

一、单项选择题

1. 编译程序是一种（ A ）软件。
A. 系统 B.应用 C.MIS D.ERP
2. 下面对编译原理的有关概念描述正确的是（ D ）
A. 目标语言只能是机器语言
B. 编译程序处理的对象是源语言
C. Lex 是语法分析自动生成器
D. 解释程序属于编译程序
3. （ C ）不是编译程序的组成部分。
A. 词法分析程序 B.代码生成程序 C.设备管理程序 D.语法分析程序
4. 下面对编译程序分为“遍”描述正确的是（ A ）
A. 分“遍”可以使编译程序结构清晰
B. 可以提高程序的执行效率
C. 可以提高机器的执行效率
D. 可以增加对内容容量的要求
5. 下面对编译程序分“遍”应考虑的因素描述不正确的是（ C ）
A. 源语言的特征和约束 B.代码优化的因素
C. 编译程序的功能 D.目标代码的选择
6. 编译程序各阶段的工作都涉及到（ D ）
①表格管理 ②语法分析 ③出错处理 ④代码优化
A. ①② B. ②③ C. ③④ D. ①③
7. 设有表达式 $a*b-c$ ，将其中 $a*b$ 识别为表达式的编译阶段是（ B ）
A.词法分析 B.语法分析 C.语义分析 D.代码生成
8. BNF 是一种广采用的（ C ）的工具。
A. 描述规则
B. 描述语言
C. 描述文法
D. 描述句子
9. 无符号常数的识别和拼数工作通常在（ B ）阶段完成。
A. 词法分析
B. 语法分析
C. 语义分析
D. 代码生成
10. “运算符与运算对象类型不匹配”属于（ B ）。
A. 语法错误
B. 语义错误
C. 语用错误
D. 词法错误
11. 一遍扫描的编译程序的优点是（ D ）。
A. 算法清晰
B. 便于分工
C. 便于优化
D. 编译速度快
12. 编译程序生成的目标代码程序（ A ）是可执行的程序。

- A. 不一定
B. 一定
C. 一定不
D. 必须经链接装配后
13. 编译程序是对 (D)。
- A. 汇编程序的翻译
B. 高级语言的解释执行
C. 机器语言的执行
D. 高级语言的翻译
14. 测试一个编译程序时使用的测试数据是 (A)。
- A. 源程序
B. 中间代码
C. 目标代码
D. 任意数据
15. 由“非终结符 \rightarrow 符号串”形式的规则构成的文法是 (C)。
- A. 0 型文法
B. 1 型文法
C. 2 型文法
D. 3 型文法
16. 文法识别符号经过任意步推导得到的结果是 (A)。
- A. 句型
B. 句柄
C. 句子
D. 短语
17. * 关于短语和句柄, 正确的描述是 (B)。
- A. 短语就是句柄
B. 直接短语才可能是句柄
C. 最左短语一定是句柄
D. 最右短语一定是句柄
18. 一个语言的文法是 (C)。
- A. 唯一的
B. 不唯一的
C. 个数有限的
D. 无数个
19. 文法 G 所描述的语言是 (D) 集合。
- A. 文法 G 的字母表 V 中所有符号组成的符号串
B. 文法 G 的字母表 V 的闭包 V^* 中的所有符号串
C. 由文法的开始符号推出的所有符号串
D. 由文法的开始符号推出的所有的终结符号串
20. 文法分为四种类型: 0 型文法、1 型文法、2 型文法、3 型文法, 其中 3 型文法是 (B)。
- A. 短语文法
B. 正规文法
C. 上下文有关文法
D. 上下文无关文法

21. 一个上下文无关文法包含四个部分，一组非终结符，一组终结符，一个开始符号以及一组（ C ）。
- A. 句子
B. 句型
C. 产生式
D. 单词
22. 在编译中产生语法树是为了（ A ）。
- A. 语法分析
B. 语义分析
C. 词法分析
D. 目标代码生成
23. 如果一个文法无二义性，则它的任何句子（ A ）。
- A. 最左推导和最右推导对应的语法树必定相同
B. 最左推导和最右推导对应的语法树可能不同
C. 最左推导和最右推导必定相同
D. 可能存在两个不同的最左推导，但它们对应的语法树相同
24. 正则式（ D ）与 $(a^*b)^*(cld)$ 等价。
- A. $a^*(cld)lb(cld)$
B. $a^*(cld)^*lb(cld)^*$
C. $a^*(cld)lb^*(cld)$
D. $(alb)^*cl(alb)^*d$
25. 下述正规式中与 $(a^*b^*)(cld)$ 等价的是（ C ）。
- A. $a^*(cld)lb(cld)$
B. $a^*(cld)^*lb(cld)^*$
C. $a^*(cld)lb^*(cld)$
D. $(alb)^*cl(alb)^*d$
26. 同正则式 $(a|b)^*$ 等价的正则式是（ B ）。
- A. $(a|b)^*$
B. $(a|b)(a|b)^*$
C. $(ab)^*(ab)$
D. $(a|b)|(a|b)^*$
27. 同正规式 a^*b^* 等价的文法是（ C ）。
- A. $S \rightarrow aS | bS | \epsilon$
B. $S \rightarrow aSb | \epsilon$
C. $S \rightarrow aS | Sb | \epsilon$
D. $S \rightarrow aS | \epsilon$
28. 设正规式 $r=(a|b)(x|y)^*$ ，则下面错误的正规集元素是（ A ）。
- A. abx B. $bxxx$ C. a D. $bxyyxy$
29. 设有文法 $G(S)$ 为: $S \rightarrow AB | AS$ $A \rightarrow aA | a$ $B \rightarrow b$, 则下面与该文法等价的正规式是（ B ）。
- A. aa^*bb^* B. aa^*b C. $(ab)^*$ D. $a(ab)^*b$
30. 给定语言 $L(G) = \{a^n b b^n | n \geq 1\}$, 则文法（ D ）可产生语言 $L(G)$ 。
- A. $Z \rightarrow aZb | aAb | b, A \rightarrow aAb | b$
B. $A \rightarrow aAb | b$
C. $Z \rightarrow AbB, A \rightarrow aA | a, B \rightarrow bB | b$

- D. $Z \rightarrow aAb, A \rightarrow aAb|b$
31. 由文法识别符号通过若干步推导得到的终结符号串是 (C)。
- A. 语言
B. 句型
C. 句子
D. 句柄
32. 设文法 $G = (\{S\}, \{0,1\}, P, S)$, 其中 $P = \{S \rightarrow SS | 0S1 | 1S0 | \epsilon\}$, 该文法所描述的语言为 (B)。
- A. $\{0^n 1^n | n \geq 0\}$
B. $\{w | w \in \{0,1\}^* \text{ 且 } w \text{ 中 } 0 \text{ 和 } 1 \text{ 的个数相等}\}$
C. $\{0^m 1^k | m, k \geq 0\} \cup \{1^m 0^k | m, k \geq 0\}$
D. $\{0^n 1^n | n \geq 0\} \cup \{1^n 0^n | n \geq 0\}$
33. 词法分析所依据的是 (B)。
- A. 语法规则
B. 构词规则
C. 语义规则
D. 等价变换规则
34. 正规式 $M1$ 和 $M2$ 等价是指 (C)。
- A. $M1$ 和 $M2$ 的状态数相等
B. $M1$ 和 $M2$ 的有向弧条数相等
C. $M1$ 和 $M2$ 的所识别的语言集相等
D. $M1$ 和 $M2$ 的状态数和有向弧条数相等
35. 状态转换图 (见图 3-6-1) 接受的字集为 (D)。

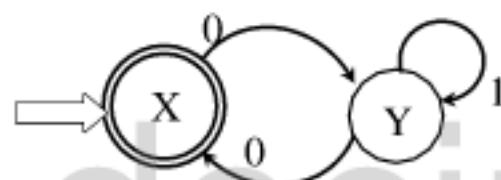


图 3-6-1

- A. 以 0 开头的二进制数组成的集合 B. 以 0 结尾的二进制数组成的集合
C. 含奇数个 0 的二进制数组成的集合 D. 含偶数个 0 的二进制数组成的集合
36. 词法分析器作为独立的阶段使整个编译程序结构更加简洁、明确, 因此 (B)。
- A. 词法分析器应作为独立的一遍
B. 词法分析器作为子程序较好
C. 词法分析器分解为多个过程, 由语法分析器选择使用
D. 词法分析器并不作为一个独立的阶段
37. 设有 C 语句的程序段如下:

```
while(i&&++j)
{
    c=2.19;
    j+=k;
    i++;
}
```

则经过词法分析后可以识别的单词个数是 (B)

