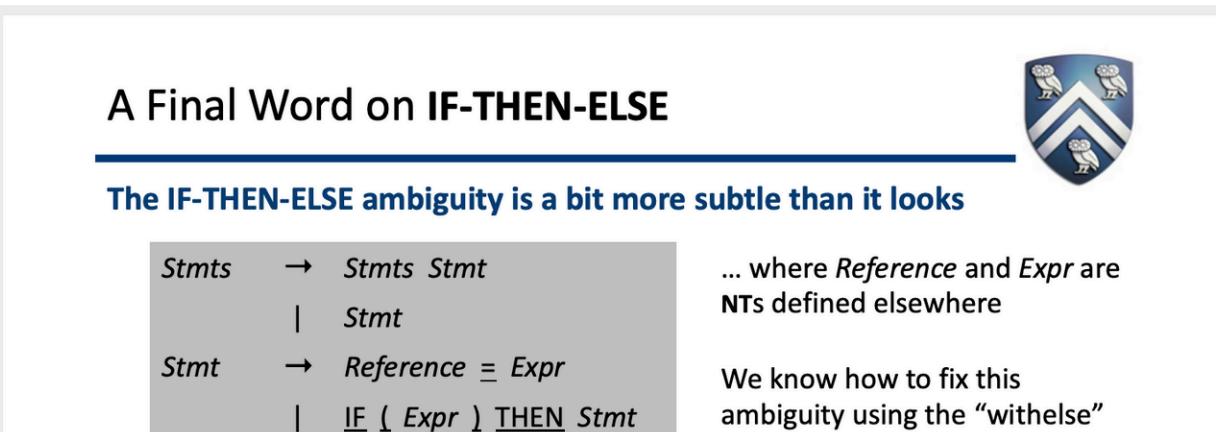
SSR : 自学+讨论 : 在GitHub上提问题

请教一下，下面这页PPT说增加WHILE语句就会导致IF语句有歧义，不太明白 #11

 Closed 01nomagic opened this issue 17 hours ago · 3 comments

01nomagic commented 17 hours ago • edited  

PPT地址: <https://www.clear.rice.edu/comp412/Lectures/L08Parsing-1.pdf>



Assignees
No one—assign yourself

Labels
None yet

Projects
None yet

SSR : 自学+讨论 : 也可以汇总在一起

<input type="checkbox"/>	① 7 Open ✓ 5 Closed	Author ▾	Label ▾
<input type="checkbox"/>	① 作业上传讨论: 语法分析、《现代编译原理》第三章 #15 opened 3 hours ago by lazyparser		
<input type="checkbox"/>	① 作业上传讨论: 语法分析、《编译器设计》第三章 #14 opened 3 hours ago by lazyparser		
<input type="checkbox"/>	① 作业上传讨论: 词法分析、《现代编译原理》第二章 #13 opened 3 hours ago by lazyparser		
<input type="checkbox"/>	① 作业上传讨论: 《编译器设计》第二章词法分析 #12 opened 3 hours ago by lazyparser		

编译技术课程的讨论注意事项

- 就像是RE到NFA有很多种转换方法，**答案不唯一**
- 很多时候**需要**讨论来确定自己的答案和想法是否正确
- 不要盲信：包括我和所有助教/学生的答案可能都是错误的
- 遇到以上情况，**有任何不确定的感觉，直接issues里贴出来问**
 - 说不定还能帮助到同样有问题却不好意思发帖的人 ☺

《虎书》习题 2.1

2.1 写出下面每一种单词的正则表达式。

- a. 字母表 $\{a, b, c\}$ 上满足后面条件的字符串：首次出现的 a 位于首次出现的 b 之前。
- b. 字母表 $\{a, b, c\}$ 上由偶数个 a 组成的字符串。
- c. 是 4 的倍数的二进制数。
- d. 大于 101001 的二进制数。
- e. 字母表 $\{a, b, c\}$ 上不包含连续子串 baa 的字符串。
- f. C 语言中非负整常数组成的语言，其中以 0 开头的数是八进制常数，其他数是十进制常数。35
- g. 使得方程 $a^n + b^n = c^n$ 存在着整数解的二进制整数 n 。

提交的一些答案（课堂讨论用，不一定是正确答案）

- (a) $c^*(a[bc]^*)^*$
- (b) $[bc]^*(a[bc]^*a[bc]^*)^*$
- (c) $[01]^*00$ 2.1 (d) $(10101[01]+)|(1011[01]\{2\})|(11[01]\{4\})|([01]^*1[01]^*[01]\{6\})$
- (d) $(10101[01]+)|(1011[01]\{2\})|(11[01]\{4\})|([01]^*1[01]\{6\})$
- (e) $(a|c)^*(b|bc(a|c)^*|ba|bac(a|c)^*)^*$
- (f) $(0[0-7]+)|(0|[1-9][0-9]^*)$
- (g) $1|10$

