

习题课 - H2-1

Zevin

2.3 (c)

- 叙述有下列正规式描述的语言：

$$(0|1)^* 0 (0|1) (0|1)$$

2.3 (c)

- 叙述有下列正规式描述的语言：

$$(0|1)^* 0 (0|1) (0|1)$$

- 由字符0, 1组成的, 以 000、001、010或011结尾的字符串

或者：

2.3 (c)

- 叙述有下列正规式描述的语言：

$$(0|1)^* 0 (0|1) (0|1)$$

- 由字符0, 1组成的, 以 000、001、010或011结尾的字符串

或者：

- 由字符0, 1组成的, 倒数第三个字符为0的字符串

2.4(c)

- 某语言的注释, 它是以/*开始并以*/结束的任意字符串, 但它的任何前缀(本身除外)不以*/结尾

2.4(c)

- 某语言的注释, 它是以/*开始并以*/结束的任意字符串, 但它的任何前缀(本身除外)不以*/结尾
- 这道题的正确率很低.
- 一部分人的解答跟参考答案十分相似. (也许只是巧合)
- 其他跟参考答案截然不同的解答中:
 - 只看到一个完全正确的自己的解答. (我没发现错误)
 - 还有一个只差了一个括号就正确了
 - 其他都错了

2.4(c)

- 某语言的注释, 它是以/*开始并以*/结束的任意字符串, 但它的任何前缀(本身除外)不以*/结尾
- 对于配套的习题以及参考答案:
 - 我个人认为是有必要看的.
 - 但是在写作业的时候建议最后再看, 要有自己的理解.
 - 为什么要看呢? 甚至为什么要做其中额外的习题呢?
 - 因为考试的比重还是很大的, 不额外做题不思考很难拿高分.
 - 尤其是后面的文法部分.

2.4(c)

- 某语言的注释, 它是以/*开始并以*/结束的任意字符串, 但它的任何前缀(本身除外)不以*/结尾
- 怎么做呢?
- $\text{char} \rightarrow a \mid b \mid \dots$
注: 除了*、/以外, 该右部的选择包括该语言其他所有字符
- $\text{string} \rightarrow \text{char}^+$
注: 由*、/以外的字符组成的长度大于0的字符串

- 某语言的注释, 它是以/*开始并以*/结束的任意字符串, 但它的任何前缀(本身除外)不以*/结尾

- 怎么做呢?

- char → a | b | ...

 - 注: 除了*、/以外, 该右部的选择包括该语言其他所有字符

- string → *char*⁺

 - 注: 由*、/以外的字符组成的长度大于0的字符串

- 关键点1在于不能让 / 出现在 * 的后面,

 - 这就要保证中间的*后面必须是string或者* (除了结尾)

 - 这个可以用*⁺ string, 结尾可以 *⁺/

- 关键点2在于一开始的/*/*/*中/是可以在这个*后面的.

 - 这可以 (/ string), 但是 / 后面可以没有 string 那就(/ string?)

- 那就

 - /* ((/ string?) | (*⁺ string))^{*} *⁺/ 这样吗?

- 不行, /* 后面可以有 string, 那就

 - comment → /* string? ((/ string?) | (*⁺ string))^{*} *⁺/

- 某语言的注释, 它是以`/*`开始并以`*/`结束的任意字符串, 但它的任何前缀(本身除外)不以`*/`结尾
- 所以最后:
- `char` → `a | b | ...`
注: 除了`*`、`/`以外, 该右部的选择包括该语言其他所有字符
- `string` → `char+`
注: 由`*`、`/`以外的字符组成的长度大于0的字符串
- `comment` → `/* string? ((/ string?) | (*+ string))* *+/`
- 这个解答参考了ln同学的解答, 比较好理解.
- 不过ta弄错了 `|` 的优先级, 少了个括号.

2.4(c)

- 某语言的注释, 它是以/*开始并以*/结束的任意字符串, 但它的任何前缀(本身除外)不以*/结尾
- 参考答案:
- other → a | b | ...
注: 除了*以外, 该右部的选择包括该语言其他所有字符
- other1 → a | b | ...
注: 除了*、/以外, 该右部的选择包括该语言其他所有字符
- comment → / * other* (* ** other1 other*)* ** * /

2.4(e)

- 最多只有一处相邻数字相同的所有数字串

2.4 (e)

- 最多只有一处相邻数字相同的所有数字串
- 首先考虑相邻数字都不相同的串

2.4 (e)