- A. 19 B. 20 C. 21 D. 23

38. 下列选项中, 不属于预处理程序要完成的功能的是(B)
- A. 滤掉源程序中的注释
 - B. 查找源程序中的无用字符
 - C. 进行宏替换
 - D. 实现文件包含的嵌入和条件编译的嵌入
39. 编译过程中扫描器的任务包括 (D)
- ① 组织源程序的输入
 - ② 按词法规则分割单词, 识别出其属性, 并转换成 token 串输出
 - ③ 删除注释
 - ④ 删除空格和无用字符
 - ⑤ 行计数, 列计数
 - ⑥ 发现并定位词法错误
 - ⑦ 建立符号表
- A. ②③④⑦
 - B. ②③④⑥⑦
 - C. ①②③④⑥⑦
 - D. ①②③④⑤⑥⑦
40. 将识别各类单词的有限自动机合并后得到的有限自动机 (A)
- A. 可能是 NFA 也可能是 DFA
 - B. 一定是 DFA
 - C. 一定是 NFA
 - D. 是最小的 DFA
41. 在词法分析阶段不能识别的是 (C)
- A. 标识符
 - B. 运算符
 - C. 四元式
 - D. 常数
42. 有穷自动机能够识别 (A)。
- A. 正则文法
 - B. 上下文无关文法
 - C. 上下文有关文法
 - D. 短语文法
43. 下列哪项不是 DFA 的构成成分 (B)
- A. 有穷字母表
 - B. 初始状态集合
 - C. 终止状态集合
 - D. 有限状态集合
44. 有限自动机 M 和 N 等价是指 (D)
- A. M 和 N 的字母表相同
 - B. M 和 N 的状态数和有向边数相等
 - C. M 和 N 的状态数或有向边数相等
 - D. M 和 N 识别的字符串集合相同
45. 下列哪项不是编译程序的组成部分 (B)。
- A. 词法分析程序
 - B. 设备管理程序
 - C. 代码生成程序
 - D. 语法分析程序
46. 某个语言, 它能用正则表达式表达, 但不能使用任何正则文法表达, 则这个语言 (D)。
- A. 含有二义性
 - B. 是 0 型文法所对应的语言
 - C. 既含有左递归又含有右递归
 - D. 不存在
47. 编译程序中的语法分析程序接受以 (A) 为单位的输入, 并产生有关信息供以后各阶段使用。

- A. 单词
 - B. 表达式
 - C. 规则
 - D. 语句
48. 递归子程序法属于（ A ）语法分析方法。
- A. 自顶向下
 - B. 自底向上
 - C. 自左至右
 - D. 自右向左
49. 语法分析方法中的 LL(1)分析法属于（ B ）分析方法。
- A. 自左至右
 - B. 自上而下
 - C. 自下而上
 - D. 自右至左
50. 采用确定的自顶向下分析时，必须（ C ）。
- A. 消除左递归
 - B. 消除右递归
 - C. 避免回溯
 - D. 提取左公因子
51. 自上而下语法分析的主要分析动作是（ A ）。
- A. 推导
 - B. 移进
 - C. 归约
 - D. 匹配
52. LL(1)文法（ B ）二义性的。
- A. 都是有
 - B. 都没有
 - C. 不一定都有
 - D. 极少具有
53. 一个字符属于 FOLLOW(S)，这个字符的含义是（ A ）。
- A. 一定会有一个句型中后随 S 的终结符
 - B. S 可能推导出的第一个字符
 - C. S 可能推导出的最后一个字符
 - D. 在某句型中直接跟在 S 后的字符
54. 在递归子程序方法中，若文法存在左递归，则会使分析过程产生（ D ）。
- A. 回溯
 - B. 非法调用
 - C. 有限次调用
 - D. 无限循环
55. 若一个正则式描述的正则集中的元素有无穷个，则其必然包含的运算是（ D ）。
- A. 连接运算 “ \cdot ”
 - B. 或运算 “ $|$ ”
 - C. 括号运算 “(” 和 “)”
 - D. 闭包运算 “ $*$ ”
56. 将编译程序分成若干“遍”是为了（ B ）。
- A. 提高程序的运行效率
 - B. 使程序的结构更加清晰

- C. 利用有限的机器内存并提高机器的运行效率
D. 利用有限的机器内存但降低了机器的运行效率
57. 构造编译程序应掌握 (D)。
- A. 源语言
B. 目标语言
C. 编译方法
D. 以上三项都是
58. 编译程序绝大多数时间花在 (D) 上。
- A. 出错处理
B. 词法分析
C. 目标代码生成
D. 表格管理
59. 词法分析器的输入是 (B)。
- A. 单词符号串
B. 源程序
C. 语法单位
D. 目标程序
60. 词法分析器的输出结果是 (C)。
- A. 单词的种别编码
B. 单词在符号表中的位置
C. 单词的种别编码和自身值
D. 单词自身值
61. 词法分析器用于识别 (C)。
- A. 字符串
B. 语句
C. 单词
D. 标识符
62. 语法分析器可以发现源程序中的 (D)。
- A. 语义错误
B. 语法和语义错误
C. 错误并校正
D. 语法错误
63. 下列关于解释程序的描述正确的是 (A)。
- A. 解释程序的特点是处理程序时不产生目标代码
B. 解释程序适用于 COBOL 和 FORTRAN 语言
C. 解释程序是为打开编译程序技术的僵局而开发的
D. 以上描述均不正确
64. 高级语言编译程序常用的语法分析方法中, 递归下降分析法属于 (B) 分析方法。
- A. 自左向右 B. 自顶向下 C. 自底向上 D. 自右向左
65. 设有文法 $G[I]: I \rightarrow I1 | I0 | Ia | Ic | a | b | c$ 下列符号串中是该文法句子的有 (B)。
- ① ab0 ② a0c01 ③ aaa ④ bc10
- 可选项有:
- A. ① B. ②③④ C. ③④ D. ①②③④
66. 下面不能用于对文法进行描述的是 (A)

- A.源语言 B.EBNF C.BNF D.语法图
67. 设有文法的符号集 V , 非终结符集 V_N , 终结符集 V_T , 下列叙述中正确的是 (D)
- A. $V=V_T$ B. $V=V_N$ C. $V=V_N \cap V_T$ D. $V=V_N \cup V_T$
68. 设有文法 $G(S)$ 为: $S \rightarrow 0A$ $A \rightarrow 1B$ $B \rightarrow 0|0S$ 则 $L(G(S))$ 为(B)
- A. $L_1=\{(01)^n 0 | n \geq 1\}$ B. $L_2=\{(010)^n | n \geq 1\}$
 C. $L_3=\{0(10)^n | n \geq 1\}$ D. $L_4=\{(010)^n | n \geq 0\}$
69. 设有文法 $G(S)$ 为: $S \rightarrow (B)a$ $B \rightarrow B|b| \epsilon$ 下列叙述错误的是(C)
- A. G 是 2 型文法 B. $L(G)=\{(b^n)a | n \geq 0\}$
 C. $L(G)=\{(b)^n a | n \geq 0\}$ D. 有文法 G' 为 $S \rightarrow ()a|(B)a$ $B \rightarrow bB|b$, 则 $G'=G$
70. 给定文法 $G(S)$ 为: $S \rightarrow 0S | 1A | 0$ $A \rightarrow 1 | 1S | 0B$ $B \rightarrow 1A | 0B$ 下列符号串是 $L(G)$ 的元素的是(A)
- A. 10100010011011 B. 0101001110010010
 C. 1101010011110111 D. 1010011101101010
71. 设有文法 $G(S): S \rightarrow S1 | S0 | Sa | Sc | a | b | c$ 下列符号串中不是该文法的句子的是 (A)
- A. ab0 B. a0c01 C. aaa D. bc01
72. 设定义在字母表 $\{a,b,c,x,y,z\}$ 上的正规式 $r=(a|b|c)(x|y|z)$, 则 $L(r)$ 中元素有 (A) 个。
- A. 9 B. 6 C. 18 D. 27
73. 文法 $G: S \rightarrow xSx|y$ 所识别的语言是 (C)。
- A. xyx B. $(xyx)^*$ C. $x^n y x^n (n \geq 0)$ D. $x^* y x^*$
74. 文法 G 描述的语言 $L(G)$ 是指 (B)。
- A. $L(G)=\{\alpha | S \xrightarrow{*} \alpha, \alpha \in V_T^*\}$ B. $L(G)=\{\alpha | S \xrightarrow{\epsilon} \alpha, \alpha \in V_T^*\}$
 C. $L(G)=\{\alpha | S \xrightarrow{*} \alpha, \alpha \in (V_T \cup V_N^*)\}$ D. $L(G)=\{\alpha | S \xrightarrow{*} \alpha, \alpha \in (V_T \cup V_N^*)\}$
75. 设文法为: $S \rightarrow SA|A$ $A \rightarrow a|b$
 则对句子 aba , 下面 (D) 是规范推导。
- A. $S \Rightarrow SA \Rightarrow SAA \Rightarrow AAA \Rightarrow aAA \Rightarrow abA \Rightarrow aba$
 B. $S \Rightarrow SA \Rightarrow SAA \Rightarrow AAA \Rightarrow AAa \Rightarrow Aba \Rightarrow aba$
 C. $S \Rightarrow SA \Rightarrow SAA \Rightarrow SAa \Rightarrow Sba \Rightarrow Aba \Rightarrow aba$
 D. $S \Rightarrow SA \Rightarrow Sa \Rightarrow SAa \Rightarrow Sba \Rightarrow Aba \Rightarrow aba$
76. 产生正规语言的文法为 (D)。
- A. 0 型 B. 1 型 C. 2 型 D. 3 型
77. * 在规范归约中, 用(B)来刻画可归约串。
- A. 直接短语 B. 句柄 C. 最左素短语 D. 素短语
78. 编译程序是一种 (B)。
- A. 汇编程序
 B. 翻译程序
 C. 解释程序
 D. 目标程序
79. 编译程序各阶段工作都涉及 (D)。
- A. 词法分析
 B. 语法分析
 C. 语义分析
 D. 表格管理
80. 经编译得到的目标程序是 (A)
- A. 机器语言程序或汇编语言程序