感谢孙月影同学提供本次作业；感谢李威威同学进行批注

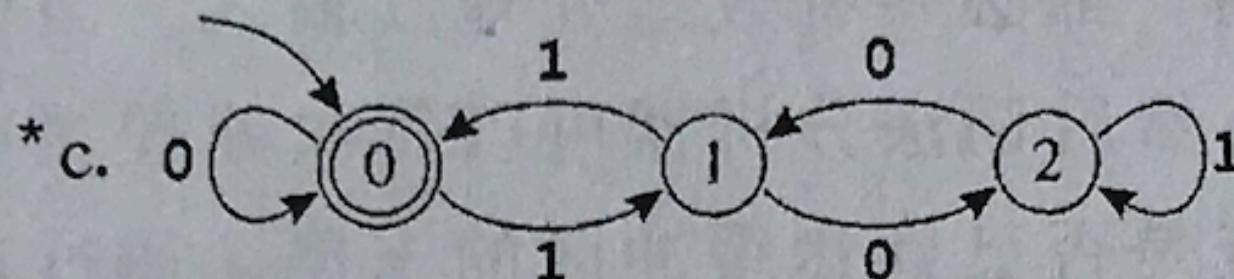
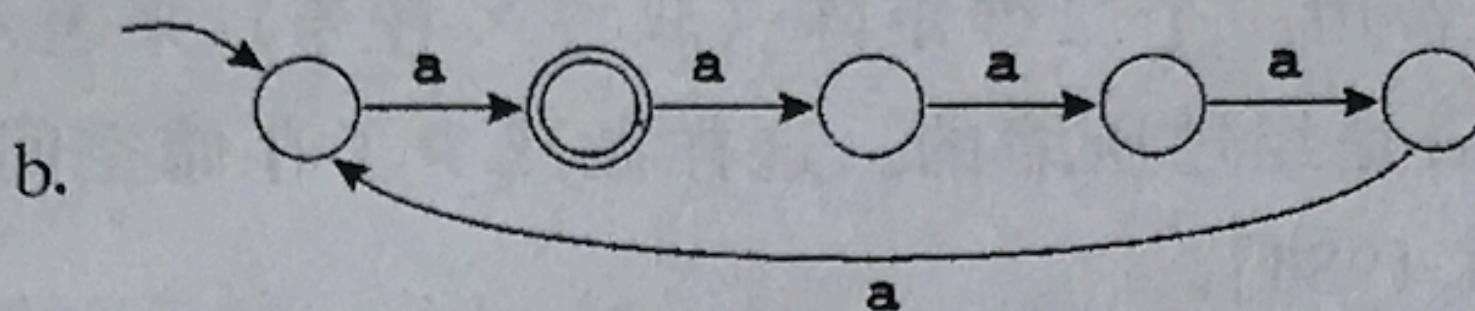
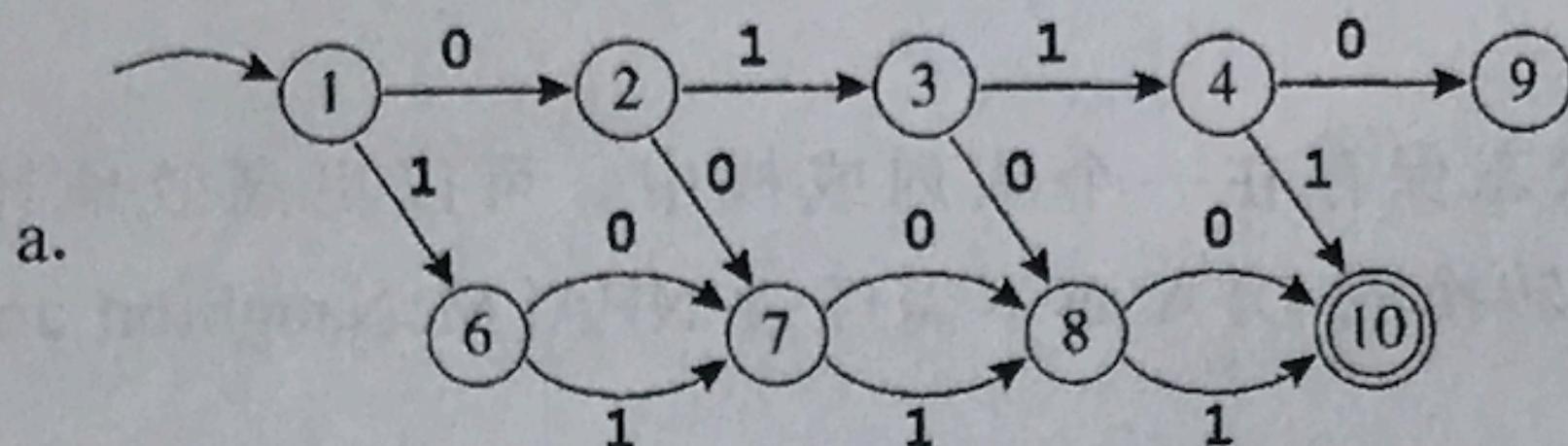
- 2.2 对于下列描述，试解释为什么不存在对应的正则表达式。
- a. 由 a 和 b 组成的字符串，其中 a 的个数要多于 b 。
 - b. 由 a 和 b 组成的回文字符串（顺读与倒读相同）。
 - c. 语法上正确的 C 程序。

