

Slide07 必做题

Exercise 6.2.1(b)

参考解答:

构造以终态方式接受的 PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, 其中

$Q = \{q_0\}$; 状态 q_0 表示当前扫描过的输入串的任何前缀中 1 的个数不比 0 的个数多;

$\Sigma = \{0, 1\}$;

$\Gamma = \{Z_0, X\}$; 下推栈中, X 的个数表示当前扫描过的输入串中 0 的个数比 1 的个数多出来的数目;

$F = \{q_0\}$;

$\delta(q_0, 0, Z_0) = \{(q_0, X Z_0)\}$, $\delta(q_0, 0, X) = \{(q_0, X X)\}$, $\delta(q_0, 1, X) = \{(q_0, \varepsilon)\}$.

Exercise 6.2.1(c)

参考解答:

构造以空栈方式接受的 PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$, 其中

$Q = \{q_0, q_1\}$; 状态 q_0 表示当前扫描过的输入串的任何前缀中 0 的个数不少于 1 的个数, 状态 q_1 表示当前扫描过的输入串的任何前缀中 1 的个数不少于 0 的个数;

$\Sigma = \{0, 1\}$;

$\Gamma = \{Z_0, X\}$; 下推栈中, X 的个数表示当前扫描过的输入串中 0 的个数比 1 的个数或 1 的个数比 0 的个数多出来的数目;

$\delta(q_0, 0, Z_0) = \{(q_0, X Z_0)\}$, $\delta(q_0, 1, Z_0) = \{(q_1, X Z_0)\}$;

$\delta(q_1, 0, Z_0) = \{(q_0, X Z_0)\}$, $\delta(q_1, 1, Z_0) = \{(q_1, X Z_0)\}$;

$$\delta(q_0, 0, X) = \{(q_0, XX)\}, \delta(q_0, 1, X) = \{(q_0, \varepsilon)\};$$

$$\delta(q_1, 0, X) = \{(q_1, \varepsilon)\}, \delta(q_1, 1, X) = \{(q_1, XX)\};$$

$$\delta(q_0, \varepsilon, Z_0) = \{(q_0, \varepsilon)\}, \delta(q_1, \varepsilon, Z_0) = \{(q_1, \varepsilon)\}.$$

Exercise 6.2.6

- a) 把 P 转换为另外一个 PDA P_1 , 使得 P_1 能够以空栈方式接受 P 以终结状态方式接受的同样的语言, 即 $N(P_1) = L(P)$ 。
- b) 找出一个 PDA P_2 使得 $L(P_2) = N(P)$, 也就是说 P_2 能够以以终结状态方式接受 P 以空栈方式接受的同样的语言。

参考解答:

- a) 转换为 PDA $P_1 = (\{q, p\}, \{0, 1\}, \{Z_0, X\}, \delta, q, Z_0)$, 它的转移函数如下:

1. $\delta(q, 0, Z_0) = \{(q, XZ_0)\}.$
2. $\delta(q, 0, X) = \{(q, XX)\}.$
3. $\delta(q, 1, X) = \{(q, X)\}.$
4. $\delta(q, \varepsilon, X) = \{(p, \varepsilon)\}.$
5. $\delta(p, \varepsilon, X) = \{(p, \varepsilon)\}.$
6. $\delta(p, 1, X) = \{(p, XX)\}.$
7. $\delta(p, 1, Z_0) = \{(p, \varepsilon)\}.$
8. $\delta(p, \varepsilon, Z_0) = \{(p, \varepsilon)\}.$
9. $\delta(p, \varepsilon, X) = \{(p, \varepsilon)\}.$

b) 转换为 PDA $P_2 = (\{p_0, p_f, q, p\}, \{0, 1\}, \{Z_0, X, X_0\}, \delta, q, Z_0, \{p_f\})$, 它的转移函数如下:

1. $\delta(p_0, \varepsilon, X_0) = \{(q, Z_0 X_0)\}$ 。

2. $\delta(q, 0, Z_0) = \{(q, XZ_0)\}$ 。

3. $\delta(q, 0, X) = \{(q, XX)\}$ 。

4. $\delta(q, 1, X) = \{(q, X)\}$ 。

5. $\delta(q, \varepsilon, X) = \{(p, \varepsilon)\}$ 。

6. $\delta(p, \varepsilon, X) = \{(p, \varepsilon)\}$ 。

7. $\delta(p, 1, X) = \{(p, XX)\}$ 。

8. $\delta(p, 1, Z_0) = \{(p, \varepsilon)\}$ 。

9. $\delta(q, \varepsilon, X_0) = \{(p_f, \varepsilon)\}$ 。

10. $\delta(p, \varepsilon, X_0) = \{(p_f, \varepsilon)\}$ 。

第七讲思考题

!Exercise 6.2.2 (b)