- 最多只有一处相邻数字相同的所有数字串
- 首先考虑相邻数字都不相同的串
- 然后从两个数字开始，用归纳的方法扩展到十个数字

2.4(e)

- 最多只有一处相邻数字相同的所有数字串
- 首先考虑相邻数字都不相同的串
- 然后从两个数字开始，用归纳的方法扩展到十个数字
- 记相邻数字都不相同的串为 S ，则所求为 SS

2.4 (e) 最多只有一处相邻数字相同的所有数字串

- 首先考虑相邻数字都不相同的串
- 然后从两个数字开始，用归纳的方法扩展到十个数字

- 从0开始,
- $S_0 \rightarrow 0$
 - 因为 S_0 只有一个字符, 所以0自己就是相邻数字都不相同的串.

2.4 (e) 最多只有一处相邻数字相同的所有数字串

- 首先考虑相邻数字都不相同的串
- 然后从两个数字开始，用归纳的方法扩展到十个数字

- $S_0 \rightarrow 0$
- 然后是0, 1
- $S_1 \rightarrow S_0 \mid S_0? 1 (S_0 1)^* S_0?$
 - 为了归纳下去，所以要用统一的格式
 - 然后这里就是两种情况：
 - 1. 不含新的1;
 - 2. 含新的1.
 - 含的话，可能是1开头，可能是 S_0 开头，所以 $S_0?1$,
 - 之后自然地需要 $(S_0 1)^*$ ，因为不能相邻. 结尾就自然地 $S_0?$

2.4 (e) 最多只有一处相邻数字相同的所有数字串

- 首先考虑相邻数字都不相同的串
- 然后从两个数字开始，用归纳的方法扩展到十个数字
- 记相邻数字都不相同的串为S，则所求为SS

- $S_0 \rightarrow 0$
- $S_1 \rightarrow S_0 \mid S_0? 1 (S_0 1)^* S_0?$
- 之后递归...
- $S_9 \rightarrow S_8 \mid S_8? 9 (S_8 9)^* S_8?$

2.4 (e) 最多只有一处相邻数字相同的所有数字串

- 首先考虑相邻数字都不相同的串
- 然后从两个数字开始，用归纳的方法扩展到十个数字
- 记相邻数字都不相同的串为S，则所求为SS

• $S_0 \rightarrow 0$

• $S_1 \rightarrow S_0 \mid S_0? 1 (S_0 1)^* S_0?$

• 之后递归...

• $S_9 \rightarrow S_8 \mid S_8? 9 (S_8 9)^* S_8?$

• RESULT $\rightarrow S_9 S_9$

2.4 (e)

最多只有一处相邻数字相同的所有数字串

- $S_0 \rightarrow 0$
- $S_1 \rightarrow S_0 \mid S_0? 1 (S_0 1)^* S_0?$
- 之后递归...
- $S_9 \rightarrow S_8 \mid S_8? 9 (S_8 9)^* S_8?$

- RESULT $\rightarrow S_9 S_9$

错啦

2.4 (e)

最多只有一处相邻数字相同的所有数字串

- $S_0 \rightarrow 0$
- $S_1 \rightarrow S_0 \mid S_0? 1 (S_0 1)^* S_0?$
- 之后递归...
- $S_9 \rightarrow S_8 \mid S_8? 9 (S_8 9)^* S_8?$

- RESULT $\rightarrow S_9 S_9$

- 没有考虑数字串长度的问题!
- 长度为0、1应该也符合鸭!

2.4 (e)

最多只有一处相邻数字相同的所有数字串

- $S_0 \rightarrow 0$
- $S_1 \rightarrow S_0 \mid S_0? 1 (S_0 1)^* S_0?$
- 之后递归...
- $S_9 \rightarrow S_8 \mid S_8? 9 (S_8 9)^* S_8? \mid \epsilon$

- RESULT $\rightarrow S_9 S_9$

- 这回对了

2.7(c)

- 用算法2.4用下列正规式构造NFA, 给出处理输入串ababbab的状态转换序列.
 - (手工构造NFA和DFA, 用算法将NFA变换成DFA, 再构造最简的DFA)
 - $((\varepsilon | a) b^*)^*$
- 大家写的都还好, 个别同学步骤太简单了. 最好把转换表等写出来.
- 有些同学没有完全按照算法2.4来, 不过也是对的.
- 这题很简单, 按照书上的算法一步一步来就ok了.
 - 构造NFA和DFA按照算法2.4、算法2.1
 - NFA到DFA的转换按照算法2.2
 - 化简DFA参考算法2.3

Thanks

Zevin