提交的一些答案（课堂讨论用，不一定是正确答案）

- (a) 统计a和b出现的次数，会出现无限计数的情况，使状态不够用
- (b) 无法确定何时进行断句
- (c) 可能出现语句或者循环嵌套，无法识别

感谢孙月影同学提供本次作业；感谢李威威同学进行批注

2.3 用自然语言描述下述有限状态自动机识别的语言。



提交的一些答案（课堂讨论用，不一定是正确答案）

- (a) 除0110外的四位二进制数
- (b) 起始值为a, $d=5a$ 的等差数列字符串 (a aaaaaa aaaaaaaaaaa ...)
- (c) 回文二进制串 (如000 1001 101101

2.3(b) 感觉说成包含 $5n+1$ ($n = 0, 1, 2 \dots$) 个a的字符串好一些

2.3(c) 不完全是回文串 0001100它也接收，准确说是1开头的回文串+若干个0

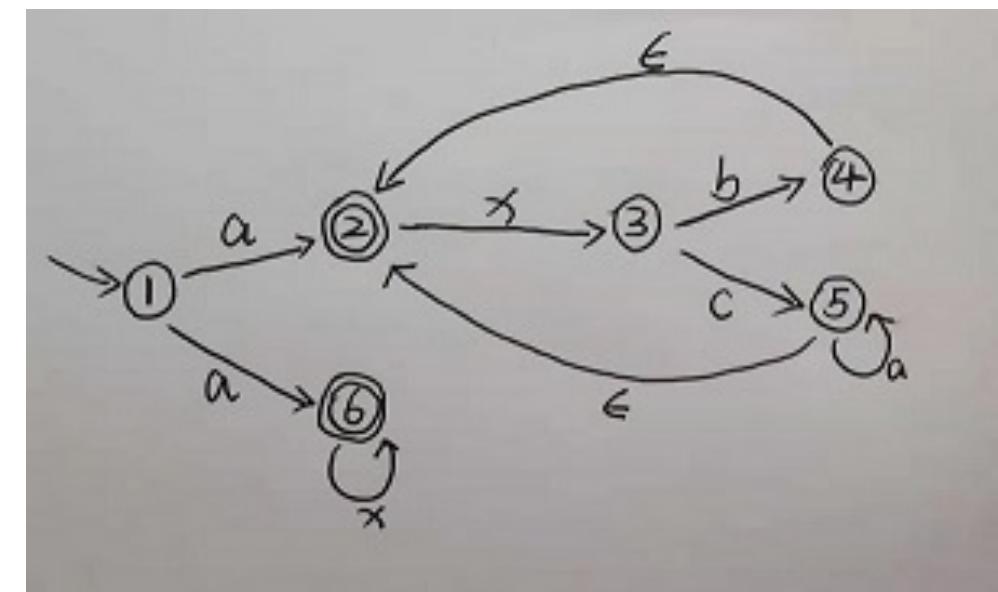
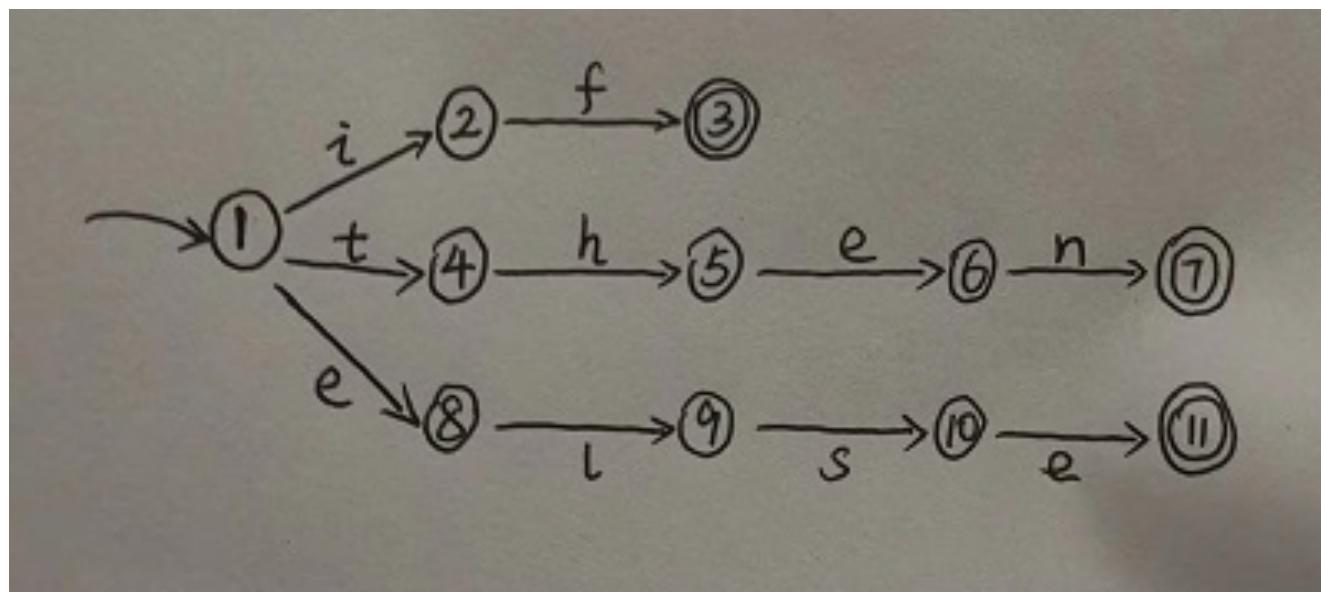
感谢孙月影同学提供本次作业；感谢李威威同学进行批注

