**Compiler Principle:** 语法分析 **Spring 2025**

# Homework 5 — April 3

*Lecturer: Feng Xiaobing* *Completed by: 2022K8009929010 Zhang Jiawei*

**5.1**

当非终结符用某个选择匹配成功时，这种成功可能是暂时的。由于这种虚假现象的存在，故需要回溯技术。仅使用 FIRST 集合无法避免回溯，因为 LL(1) 文法的必要条件还包括：文法 $G$ 的任意两个产生式 $A \to \alpha|\beta$，若 $\beta$ 可以推导得到 $\varepsilon$，$FIRST(\alpha) \cap FOLLOW(A) = \varnothing$，反之亦然。

**5.2**

(1) 消除左递归之后的文法如下：

$$
\begin{aligned}
rexpr &\to rterm\ rexpr' \\
rexpr' &\to +\ rterm\ rexpr' \mid \varepsilon \\
rterm &\to rfactor\ rterm' \\
rterm' &\to rfactor\ rterm' \mid \varepsilon \\
rfactor &\to rprimary\ rfactor' \\
rfactor' &\to *\ rprimary\ rfactor' \mid \varepsilon \\
rprimary &\to a \mid b
\end{aligned}
$$

(2) FIRST 集合：

$$
\begin{aligned}
&FIRST(rexpr) = \{a, b\} \\
&FIRST(rexpr') = \{+, \varepsilon\} \\
&FIRST(rterm) = \{a, b\} \\
&FIRST(rterm') = \{a, b, \varepsilon\} \\
&FIRST(rfactor) = \{a, b\} \\
&FIRST(rfactor') = \{*, \varepsilon\} \\
&FIRST(rprimary) = \{a, b\}
\end{aligned}
$$

FOLLOW 集合：

$$
\begin{aligned}
&FOLLOW(rexpr) = FOLLOW(rexpr') = \{\$\} \\
&FOLLOW(rterm) = FOLLOW(rterm') = \{+, \$\} \\
&FOLLOW(rfactor) = FOLLOW(rfactor') = \{+, a, b, \$\} \\
&FOLLOW(rprimary) = \{+, *, a, b, \$\}
\end{aligned}
$$

(3) 考察其有两个生成式的 FIRST 和 FOLLOW 集合:

$$FIRST(+\ rterm\ rexpr') \cap FIRST(\varepsilon) = \{+\} \cap \{\varepsilon\} = \varnothing$$
$$FIRST(+\ rterm\ rexpr') \cap FOLLOW(rexpr') = \{+\} \cap \{\$\} = \varnothing$$

$$FIRST(rfactor\ rterm') \cap FIRST(\varepsilon) = \{a, b\} \cap \{\varepsilon\} = \varnothing$$
$$FIRST(rfactor\ rterm') \cap FOLLOW(rterm') = \{a, b\} \cap \{+, \$\} = \varnothing$$

$$FIRST(*rprimary\ rfactor') \cap FIRST(\varepsilon) = \{*\} \cap \{\varepsilon\} = \varnothing$$
$$FIRST(*rprimary\ rfactor') \cap FOLLOW(rfactor') = \{*\} \cap \{+, a, b, \$\} = \varnothing$$

$$FIRST(a) \cap FIRST(b) = \{a\} \cap \{b\} = \varnothing$$

由此可知,文法是 LL(1) 的。

(4) 构造 LL(1) 分析表如下:

| 非终结符 | 终结符 | | | | |
|---|---|---|---|---|---|
| | a | b | + | * | $ |
| rexpr | rexpr → rterm rexpr' | rexpr → rterm rexpr' | | | |
| rexpr' | | | rexpr' → + rterm rexpr' | | rexpr' → $\varepsilon$ |
| rterm | rterm → rfactor rterm' | rterm → rfactor rterm' | | | |
| rterm' | rterm' → rfactor rterm' | rterm' → rfactor rterm' | rterm' → $\varepsilon$ | | rterm' → $\varepsilon$ |
| rfactor | rfactor → rprimary rfactor' | rfactor → rprimary rfactor' | | | |
| rfactor' | rfactor' → $\varepsilon$ | rfactor' → $\varepsilon$ | rfactor' → $\varepsilon$ | rfactor' → * rprimary rfactor' | rfactor' → $\varepsilon$ |
| rprimary | rprimary → a | rprimary → b | | | |

(5) 对于输入串 $a + a * b + b * b$,分析过程如下(表格见下页):

| 栈 | 输入 | 动作 |
|---|---|---|
| $ | a + a * b + b * b $ | |
| $ rexpr | a + a * b + b * b $ | |
| $ rexpr’ rterm | a + a * b + b * b $ | rexpr → rterm rexpr’ |
| $ rexpr’ rterm’ rfactor | a + a * b + b * b $ | rterm → rfactor rterm’ |
| $ rexpr’ rterm’ rfactor’ rprimary | a + a * b + b * b $ | rfactor → rprimary rfactor’ |
| $ rexpr’ rterm’ rfactor’ a | a + a * b + b * b $ | match |
| $ rexpr’ rterm’ rfactor’ | + a * b + b * b $ | rfactor’ → $\varepsilon$ |
| $ rexpr’ rterm’ | + a * b + b * b $ | rterm’ → $\varepsilon$ |
| $ rexpr’ | + a * b + b * b $ | rexpr’ → + rterm rexpr’ |
| $ rexpr’ rterm + | + a * b + b * b $ | match |
| $ rexpr’ rterm | a * b + b * b $ | rterm → rfactor rterm’ |
| $ rexpr’ rterm’ rfactor | a * b + b * b $ | rfactor → rprimary rfactor’ |
| $ rexpr’ rterm’ rfactor’ rprimary | a * b + b * b $ | rprimary → a |
| $ rexpr’ rterm’ rfactor’ a | a * b + b * b $ | match |
| $ rexpr’ rterm’ rfactor’ | * b + b * b $ | rfactor’ → * rprimary rfactor’ |
| $ rexpr’ rterm’ rfactor’ rprimary * | * b + b * b $ | match |
| $ rexpr’ rterm’ rfactor’ rprimary | b + b * b $ | rprimary → b |
| $ rexpr’ rterm’ rfactor’ b | b + b * b $ | match |
| $ rexpr’ rterm’ rfactor’ | + b * b $ | rfactor’ → $\varepsilon$ |
| $ rexpr’ rterm’ | + b * b $ | rterm’ → $\varepsilon$ |
| $ rexpr’ | + b * b $ | rexpr’ → + rterm rexpr’ |
| $ rexpr’ rterm + | + b * b $ | match |
| $ rexpr’ rterm | b * b $ | rterm → rfactor rterm’ |
| $ rexpr’ rterm’ rfactor | b * b $ | rfactor → rprimary rfactor’ |
| $ rexpr’ rterm’ rfactor’ rprimary | b * b $ | rprimary → b |
| $ rexpr’ rterm’ rfactor’ b | b * b $ | match |
| $ rexpr’ rterm’ rfactor’ | * b $ | rfactor’ → * rprimary rfactor’ |
| $ rexpr’ rterm’ rfactor’ rprimary * | * b $ | match |
| $ rexpr’ rterm’ rfactor’ rprimary | b $ | rprimary → b |
| $ rexpr’ rterm’ rfactor’ b | b $ | match |
| $ rexpr’ rterm’ rfactor’ | $ | rfactor’ → $\varepsilon$ |
| $ rexpr’ rterm’ | $ | rterm’ → $\varepsilon$ |
| $ rexpr’ | $ | rexpr’ → $\varepsilon$ |
| $ | $ | match |