- B. 四元式序列
C. 三元式序列
D. 二元式序列
81. (B)不可能是目标代码。
A. 汇编代码
B. 中间代码
C. 绝对指令代码
D. 可重定位指令代码
82. 对于编译程序而言,输入数据是源程序,输出结果是 (C)。
A. 源程序
B. 中间代码
C. 目标程序
D. 汇编程序
83. 下面不属于 LL(1)分析器的组成部分的是 (D)
A. LL(1)总控程序 B. LL(1)分析表 C. 分析栈 D. 源程序串
84. 编译过程中,语法分析程序的任务是 (D)
①分析单词是怎样构成的②分析单词串是如何构成语句和说明的
③分析语句和说明是如何构成程序的④分析程序的结构
A. ①② B. ①②③ C. ①②④ D. ②③④

二、多项选择题

1. 句子 x 和句型 y 均可从无二义性文法 $G[S]$ 的识别符号 S 推导得到,则 (AD)。
A. 一定存在 S 到 x 的最左推导
B. 一定存在 S 到 y 的两种不同的推导
C. 一定存在 S 到 y 的最左推导
D. 一定存在唯一与 x 对应的语法树
2. 下面哪些说法是错误的 (AC)。
A. 有向图是一个状态转换图
B. 状态转换图是一个有向图
C. 有向图是一个 DFA
D. DFA 可以用状态转换图表示
3. 对无二义性文法来说,一棵语法树往往代表了 (ACE)。
A. 多种推导过程 B. 多种最左推导过程 C. 一种最左推导过程
D. 仅一种推导过程 E. 一种最右推导过程
4. 如果文法 G 存在一个句子,满足下列条件 (BCD) 之一时,则称该文法是二义文法。
A. 该句子的最左推导与最右推导相同
B. 该句子有两个不同的最左推导
C. 该句子有两棵不同的最右推导
D. 该句子有两棵不同的语法树
E. 该句子的语法树只有一个
5. 有一文法 $G: S \rightarrow AB$
 $A \rightarrow aAb | \epsilon$
 $B \rightarrow cBd | \epsilon$
它不产生下面 (AC) 集合。
A. $\{a^n b^m c^n d^m | n, m \geq 0\}$ B. $\{a^n b^n c^m d^m | n, m > 0\}$

- C. $\{a^n b^m c^m d^n | n, m \geq 0\}$ D. $\{a^n b^n c^m d^m | n, m \geq 0\}$
 E. $\{a^n b^n c^n d^n | n \geq 0\}$
6. 在词法分析中，能识别出（ACE）。
 A. 基本字 B. 四元式 C. 运算符
 D. 逆波兰式 E. 常数
7. 编译程序的生成方式可以是（ABD）。
 A. 自编译 B. 高级程序设计语言编写 C. 完全自动生成 D. 汇编语言编写
8. 令 $\Sigma = \{a, b\}$ ，则 Σ 上所有以 b 开头，后跟若干个 ab 的字的全体对应的正规式为（ABD）。
 A. $b(ab)^*$ B. $b(ab)^+$ C. $(ba)^*b$
 D. $(ba)^+b$ E. $b(alb)$

三、判断题

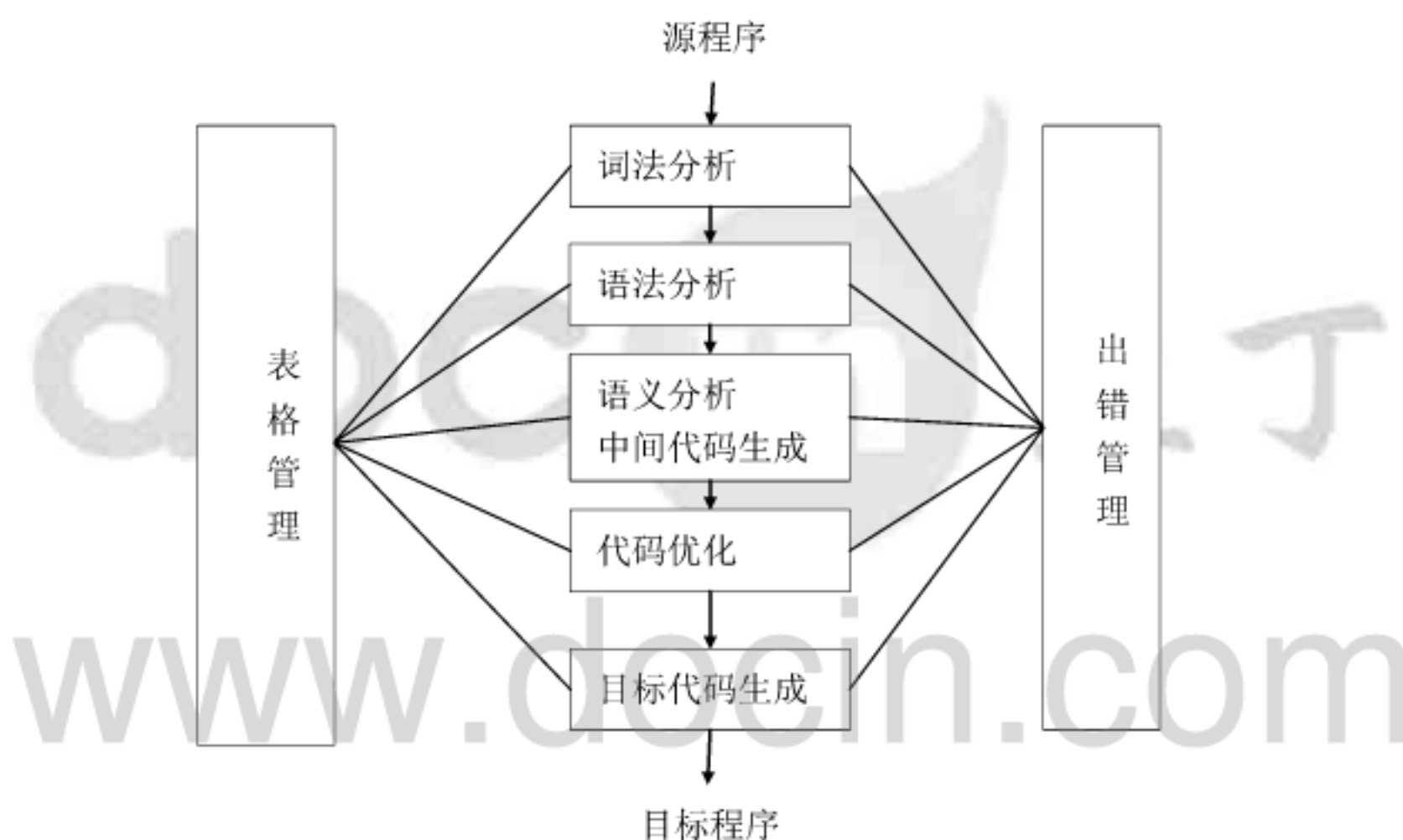
- 解释执行与编译执行的根本区别在于解释程序对源程序没有真正进行翻译。（×）
- “遍”是指对源程序的从头到尾扫描。（×）
- 编译程序是将用某一种程序设计语言编写的源程序翻译成等价的另一种语言程序（目标程序）。（×）
- 编译程序是应用软件。（×）
- 编译程序是系统软件。（√）
- 编译程序的基本组成中，词法分析、语法分析和语义分析应该是有序的。（√）
- 用高级语言书写的源程序都必须通过翻译，产生目标代码后才能运行。（√）
- 一个语言的文法是不唯一的。（√）
- BNF 是一种广泛采用的描述文法的工具。（√）
- 设有文法符号集 V ，则 $V = V_N \cup V_T$ 。（×）
- 设有文法符号集 V ，则 $V = V_N \cup V_T$ ，同时 $V_N \cap V_T = \emptyset$ 。（√）
- 一部文法 G 的文法符号不属于 V_N 就属于 V_T 。（√）
- 有文法 $G_1 = G_2$ ，则 $L(G_1) = L(G_2)$ 。（√）
- 源语言是描述另一种语言的语言。（√）
- 文法 G 所描述的语言就是 G 的终结符集 V_T 的闭包 V_T^* 。（×）
- 文法 G 所描述的语言就是 G 的终结符集 V_T 的闭包 V_T^* 的一个子集，子集是满足开始符号可以推导出的句子的集合。（√）
- 对于文法 $G(S) = (V_N, V_T, P, S)$ ， $V = V_N \cup V_T$ ， r 是文法 $G(S)$ 的句型当且仅当 $S \Rightarrow r$ ，且 $r \in V^*$ ； r 是文法 $G(S)$ 的句子当且仅当 $S \Rightarrow r$ ，且 $r \in V_T^*$ 。（√）
- 文法 G 的一个句子对应于多个推导，则 G 是二义的。（×）
- 文法 G 的一个句子对应于多个最左（或最右）推导，则 G 是二义的。（√）
- 对给定的文法 $G(S)$ ，若至少有一个句型存在两个或两个以上的不同的最左（或最右）推导，这是判定 G 是二义文法的充分非必要条件。（×）
- 对给定的文法 $G(S)$ ，若至少有一个句型存在两个或两个以上的不同的最左（或最右）推导，这是判定 G 是二义文法的充分必要条件。（√）
- 一棵分析树反映了其叶结点从左向右连接形成的句型的任意推导情况。（×）
说明：对应于二义性文法，二义性的句子有不只一棵分析树。
- NFA 和 DFA 的区别之一是映射函数是否唯一。（√）
- 一个正规式只能等价于一个确定的有限自动机。（×）
- 描述同一个正规式的确定的有限自动机可以有多个。（√）
- 任意有限自动机都能转化为一等价的特殊自动机，其状态图中初态无射入弧，终态无射