当然是用第二课课件中提到的各种工具:P

2. 4 将下面两个正则表达式转换为非确定的有限自动机。

- a. (if|then|else)
- b. a((b|a*c)x)*|x*a

提交的一些答案（课堂讨论用，不一定是正确答案）

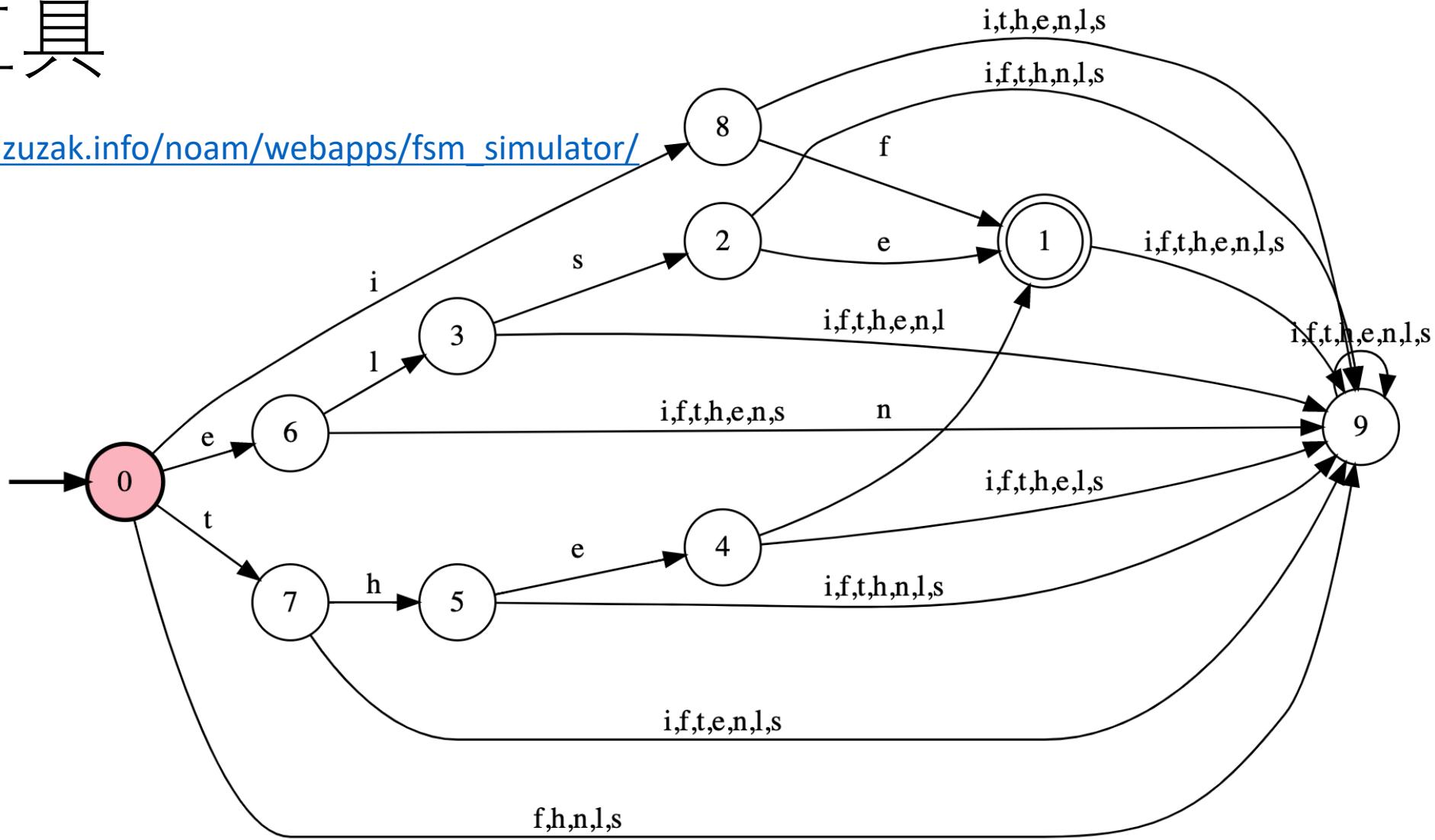


2.4 b) 识别字符串有问题，不过这个RE中|和连接的优先级哪个高我还有些不太确定，这个是识别axa还是只能识别xa？感觉应该是xa