参考解答:

一种设计思路: 因为可接受的字符串中 0 的数目是 1 的两倍, 可以通过对栈的操作把 1 的数目变为字符串中 1 的两倍, 从而可以简化设计。

具体地, q_0 状态时, 每次输入 0, 如果栈顶为 1, 则将此 1 pop 掉, 状态不

变；如果栈顶为 0，则一个 0 入栈。每次输入 1，如果栈顶为 1，则 11 入栈；如果栈顶为 0，那么需要 pop 两个 0，但每次操作最多能 pop 一个元素，所以增加辅助状态 q_1 ，从 q_0 先 pop 一个 0 进入状态 q_1 ，这里栈顶其实只有两种情况，0 或 Z（不可能出现 1），如果为 0 则将 0 pop 掉，回到状态 q_0 ，如果为 Z，则将剩下的一个 1 入栈。如果在 q_0 时栈顶为 Z，那说明输入字符串满足题设条件，便可以 ϵ 转移至终态 q_2 。

构造终态接受方式的 PDA(注：可以很容易转化为空栈接受的 PDA)如下：

PDA $P = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z\}, \delta, q_0, Z, \{q_2\})$

δ 函数定义如下：

$\delta(q_0, 0, Z) = \{ (q_0, 0Z) \}$

$\delta(q_0, 1, Z) = \{ (q_0, 11Z) \}$

$\delta(q_0, 0, 0) = \{ (q_0, 00) \}$

$\delta(q_0, 0, 1) = \{ (q_0, \epsilon) \}$

$\delta(q_0, 1, 1) = \{ (q_0, 111) \}$

$\delta(q_0, 1, 0) = \{ (q_1, \epsilon) \}$

$\delta(q_1, \epsilon, 0) = \{ (q_0, \epsilon) \}$

$\delta(q_1, \epsilon, Z) = \{ (q_0, 1Z) \}$

$\delta(q_0, \epsilon, Z) = \{ (q_2, Z) \}$

//注：如果改为 $\delta(q_0, \epsilon, Z) = (q_2, \epsilon)$ 则变为空栈接受

另一种思路：输入的 0、1 依次入栈。任何时候都保证栈顶的两个元素加上当前扫描的符号这三个符号中不构成 2 个 “0”，1 个 “1” 的情形。若出现这样的组合，则把这 3 个字符都去掉（即 pop 两个栈顶元素、当前扫描的符号不入栈，扫描下一个符号）。设计空栈接受方式的 PDA 即可。

构造以空栈方式接受的 PDA $P = (Q, \Sigma, \Gamma, \delta, q_{xx}, Z_0)$ ，其中

$Q = \{q_{xx}, q_{0x}, q_{1x}, q_{00}, q_{01}, q_{10}, q_{11}, p_0, p_1, r_0, r_1, r\}$; 状态 q_{01} 表示处于该状态时，栈顶的两个元素分别为 0 和 1（0 为栈顶元素，1 为栈顶第二个元素）， q_{00}, q_{10}, q_{11} 类似； q_{0x}, q_{1x} 分别表示当前堆栈只有 0 或 1 一个元素（ Z_0 除外）；而 q_{1x} 则表示当前堆栈中没有 0 或 1。进入 p_0 ，表示前一个扫描的符号和前一个栈顶元素中有一个 0 和一个 1；进入 p_1 ，表示前一个扫描的符号和前一个栈顶元素都是 0。进入状态 r ，表示刚处理完关于上述三个字符的组合（即把这 3 个字符都去掉，pop 两个栈顶元素、当前扫描的符号不入栈，扫描下一个符号）。进入 r_0 和 r_1 ，表示需要查看栈顶第二个元素。

$\Sigma = \{0, 1\}$;

$\Gamma = \{Z_0, 0, 1\}$;

δ 函数由下列转移图表示。注意：因为该图较大，所以将其分成了两个部分来绘制。