- 出弧。(√)
27. 对于任何正规集 L , 都有正规式 r , 满足 $L(r)=L$ 。(√)
28. 设有 r 和 s 都是非 ε 的正规式, 则有 $L(rs)=L(sr)$ 。(×)
说明: 正规式的连接运算不满足交换律。
29. 使用正规式运算能够描述定义在字母表上的所有符号串集合。(×)
30. 使用正规式运算能够描述定义在字母表上的正规集合。(√)
31. 正规式产生的语言都可以用上下文无关文法来描述。(√)
32. 正规式产生的语言都可以用正则文法来描述。(√)
说明: 正则文法包含于上下文无关文法。
33. 对任何正规式 r , 都存在一个 NFAM, 满足 $L(M)=L(r)$ 。(√)
34. 正则文法、正规式、NFA 和 DFA 在接受语言的能力上是相互等价的。(√)
35. 自动机 M_1 和 M_2 状态数不同, 则二者必不等价。(×)
36. 一个有限自动机至包含有限个状态, 其中一个被认为是初态, 最多只有一个终态。(×)
37. 一个有限自动机识别的语言是一个无限集合, 则该有限自动机的状态图一定含有回路。(√)
38. 如果一个有限自动机接受空串 ε , 则它的状态转换图一定含有 ε 弧。(×)
说明: 一个有限自动机的状态转换不含有 ε 弧, 但它的初态又属于终态集, 则该自动机接受空串 ε 。
39. 一个确定的有限自动机, 可以通过多条路识别一个符号串。(×)
40. 一个确定的有限自动机, 识别一个符号串的路径是唯一的。(√)
41. NFA 的确定化算法具有消除 ε 弧的功能。(√)
42. 正则文法一定不是二义文法。(×)
说明: 任何类型的文法都可以是二义文法。
43. 对每一个左线性文法 G_1 , 一定存在一个右线性文法 G_2 , 使得 $L(G_1)=L(G_2)$ 。(√)
44. 词法分析的依据是源语言的文法规则。(×)
45. LEX 是典型的词法分析程序。(×)
46. 源程序中的单词是具有独立意义的短语。(×)
47. 源程序中的单词是具有独立意义的最小语法单位。(√)
48. 单词的属性字一般应该包括单词类别和单词。(×)
49. 单词的属性字一般应该包括单词类别和单词值。(√)
50. 在 C 语言中有一个语句 `int int1;` 词法分析后识别出 `int`、`int`、`1` 和 `;` 四个单词。(×)
51. 在 C 语言中有一个语句 `int int1;` 词法分析后识别出 `int`、`int1` 和 `;` 3 个单词。(√)
52. 一个字母或数字在 C 语言中不一定是单词, 因为它们不一定具有独立的意义。(√)
53. 构造识别单词的有限自动机时, 先要对程序语言的单词按类构造出相应的有限自动机。(√)
54. 语法分析的任务是分析语句是如何构成程序的。(×)
55. 语法分析必须在语义分析之前完成。(√)
56. 自上而下语法分析实施的是最左推导。(√)
57. 设有文法 $G: S \rightarrow qQ \mid q \quad Q \rightarrow cQd \mid \varepsilon$, 该文法是 LL(1) 文法。(×)
58. LL(1) 分析法中的 “1” 是指扫描源程序当前指针 $P++$ 所指的源程序串的字符。(×)
59. LL(1) 分析法中的 “1” 表示在分析过程中, 每一步推导, 最多只要向前查看一个输入字符, 即能确定当前推导所应选用的文法规则。(√)
60. 自上而下分析及自下而上分析中的 “下” 指的是被分析的源程序串。(√)
61. 自上而下分析及自下而上分析中的 “上” 指的是分析树的根结点或文法的开始符号。(√)

62. LL(1)分析必须满足文法不含左公共因子和不含左递归。(✓)
63. 语法分析方法中的递归下降分析法属于自上而下分析方法。(✓)
64. 文法若存在左递归,则在自上而下语法分析过程中会因为假匹配造成算法的回溯。(×)
65. 文法若存在左递归,则在自上而下语法分析过程中会导致无止境的最左推导,使得分析算法陷入死循环。(✓)
66. LL(1)分析过程中使用的分析栈只能存放文法的终结符。(×)
67. LL(1)分析过程中使用的分析栈存放的是已经扫描且分析过的源程序串。(✓)
68. LL(1)分析的主要依据是 LL(1)分析表。(✓)
69. LL(1)中的“1”通过求 FIRST 得到。(×)
70. 设有文法中产生式 $T \rightarrow \epsilon$, 若采用递归下降分析法,则 T 对应的函数是空函数。(✓)

四、解答题

1. 请填写完整下图所示的编译程序的组成结构。



2. 设有文法 $G[S]: S \rightarrow Aa \quad A \rightarrow Bb \mid CD \quad C \rightarrow e$ 求 V_N 、 V_T 和该文法所描述的语言 $L(G)$ 。
 解: $V_N = \{S, A, C\}$
 $V_T = \{a, B, b, D, e\}$
 $L(G) = \{Bba, eDa\}$
3. 设文法 $G(S)$ 的 BNF 描述为:
 $S \rightarrow S, E \mid E$
 $E \rightarrow E + T \mid T$
 $T \rightarrow T * F \mid F$
 $F \rightarrow a \mid (E) \mid a[S]$
- ① 给出 $G(S)$ 的元语言符号集、文法符号集、终结符集和非终结符集。
 - ② $G(S)$ 属于哪类文法? 写出 $L(G(S))$ 集合。
 - ③ 判断符号串 $S1: a, a + a[a[S]]$ 和 $S2: a * a, a + a[a]$ 是否为文法 $G(S)$ 的句子, 对是 $L(G(S))$ 的句子给出相应的分析树。

解: (1) 元语言符号集: $\{\rightarrow, \mid\}$

$$V_N = \{S, E, T, F\}$$

$$V_T = \{., +, *, a, (,), [,]\}$$

$$V = V_N \cup V_T = \{S, E, T, F, ., +, *, a, (,), [,]\}$$

(2) $G(S)$ 属于 2 型文法。

$L(G) = \{\text{连续多个可用逗号连接的简单算术表达式, 表达式的运算对象由“+”或“*”连接, 运算对象可以为单一的“a”或带括号的表达式或 a[连续多个可用逗号连接的简单算术表达式]}\}$

(3) 由于 $S1:a, a+a[a[S]]$ 串中含有非终结符 S , 所以该符号串不是文法 $G(S)$ 的句子。

$S2:a*a, a+a[a]$ 是文法 $G(S)$ 的句子, 其对应的语法分析树如下 (略)

4. 若有 $\Sigma = \{0, 1\}$, 求 $\Sigma^* - \Sigma^+ = ?$, $\Sigma^{2*} = ?$

解: $\Sigma^* - \Sigma^+ = \epsilon$ $\Sigma^{2*} = \{00, 01, 10, 11\}$

5. 设有字母表 $A = \{0\}$, 求 A^* 。

解: $A^* = \{\epsilon, 0, 00, 000, 0000, \dots\}$

6. 令字母表 $A = \{0, 1, 2\}$ 上的字符串 $x=01$, $y=2$, $z=001$ 。

(1) 写出下列字符串及它们的长度: x^0 , xy , xyz , x^4 , $(x^3)(y^2)$, $(xy)^2$

(2) 写出集合 A^* 和 A^+ 的 7 个最短的符号串。

解: (1) $|x^0| = |z| = 0$ $|xy| = |012| = 3$ $|xyz| = |012001| = 6$

$|x^4| = |01010101| = 8$ $|(x^3)(y^2)| = |0101010122| = 8$ $|(xy)^2| = |012012| = 6$

(2) $A^* = \{\epsilon, 0, 1, 2, 00, 01, 02, \dots\}$

$A^+ = \{0, 1, 2, 00, 01, 02, 10, \dots\}$

7. 设文法 $G(Z)$ 为:

$$Z \rightarrow U0 \mid V1$$

$$U \rightarrow Z1 \mid 1$$

$$V \rightarrow Z0 \mid 0$$

(1) $G(Z)$ 的语言是什么

(2) 写出文法 $G(Z)$ 构造的长度为 6 的全部句子。

解: (1) $L(G) = \{01, 10\}^+$

(2) $\{010101, 010110, 011001, 011010, 101010, 100101, 100110, 101001\}$

8. 设有文法 $G[S]: S \rightarrow SS^* \mid SS+ \mid a$

(1) $G[S]$ 的语言 $L(G[S])$ 是什么?

(2) 下列字符串哪些是该文法的句子?