感谢孙月影同学提供本次作业；感谢李威威同学进行批注

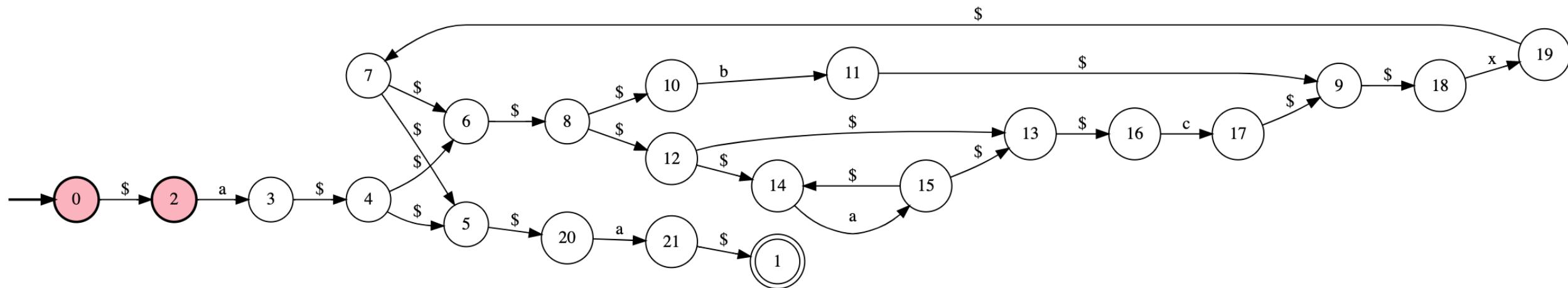
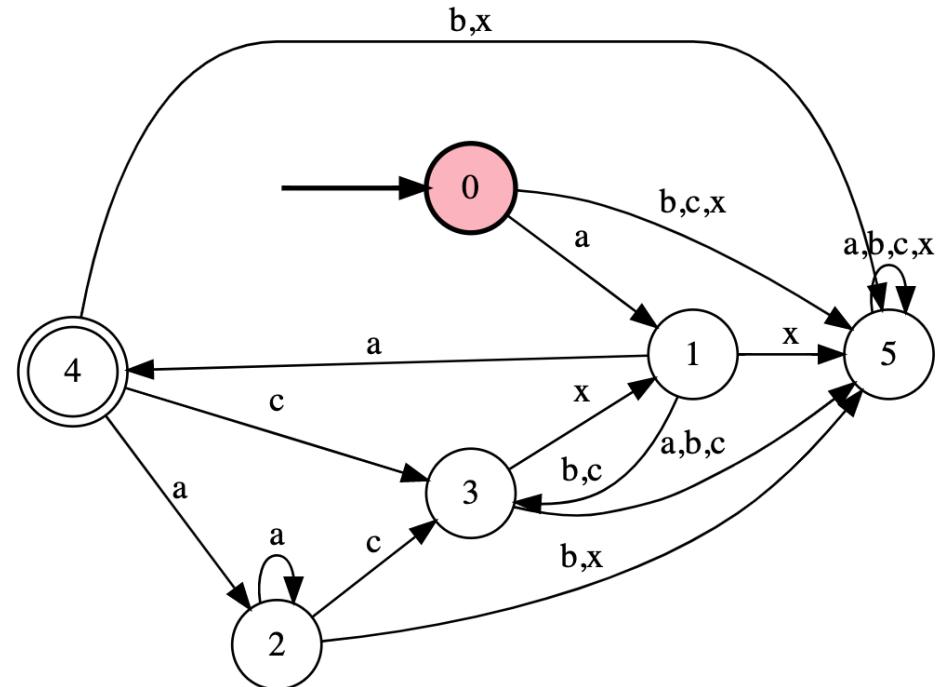
用工具

http://ivanzuzak.info/noam/webapps/fsm_simulator/

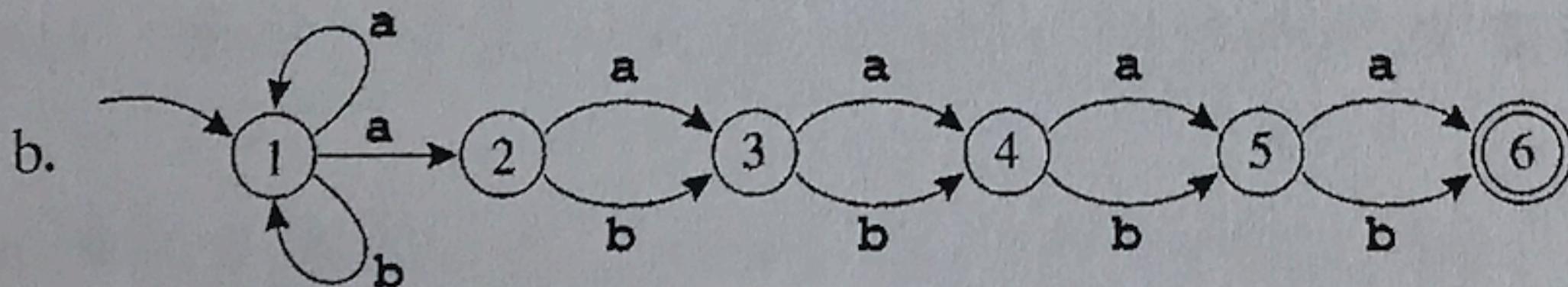
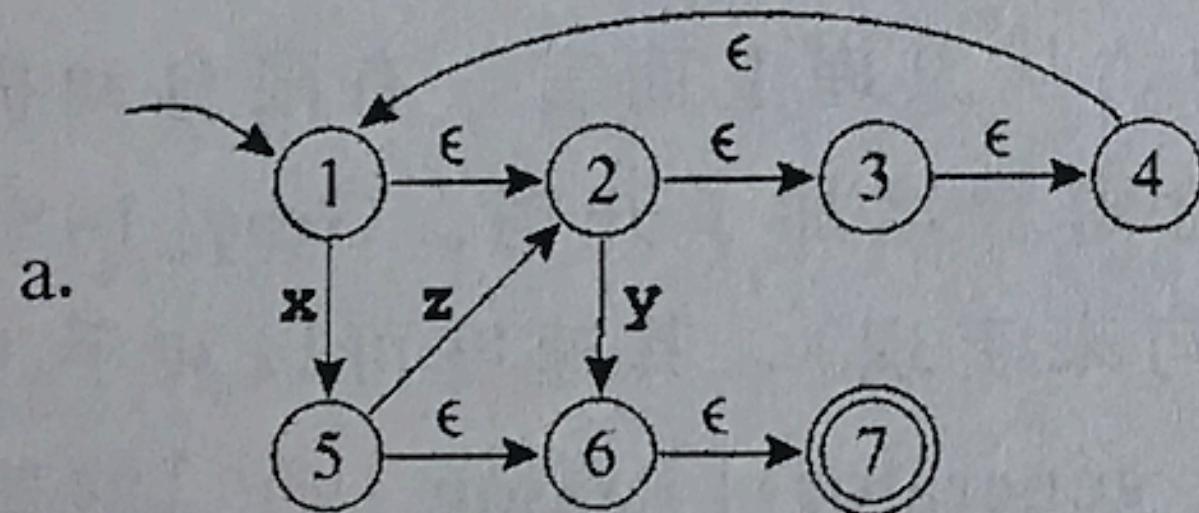


