4.12 为文法
$S \to (L) \mid a$
$L \to L, S \mid S$

$(S, S, \cdots S)$

(b) 分别写出相应的语法制导定义、翻译方案以及预测翻译器，它打印出每个 a 在句子中是第几个字符。例如，当句子是 (a,(a,(a,a),(a))) 时，打印的结果是 2 5 8 10 14 。

(1)(2)(3)

为 文法符号 S 和 L 各一个继承属性 in 和 一个综合属性 out，in 表示在句子中 该文法符号 推出的字符序列的前面 已经有多少个 字符；out 表示 该文法符号推出的 字符序列 的最后一个字符在句子中是第几个字符.

因用到左部记述,属性从左→右,∴先继承属性

用的右部做,导完推出的字符序列看完了 才知道左部. out, ∴是综合属性

(1) 拓广文法,加上新的开始符号 S' 和新的产生式 S' → S

$S' \to S$ $S.in = 0$ // 初始化,计算的起点
$S \to (L)$ $L.in = S.in + 1$, $S.out = L.out + 1$
$S \to a$ $S.out = S.in + 1$, print($S.out$)
$L \to L_1, S$ $L_1.in = L.in$, $S.in = L_1.out + 1$, $L.out = S.out$
$L \to S$ $S.in = L.in$, $L.out = S.out$

书写要点: 每一个产生式的每个文法符号的每个属性都应该在对应的 语义规则中用到 (为它人赋值或 被赋值值)

(2)   ① $S' \to$ { $S.in = 0$; }
                S

    ② $S \to$ { $L.in = S.in + 1$; }
        (L)   { $S.out = L.out + 1$; }

    ③ $S \to a$ { $S.out = S.in + 1$; print ($S.out$) ; }

    ④ $L \to$ { $L_1.in = L.in$; }
        L_1,   { $S.in = L_1.out + 1$; }
        S   { $L.out = S.out$; }

    ⑤ $L \to$ { $S.in = L.in$; }
        S   { $L.out = S.out$; }

(3) 先将(2)中翻译方案消除左递归：

$S' \to S$
$S \to (L) \mid a$
$L \to SR$
$R \to , SR_1 \mid \varepsilon$

① $S' \longrightarrow$ $\{ S.in = 0; \}$
$\quad\quad\quad S$

② $S \longrightarrow$ $\{ L.in = S.in + 1; \}$
$\quad\quad (L)$ $\{ S.out = L.out + 1; \}$

③ $S \longrightarrow a$ $\{ S.out = S.in + 1;$ print $(S.out);\}$

④ $L \longrightarrow$ $\{ S.in = L.in; \}$
$\quad\quad S$ $\{ R.in = S.out; \}$
$\quad\quad R$ $\{ L.out = R.out; \}$

⑤ $R \longrightarrow ,$ $\{ S.in = R.in + 1; \}$
$\quad\quad S$ $\{ R_1.in = S.out; \}$
$\quad\quad R_1$ $\{ R.out = R_1.out; \}$

⑥ $R \to \varepsilon$ $\{ R.out = R.in; \}$

翻译函数代码见下页。

```
void S'(){
    int S_in,S_out;
    S_in=0;
    S_out=S(S_in);
    return;
}

int S(int S_in){
    int L_in,L_out,ret;//ret存返回值
    if(lookahead=='('){
        L_in=S_in+1;
        match('(');
        L_out=L(L_in);
        match(')');
        ret=L_out+1;
    }
    else{
        match('a');
        ret=S_in+1;
        print(ret);
    }
    return ret;

}


int L(int L_in){
    int S_in,S_out,R_in,R_out,ret;//ret存返回值
    S_in=L_in;
    S_out=S(S_in);
    R_in=S_out;
    R_out=R(R_in);
    ret=R_out;
    return ret;
}


int R(int R_in){
    int S_in,S_out,R1_in,R1_out,ret;//ret存返回值
    if(lookahead==','){
        match(',');
        S_in=R_in+1;
        S_out=S(S_in);
        R1_in=S_out;
        R1_out=R(R1_in);
        ret=R1_out;
    }
    else ret=R_in; //R->空　的情况
    return ret;
}
```