$S1: aa+aa^*+a$

$S2: aa+aaa^*++$

$S3: aS+a$

(3) 对属于该文法的句子 S_i , 画出其语法分析树。

解: (1) $L(G[S]) = \{\text{运算对象为 a, 运算符为双目运算的 + 或 * 的逆波兰表达式}\}$

(2) $S2$ 是文法的句子

(3) $S2$ 的语法树略

9. 设有文法 $G[A]: A \rightarrow Ba \quad B \rightarrow bB \mid a$ 求该文法所描述的语言 $L(G)$ 。

解: $L(G) = \{b^n a^2, n \geq 0\}$

10. 已知文法 $G[E]$:

$E \rightarrow T \mid E+T \mid E-T$

$T \rightarrow F \mid T*F \mid T/F$

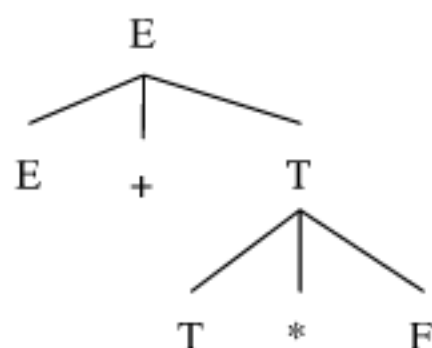
$F \rightarrow (E) \mid i$

(1) 给出该文法的 V_N 、 V_T 。

(2) 画出句型 $E+T*F$ 的语法分析树。

解: (1) $V_N = \{E, T, F\}$ $V_T = \{+, -, *, /, (,), i\}$

(2) 句型 $E+T*F$ 的语法树如下:



11. 有文法 $G: S \rightarrow aAcB \mid Bd$

$A \rightarrow AaB \mid c$

$B \rightarrow bScA \mid b$

写出句子 $acabcbddcc$ 的最左推导过程。

解: $S \Rightarrow aAcB \Rightarrow aAaBcB \Rightarrow acaBcB \Rightarrow acabcB \Rightarrow acabc bScA \Rightarrow acabc bBdcA \Rightarrow acabc b b d c c$

12. 文法 $G[T]$:

$T \rightarrow T*F \mid F$

$F \rightarrow F \uparrow P \mid P$

$P \rightarrow (T) \mid i$

证明 $T*P \uparrow (T*F)$ 是该文法的一个句型。

解: 方法一: 推导

证明:

$T \Rightarrow T*F \Rightarrow T*F \uparrow P \Rightarrow T*P \uparrow P \Rightarrow T*P \uparrow (T) \Rightarrow T*P \uparrow (T*F)$

由以上推导可知, $T*P \uparrow (T*F)$ 是该文法的一个句型。

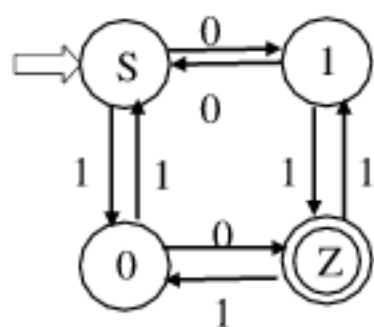
注意: 推导过程必须保证是最左推导或最右推导

方法二: 画语法分析树 (略)

13. 设有文法 $G[S]: S \rightarrow bTc \mid a \quad T \rightarrow R \quad R \rightarrow R/S \mid S$ 判断 $bR/bTc/bSc/ac$ 是否为该文法的句型。

解法同上题

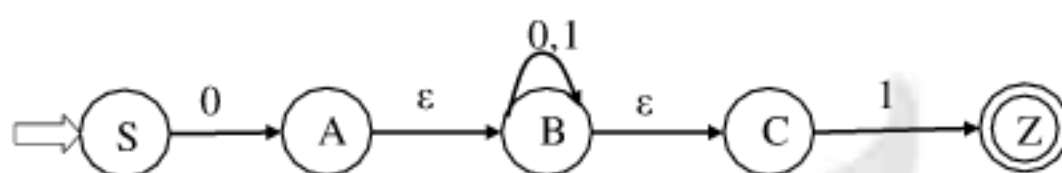
14. 给出下图所示的 DFA M 描述的语言。



解: $L(M) = \{ \text{字母表 } \Sigma = \{0,1\} \text{ 上的 } 0 \text{ 和 } 1 \text{ 的个数都是奇数的字符串} \}$

15. 设计一 DFA, 其输入字母表是 $\{0,1\}$, 接受以 0 开始以 1 结尾的所有序列。

解: 根据题意描述序列的相应的正规式为: $0(0|1)^*1$, 则该正规式对应的 NFA 状态转换图如下:



利用子集法将 NFA 确定化如下:

I	I_0	I_1
{S}	{A,B,C}	\varnothing
{A,B,C}	{B,C}	{B,C,Z}
{B,C}	{B,C}	{B,C,Z}
{B,C,Z}	{B,C}	{B,C,Z}

由 $\{B,C,Z\}$ 与 $\{Z\}$ 相交不为空可知 $\{B,C,Z\}$ 为终态。

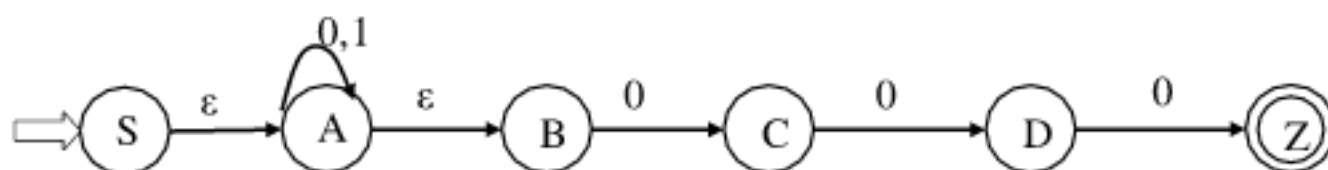
将状态集重命名, 得到相应的 DFA 状态表如下:

I	I_0	I_1
T_0	T_1	\varnothing
T_1	T_2	T_3
T_2	T_2	T_3
T_3^*	T_2	T_3

16. 构造一个 DFAM, 它接受以 000 结尾的二进制串。

解: 根据题意以 000 结尾的二进制串的正规式为: $(0|1)^*000$

相应的 NFA 状态转换图如下:



相应的 DFA 的初态为 $\varepsilon_{\text{closure}}(S) = \{S, A, B\}$

利用子集法构造相应的 DFA 如下:

I	I_0	I_1
{S,A,B}	{A,B,C}	{A,B}
{A,B,C}	{A,B,C,D}	{A,B}
{A,B}	{A,B,C}	{A,B}
{A,B,C,D}	{A,B,C,D,Z}	{A,B}
{A,B,C,D,Z}	{A,B,C,D,Z}	{A,B}

由{A,B,C,D,Z}与{Z}交集不为空可知{A,B,C,D,Z}为 DFA 的终态。

重命名如下：

I	I_0	I_1
0	1	2
1	3	2
2	1	2
3	4	2
4*	4	2

使用分割法将 DFA 最小化：

划分终态、非终态{0,1,2,3}、{4}：

检查子集{0,1,2,3}， $\{0,1,2\}_0 = \{1,3\} \subset \{0,1,2,3\}$ ， $\{3\}_0 = \{4\} \subset \{4\}$ 所以划分子集{0,1,2,3}为子集{0,1,2}、{3}：

检查子集{0,1,2}， $\{0,2\}_0 = \{1\} \subset \{0,1,2\}$ ， $\{1\}_0 = \{3\} \subset \{3\}$ ，所以划分子集{0,1,2}为子集{0,2}、{1}：

检查子集{0,2}， $\{0,2\}_0 = \{1\} \subset \{1\}$ ， $\{0,2\}_1 = \{2\} \subset \{2\}$ ，所以子集{0,2}不用再划分。

用 0 状态代替 2 状态，则最小化后的 DFA 如下：

	0	1
0	1	0
1	3	0
3	4	0
4*	4	0

17. 设 $M = (\{x,y\}, \{a,b\}, f, x, \{y\})$ 为一非确定的有限自动机，其中 f 定义如下：

$$\begin{aligned} f(x,a) &= \{x,y\} & f(x,b) &= \{y\} \\ f(y,a) &= \varnothing & f(y,b) &= \{x,y\} \end{aligned}$$