使用第二课中提到的各种
RE-NFA-DFA 转换工具

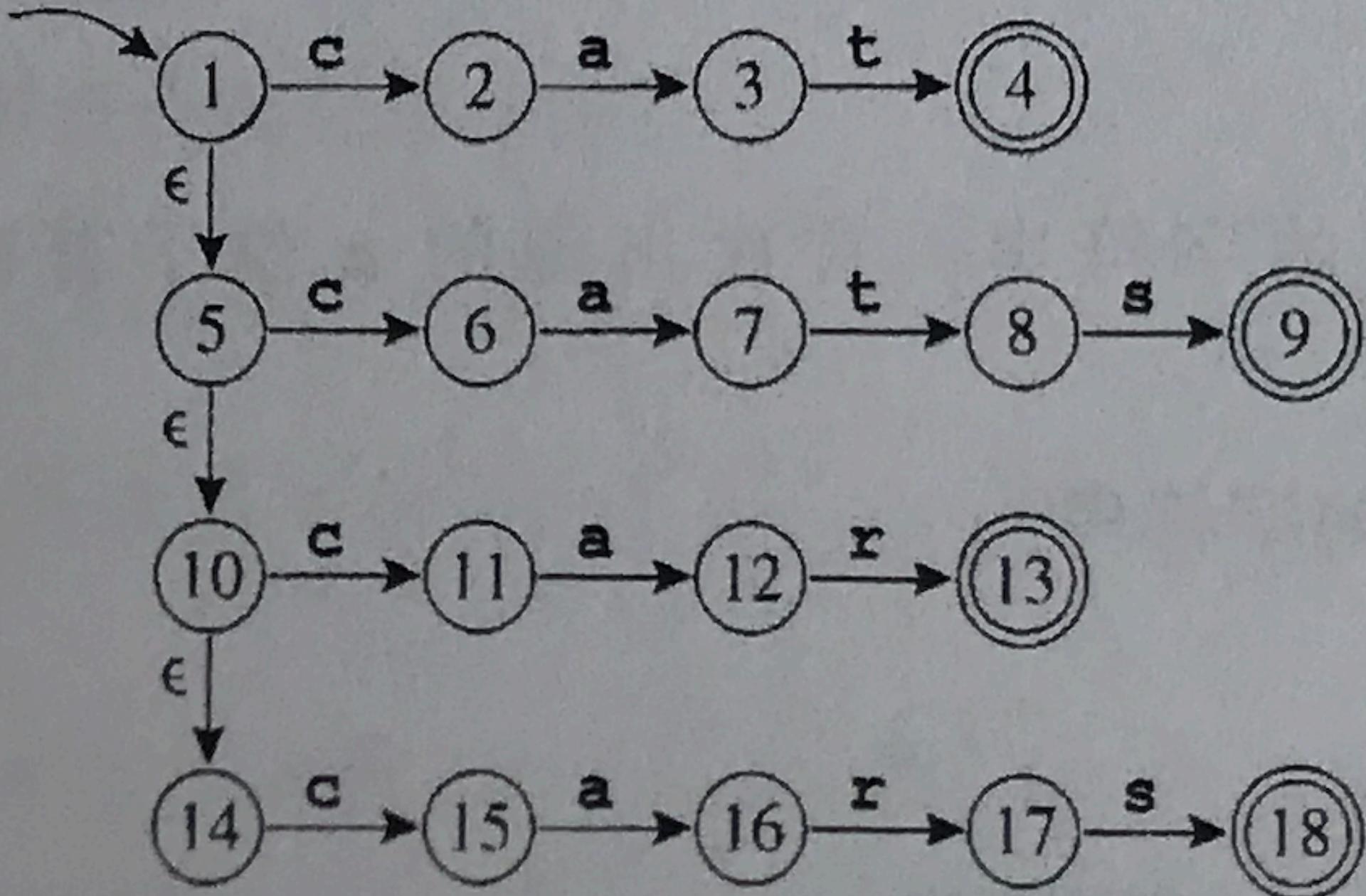
http://ivanzuzak.info/noam/webapps/fsm_simulator/



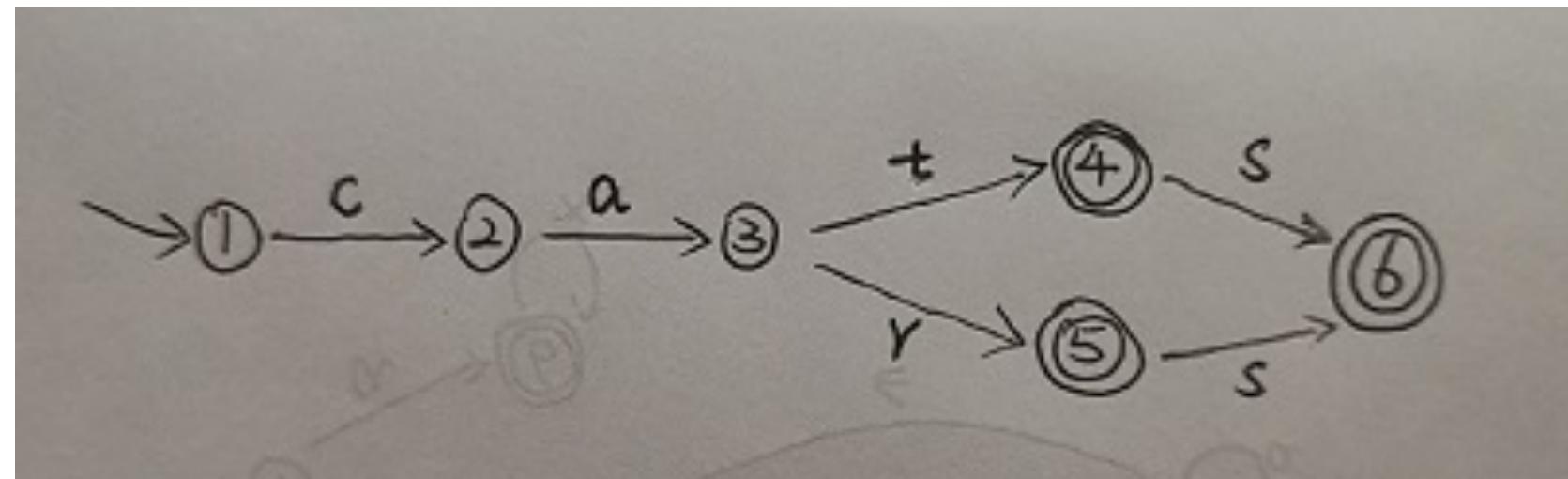
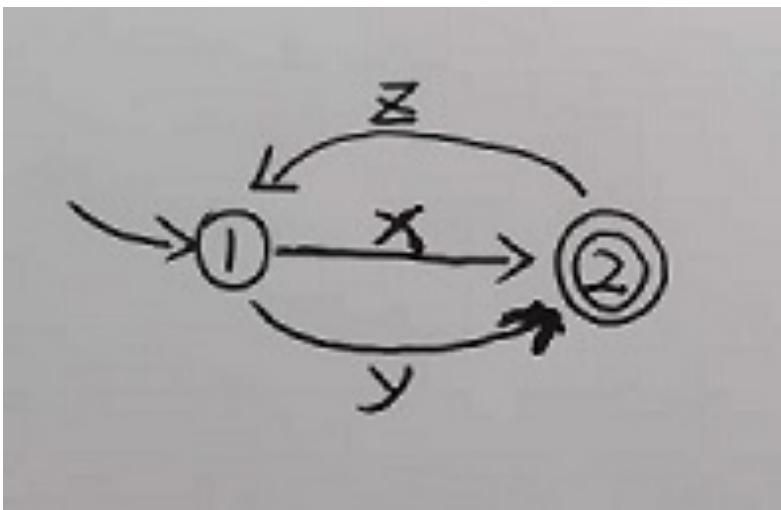
2.5 将下面的 NFA 转换为确定的有限自动机。



c.



提交的一些答案（课堂讨论用，不一定是正确答案）

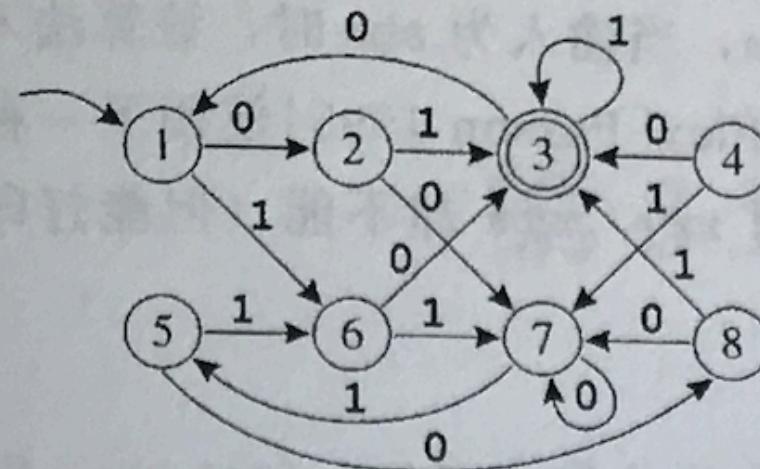


2.5 a) 应该是3个状态, $(1, 2, 3, 4) \text{ -x-} > ((5, 6, 7))$ $(1, 2, 3, 4) \text{ -y-} > ((6, 7))$
 $(5, 6, 7) \text{ -z-} > (1, 2, 3, 4)$