试构造相应的确定有限自动机 M' 。

解：确定有限自动机 M' 的初态为 $\varepsilon_closure(x) = \{x\}$

使用子集法确定化如下：

I	I_a	I_b
x	{x,y}	{y}
{x,y}	{x,y}	{x,y}
{y}	\varnothing	{x,y}

由{x,y},{y}与{y}交集不为空可知，{x,y},{y}为终态。

将 M' 的状态集重命名如下：

I	I_a	I_b
T_0	T_1	T_2

T_1^*	T_1	T_1
T_2^*	φ	T_1

18. 为下列正规式构造 NFA，并给出它们处理输入串 ababbab 的状态转换序列。

(1) $(alb)^*$

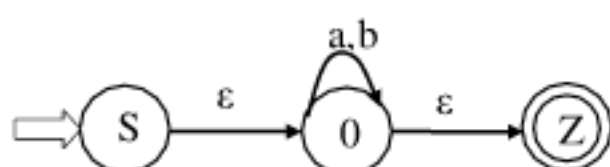
(2) $(a^* | b^*)^*$

(3) $((\epsilon la)b^*)^*$

(4) $(alb)^*abb(alb)^*$

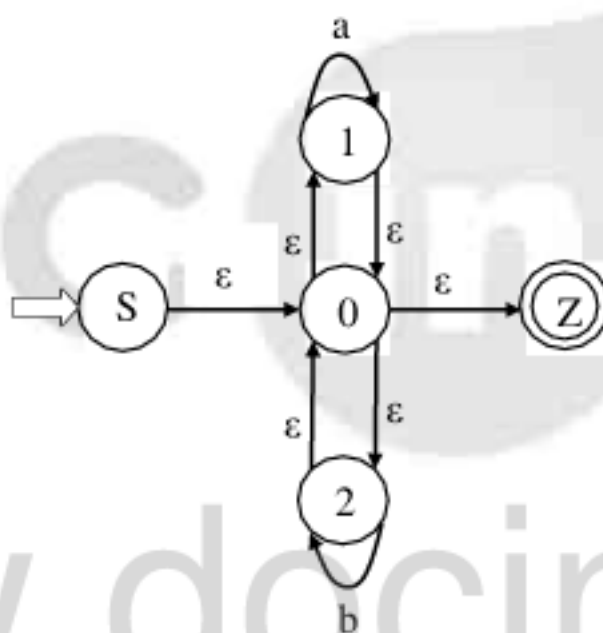
解：

(1) $(alb)^*$ 对应的 NFA 如下



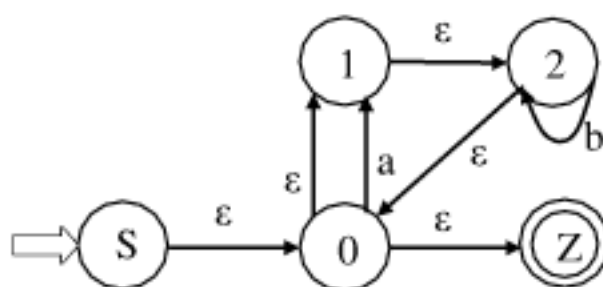
处理输入串 ababbab 的状态转换序列为：S00000000Z

(2) $(a^* | b^*)^*$ 对应的 NFA 如下



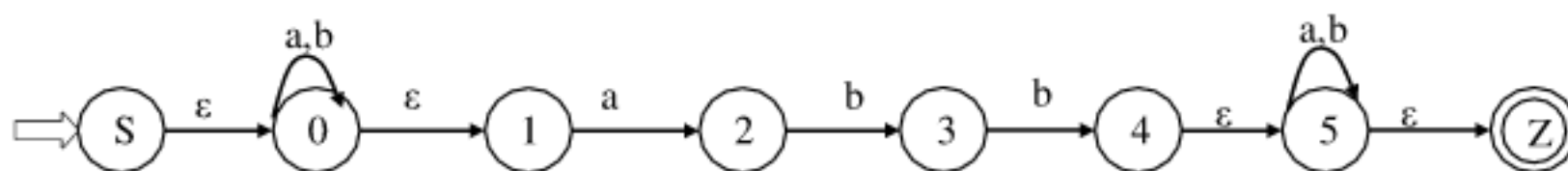
处理输入串 ababbab 的状态转换序列为：S01102201102220110220Z

(3) $((\epsilon la)b^*)^*$ 对应的 NFA 如下



处理输入串 ababbab 的状态转换序列为：S01220122201220Z

(4) $(alb)^*abb(alb)^*$ 对应的 NFA 如下



处理输入串 ababbab 的状态转换序列为: S0001234555Z

19. 给定 NFA M 如下表所示:

	ϵ	a	b	c
p	{q,r}	\varnothing	{q}	{r}
q	\varnothing	{p}	{r}	{p,q}
r*	\varnothing	\varnothing	\varnothing	\varnothing

(1) 计算每个状态的 ϵ 闭包。

(2) 将 M 确定化, 并画出其状态转换图。

解: (1) $\epsilon_closure(p) = \{p, q, r\}$

$\epsilon_closure(q) = \{q\}$

$\epsilon_closure(r) = \{r\}$

(2) 使用子集法对其确定化为对应的 DFA N 如下

DFA N 的初态集为 $\epsilon_closure(p) = \{p, q, r\}$

	I_a	I_b	I_c
{p,q,r}	{p, q, r}	{q,r}	{p,q,r}
{q,r}	{p, q, r}	{r}	{p,q,r}
{r}	\varnothing	\varnothing	\varnothing

重命名后 DFA 如下

	I_a	I_b	I_c
{p,q,r} 0*	{p, q, r} 0	{q,r} 1	{p,q,r} 0
{q,r} 1*	{p, q, r} 0	{r} 2	{p,q,r} 0
{r} 2*	\varnothing	\varnothing	\varnothing

其中由于 0、1、2 状态与 {r} 交集不为空, 所以均为终态。状态转换图略。

20. 设一 NFA $M = (\{p, q\}, \{a, b\}, f, p, \{q\})$, 其中 f 定义如下:

$f(p, a) = \{p, q\}$ $f(p, b) = \{q\}$ $f(q, a) = \varnothing$ $f(q, b) = \{p, q\}$

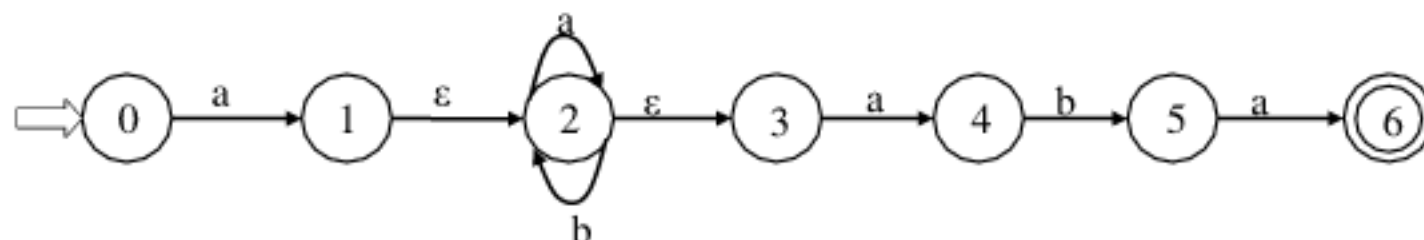
请构造相应的 DFA, 并给出状态转换图的表示形式。

解: 答案同题 18, 状态转换图略。

21. 已知 $NFA = (\{x, y, z\}, \{0, 1\}, M, \{x\}, \{z\})$, 其中: $M(x, 0) = \{z\}$, $M(y, 0) = \{x, y\}$,

$M(z, 0) = \{x, z\}$, $M(x, 1) = \{x\}$, $M(y, 1) = \varnothing$, $M(z, 1) = \{y\}$, 构造相应的 DFA。

22. 给定 NFA M 如下图所示:



(1) 写出 NFA M 的另外两种描述形式。

五元式形式:

$Q = \{0, 1, 3, 4, 5, 6\}$ $\Sigma = \{a, b\}$ $q_0 = \{0\}$ $Z = \{6\}$

$f = \{ f(0, a) = \{1\} \quad f(0, b) = \varnothing \quad f(0, \epsilon) = \varnothing$

$f(0,a)=\{1\}$	$f(0,b)=\varnothing$	$f(0,\varepsilon)=\varnothing$
$f(1,a)=\varnothing$	$f(1,b)=\varnothing$	$f(1,\varepsilon)=\{2\}$
$f(2,a)=\{2\}$	$f(2,b)=\{2\}$	$f(2,\varepsilon)=\{3\}$
$f(3,a)=\{4\}$	$f(3,b)=\varnothing$	$f(3,\varepsilon)=\varnothing$
$f(4,a)=\varnothing$	$f(4,b)=\{5\}$	$f(4,\varepsilon)=\varnothing$
$f(5,a)=\{6\}$	$f(5,b)=\varnothing$	$f(5,\varepsilon)=\varnothing$
$f(6,a)=\varnothing$	$f(6,b)=\varnothing$	$f(6,\varepsilon)=\varnothing$

}

状态表形式:

	a	b	ε
0	{1}	\varnothing	\varnothing
1	\varnothing	\varnothing	{2}
2	{2}	{2}	{3}
3	{4}	\varnothing	\varnothing
4	\varnothing	{5}	\varnothing
5	{6}	\varnothing	\varnothing
6*	\varnothing	\varnothing	\varnothing

(2)将 M 确定化且最小化为 DFA, 并给出其相应的状态转换图。

解: 使用子集法确定化如下

	I_a	I_b
{0}	{1,2,3}	
{1,2,3}	{2,3,4}	{2,3}
{2,3,4}	{2,3,4}	{2,3,5}
{2,3}	{2,3,4}	{2,3}
{2,3,5}	{2,3,4,6}	{2,3}
{2,3,4,6}	{2,3,4}	{2,3,5}

重命名如下:

	I_a	I_b
0	1	
1	2	3
2	2	4
3	2	3
4	5	3
5*	2	4

最小化的状态表如下

	a	b
0	1	
1	2	1
2	2	3
3	4	1
4*	2	3

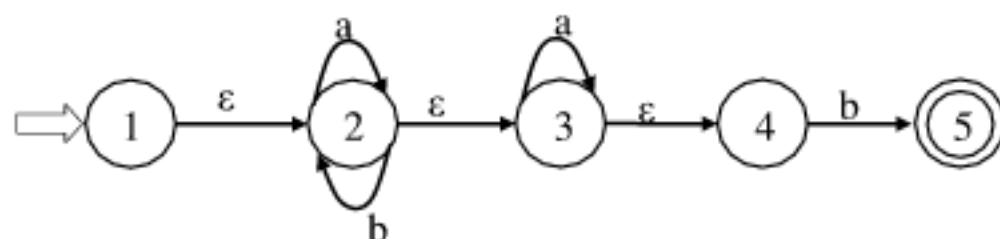
(3)用 DFA 识别字符串 aabaababaaaab 为哪几个单词。

识别单词: aaba, ababa; 不可识别 aaab

最长匹配识别单词 aabaababa; 不可识别串 aaab

23. 对下图和表所示的 NFA 分别确定化和最小化。

(1)NFA 如下图所示



(2)NFA 如下表所示

	a	b
0	{1,3}	{2,3}
1	{4}	\varnothing
2	\varnothing	{3,4}
3	{1,3}	{4}
4*	{4}	{4}

解:

(1)DFA 的初态集为 $\varepsilon_{\text{closure}}(1)=\{1,2,3,4\}$

使用子集法确定化如下

	I_a	I_b
{1,2,3,4}	{2,3,4}	{2,3,4,5}
{2,3,4}	{2,3,4}	{2,3,4,5}
{2,3,4,5}	{2,3,4}	{2,3,4,5}

其中 $\{2,3,4,5\} \cap \{5\}$ 不为空, 所以 $\{2,3,4,5\}$ 为终态。

重命名后如下所示

	I_a	I_b
{1,2,3,4} 0	{2,3,4} 1	{2,3,4,5} 2
{2,3,4} 1	{2,3,4} 1	{2,3,4,5} 2
{2,3,4,5} 2*	{2,3,4} 1	{2,3,4} 2

最小化的 DFA 如下:

	a	b
0	0	1
1*	0	1

(2) DFA 的初态集为 $\varepsilon_{\text{closure}}(0)=\{0\}$

使用子集法确定化如下

	I_a	I_b
{0}	{1,3}	{2,3}
{1,3}	{1,3,4}	{4}

{2,3}	{1,3}	{3,4}
{1,3,4}	{1,3,4}	{4}
{4}	{4}	{4}
{3,4}	{1,3,4}	{4}

其中{1,3,4}、{4}、 $\{3,4\} \cap \{4\}$ 不为空，所以均为终态。

重命名后如下所示

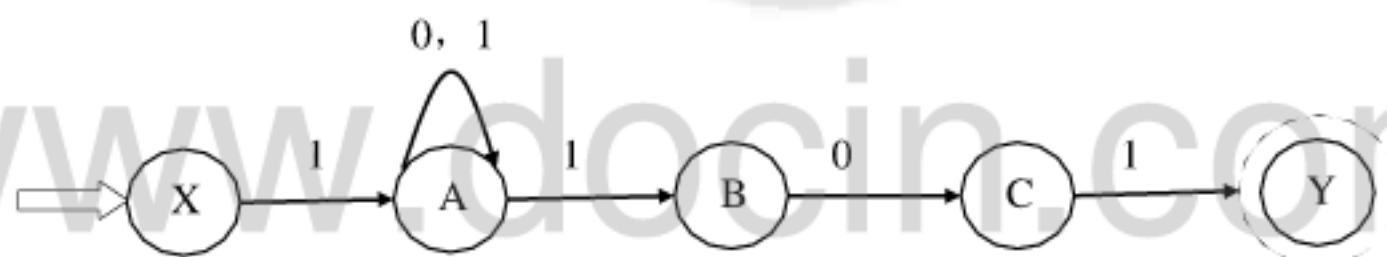
	I_a	I_b
0	1	2
1	3	4
2	1	5
3*	3	4
4*	4	4
5*	3	4

最小化的 DFA 如下：

	a	b
0	1	2
1	3	3
2	1	3
3*	3	3

24. 构造正规式 $1(0|1)^*101$ 相应的 DFA。

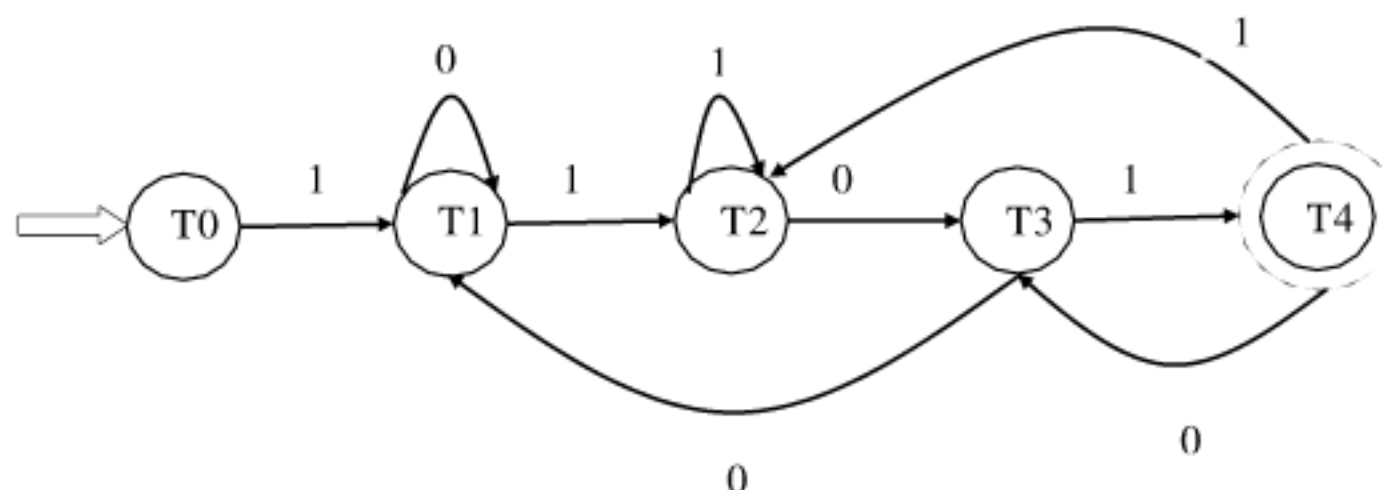
解：正规式对应的 NFA 如下图所示：



使用子集法将 NFA 确定化如下：

	0	1
{X} T0		{A}
{A} T1	{A}	{AB}
{AB} T2	{AC}	{AB}
{AC} T3	{A}	{ABY}
{ABY} T4	{AC}	{AB}

确定化后的 DFA 如下图所示：



25. 有如下 C 语言程序段:

```
int m;
```

```
m=33;
```

```
printf("m=%d\n",m);
```

指出程序段中合法的单词有哪些。

解: int 、 m、;、 m、=、 33、;、 printf、(、 “m=%d\n”、,、 m、)、;

26. 设计识别单词 “-”, “--”, “-=”, “->” 的最简的 DFA。

27. 已知文法 $G[E]$:

$$E \rightarrow T \mid E+T \mid E-T$$

$$T \rightarrow F \mid T * F \mid T / F$$

$$F \rightarrow (E) \mid i$$

消除文法中的左递归。

解: 消除左递归后文法如下:

$$E \rightarrow T E'$$

$$E' \rightarrow +T E' \mid -T E' \mid \varepsilon$$

$$T \rightarrow F T'$$

$$T' \rightarrow *F T' \mid /F T' \mid \varepsilon$$

$$F \rightarrow (E) \mid i$$

28. 文法 $G(A)$ 如下

$$A \rightarrow BaC \mid CbB$$

$$B \rightarrow Ac \mid c$$

$$C \rightarrow Bb \mid b$$

请消除其左递归。

解: 令文法 $G(A)$ 的非终结符排序为 A, B, C

对于 A , 不存在直接左递归, 将 A 带入 B , 代换后的 B 的规则为:

$$B \rightarrow BaCc \mid CbBc \mid c$$

B 存在直接左递归, 消除 B 的直接左递归

$$B \rightarrow CbBcB' \mid cB'$$

$$B' \rightarrow aCcB' \mid \varepsilon$$

将 B 代入 C , 代换后 C 的规则变为:

$$C \rightarrow CbBcB'b \mid cB'b \mid b$$

C 存在直接左递归, 消除 C 的直接左递归

$$C \rightarrow cB'bC' \mid bC'$$

$$C' \rightarrow bBcB'bC' \mid \varepsilon$$

所以消除 G(A) 的左递归后的文法为:

$$A \rightarrow BaC \mid CbB$$

$$B \rightarrow CbBcB' \mid cB'$$

$$B' \rightarrow aCcB' \mid \varepsilon$$

$$C \rightarrow cB'bC' \mid bC'$$

$$C' \rightarrow bBcB'bC' \mid \varepsilon$$

29. 将下面的左递归文法 G(S) 改为非左递归的。

$$S \rightarrow SaP \mid Sf \mid P$$

$$P \rightarrow QbP \mid Q$$

$$Q \rightarrow cSd \mid e$$

30. 将文法 G[V] 改造成为 LL(1) 的。

$$G[V]: V \rightarrow NIN[E]$$

$$E \rightarrow VIV+E$$

$$N \rightarrow i$$

解: 改造文法如下

$$V \rightarrow NA$$

$$A \rightarrow \varepsilon \mid [E]$$

$$E \rightarrow VB$$

$$B \rightarrow \varepsilon \mid +E$$

$$N \rightarrow i$$

验证改造后的文法是否为 LL(1):