感谢孙月影同学提供本次作业；感谢李威威同学进行批注

2.6 在下面这个自动机中找出两个等价的状态，并合并它们产生一个识别相同语言且较小的自动机。重复这个过程直到没有等价状态为止。

不要求掌握，有兴趣可以做一下



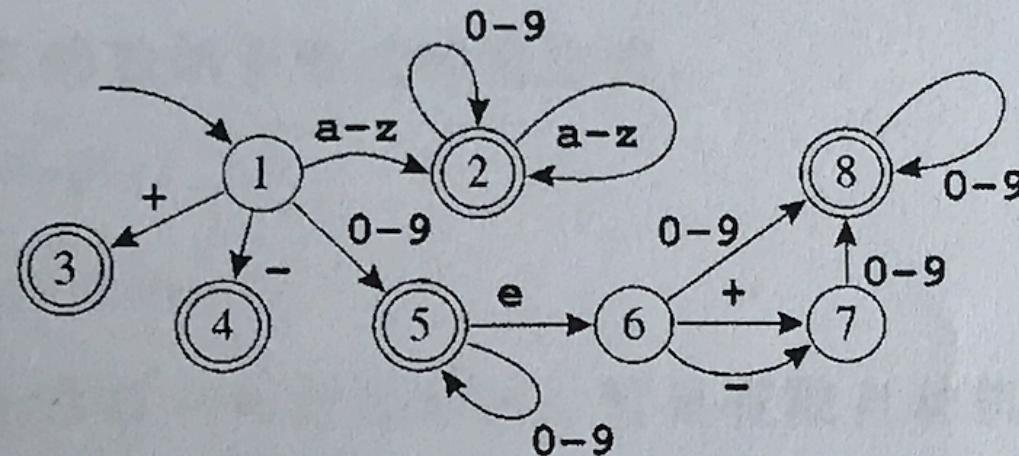
实际上，最小化有限自动机的通用算法是以相反的思路来工作的。它首先要找出的是所有不等价的状态偶对。若 X 是终结符而 Y 不是，或者（通过迭代） $X \xrightarrow{a} X'$ 且 $Y \xrightarrow{a} Y'$ 但 X' 和 Y' 不等价，则状态 X 和 Y 不等价。用这种迭代方式寻找新的不等价状态偶对且由于没有更多的不等价状态而停止后，如果 X 、 Y 仍不是不等价偶对，则它们就是等价状态。参见 Hopcroft 和 Ullman[1979]中的定理 3.10。

不要求掌握，有兴趣可以做一下

- *2.7 任何接收至少一个字符串的 DFA 都能转换为一个正则表达式。将习题 2.3c 的 DFA 转换为正则表达式。提示：首先假装状态 1 是初态。然后，编写一个通到状态 2 并返回到状态 1 的正则表达式和一个类似的通到状态 0 并返回到状态 1 的正则表达式。或者查看 Hopcroft 和 Ullman[1979]一书中定理 2.4 关于此算法的论述。

不要求掌握，有兴趣可以做一下

*2.8 假设 Lex 使用下面这个 DFA 来寻找输入文件中的单词：



- Lex 在匹配一个单词之前，必须在该单词之后再检测多少个字符？
- 设你对问题 a 的答案为 k ，写出一个至少包含两个单词的输入文件，使得 Lex 的第一次调用在返回第一个单词前需要检测该单词末尾之后的 k 个字符。若对问题 a 的答案为 0，则写出一个包含至少两个单词的输入文件，并指出每个单词的结束点。

不要求掌握，有兴趣可以做一下

2.9 一个基于 DFA 的解释型词法分析器使用以下两张表。

- edges 以状态和输入符号为索引，产生一个状态号。
- final 以状态为索引，返回 0 或一个动作号。

从下面这个词法规范开始：

(aba) +	(action 1);
(a(b*)a)	(action 2);
(a b)	(action 3);

为一个词法分析器生成 edges 和 final 表。

然后给出该词法分析器分析字符串 abaabbaba 的每一步。注意，一定要给出此词法分析器重要的内部变量的值。该词法分析器将被反复调用以获得后继的单词。

编译技术课程的讨论注意事项

- 就像是RE到NFA有很多种转换方法，**答案不唯一**
- 很多时候**需要**讨论来确定自己的答案和想法是否正确
- 不要盲信：包括我和所有助教/学生的答案可能都是错误的
- 遇到以上情况，**有任何不确定的感觉，直接issues里贴出来问**
 - 说不定还能帮助到同样有问题却不好意思发帖的人 ☺

新同学：可以加入本课程的微信学习群

- 群有人数限制，请在HelloGCC微信公众号输入「**旁听**」



方舟·编译技术入门与实战

！编程作业一定要写， 编译技术是看不会的！

有不明白的地方就及时提问😊

<https://github.com/lazyparser/becoming-a-compiler-engineer>