$$\text{SELECT}(A \rightarrow \varepsilon) = \{ \varepsilon \}$$

$$\text{SELECT}(A \rightarrow [E]) = \{ [\}$$

$$\text{SELECT}(B \rightarrow +E) = \{ + \}$$

$$\text{SELECT}(B \rightarrow \varepsilon) = \{ \varepsilon \}$$

由于文法中相同非终结符的不同产生式的选择集合交集为空, 所以改造后的文法是 LL(1) 文法。

31. 文法 G[A] 如下:

$$G(A): A \rightarrow aAB \mid a \quad B \rightarrow Bb \mid d$$

(1) 判断该文法是否为 LL(1) 文法。

(2) 若不是请改写为等价的 LL(1) 文法 G'[A]。

解: (1) 文法不是 LL(1) 文法。因为文法中含有左递归, 含有左公共因子。

(2) 改写后的文法为:

$$A \rightarrow aA'$$

$$A' \rightarrow AB \mid \varepsilon$$

$$B \rightarrow dB'$$

$$B' \rightarrow bB' \mid \varepsilon$$

验证: 对 A' 有 $\text{FIRST}(AB) = \{a\} \cap \text{FIRST}(A') = \{\#\} = \emptyset$

对 B' 有 $FIRST(b B') = \{b\} \cap FIRST(B') = \{\#\} = \emptyset$
 所以改写后的文法是 LL(1) 文法。

32. 设有文法 $G[S]: S \rightarrow Aa|b \quad A \rightarrow Ac|Sd|e$

- (1). 该文法是否含有直接左递归。
- (2). 该文法是否含有间接左递归。
- (3). 若有请给出消除递归后的文法。

解:

(1) 含有直接左递归 $A \rightarrow Ac$

(2) 含有间接左递归 $S \Rightarrow Aa \Rightarrow Sda$

(3) 消除左递归如下:

将非终结符排序 S, A

$S \rightarrow Aa|b$ 保持不变

将 S 代入 $A \rightarrow Ac|Sd|e$ 得:

$A \rightarrow Ac|Aad|bd|e$

消除上式左递归得: $A \rightarrow cB|adB \quad B \rightarrow bdB|eB|\epsilon$

所以消除左递归后的文法如下:

$S \rightarrow Aa|b$

$A \rightarrow cB|adB$

$B \rightarrow bdB|eB|\epsilon$

33. 已知文法 $G[S]$ 如下:

$S \rightarrow AB|bC$

$A \rightarrow b|\epsilon$

$B \rightarrow aD|\epsilon$

$C \rightarrow AD|b$

$D \rightarrow aS|\epsilon$

求 $FIRST(aD)$, $FOLLOW(A)$ 和 $SELECT(B \rightarrow \epsilon)$ 。

解: $FIRST(aD) = \{a\}$

$$\begin{aligned} FOLLOW(A) &= (FIRST(B) - \{\epsilon\}) \cup FOLLOW(S) \cup (FIRST(D) - \{\epsilon\}) \cup FOLLOW(C) \\ &= \{a\} \cup \{\#\} \cup FOLLOW(D) \cup \{a\} \cup FOLLOW(S) \\ &= \{a, \#\} \cup FOLLOW(B) \\ &= \{a, \#\} \cup FOLLOW(S) \\ &= \{a, \#\} \end{aligned}$$

$$SELECT(B \rightarrow \epsilon) = (FIRST(\epsilon) - \{\epsilon\}) \cup FOLLOW(B) = FOLLOW(S) = \{\#\}$$

34. 文法 $G(E)$ 如下, 求 $FIRST(T')$ 和 $FOLLOW(F)$ 。

$E \rightarrow TE'$

$E' \rightarrow +TE'|\epsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT'|\epsilon$

$F \rightarrow (E)|a$

解: $FIRST(T') = \{*, \epsilon\}$

$\text{FOLLOW}(F) = (\text{FIRST}(T') - \{\epsilon\}) \cup \text{FOLLOW}(T) \cup \text{FOLLOW}(T')$

$\text{FIRST}(T') - \{\epsilon\} = \{*\}$

$\text{FOLLOW}(T) = (\text{FIRST}(E') - \{\epsilon\}) \cup \text{FOLLOW}(E) \cup \text{FOLLOW}(E') = \{+,), \#\}$

所以 $\text{FOLLOW}(F) = \{*, +,), \#\}$

35. 判断文法 $G(A): A \rightarrow aAa \mid \epsilon$ 是否为 LL(1) 文法。

解: $\text{SELECT}(A \rightarrow aAa) = \{a\}$

$\text{SELECT}(A \rightarrow \epsilon) = \text{FOLLOW}(A) = \{a, \#\}$

由于 $\text{SELECT}(A \rightarrow aAa) \cap \text{SELECT}(A \rightarrow \epsilon) = \{a\}$

所以该文法不是 LL(1) 文法。

36. 判断文法 $G[S]$ 是否为 LL(1) 文法。

$S \rightarrow AB \mid PQx$

$A \rightarrow xy$

$B \rightarrow bc$

$P \rightarrow dP \mid \epsilon$

$Q \rightarrow aQ \mid \epsilon$

解: $\text{SELECT}(S \rightarrow AB) = \text{FIRST}(A) = \{x\}$

$\text{SELECT}(S \rightarrow PQx) = (\text{FIRST}(P) - \{\epsilon\}) \cup (\text{FIRST}(Q) - \{\epsilon\}) \cup \text{FIRST}(x) = \{d, a, x\}$

由于 $\text{SELECT}(S \rightarrow AB) \cap \text{SELECT}(S \rightarrow PQx) = \{x\}$

所以该文法不是 LL(1) 文法。

37. 判断文法 $G(A): A \rightarrow aABe \mid Ba \quad B \rightarrow dB \mid \epsilon$ 是否为 LL(1) 文法。

解: $\text{SELECT}(A \rightarrow aABe) = \{a\}$

$\text{SELECT}(A \rightarrow Ba) = (\text{FIRST}(B) - \{\epsilon\}) \cup \{a\} = \{a, d\}$

由于 $\text{SELECT}(A \rightarrow aABe) \cap \text{SELECT}(A \rightarrow Ba) = \{a\}$

所以该文法不是 LL(1) 文法。

38. 设有文法 $G[A]: A \rightarrow AaB \mid bB \quad B \rightarrow Dc \quad D \rightarrow Ad$

(1) 计算文法中每个非终结符的 FIRST 和 FOLLOW 集合。

(2) 判断该文法是否为 LL(1) 文法? 若不是请改写为 LL(1) 文法。

(3) 构造 LL(1) 分析表。

解:

(1) $\text{FIRST}(A) = \{b\} \quad \text{FIRST}(B) = \{b\} \quad \text{FIRST}(D) = \{b\}$

$\text{FOLLOW}(A) = \{a, d, \#\} \quad \text{FOLLOW}(B) = \{a, d, \#\} \quad \text{FOLLOW}(D) = \{c\}$

(2) 由于文法存在左递归 $A \rightarrow AaB$, 所以该文法不是 LL(1) 文法。

改写文法:

将 D 代入 B 得 $B \rightarrow Adc$

将 B 代入 A 得 $A \rightarrow AaAdc \mid bAdc$

消除左递归得: $A \rightarrow bAdcA'$

$A' \rightarrow aAdcA' \mid \epsilon$

由于 $\text{SELECT}(A' \rightarrow aAdcA') \cap \text{SELECT}(A' \rightarrow \epsilon) = \{a\} \cap \{d, \#\} = \emptyset$

所以改写后的文法是 LL(1) 文法。

(3) 改写后的文法的 LL(1)分析表如下

	a	b	c	d	#
A		$A \rightarrow b A d c A'$			
A'	$A' \rightarrow a A d c A'$			$A' \rightarrow \epsilon$	$A' \rightarrow \epsilon$

39. 设有下列文法:

$PROGRAM \rightarrow begin\ d;\ S\ end$

$S \rightarrow d;\ S \mid s\ T$

$T \rightarrow ;\ s\ T \mid \epsilon$

(1) 构造该文法 LL(1)分析表:

(2) 给出句子 $begin\ d;\ s;\ s\ end$ 的分析过程 (写出含有符号栈、输入串和规则的分析过程)。

解: (1) $SELECT(PROGRAM \rightarrow begin\ d;\ S\ end) = FISRT(begin\ d;\ S\ end) = \{begin\}$

$SELECT(S \rightarrow d;\ S) = FISRT(d;\ S) = \{d\}$

$SELECT(S \rightarrow s\ T) = FISRT(s\ T) = \{s\}$

$SELECT(T \rightarrow ;\ s\ T) = FISRT(;s\ T) = \{;\}$

$SELECT(T \rightarrow \epsilon) = FOLLOW(T) = \{end\}$

根据上述结果构造文法 LL(1)分析表如下:

	begin	d	s	;	end
PROGRAM	$PROGRAM \rightarrow begin\ d;\ S\ end$				
S		$S \rightarrow d;\ S$	$S \rightarrow s\ T$		
T				$T \rightarrow ;\ s\ T$	$T \rightarrow \epsilon$

(3) 句子 $begin\ d;\ s;\ s\ end$ 的分析过程如下:

步骤	分析栈	剩余输入串	推导用产生式或匹配
1	# PROGRAM	begin d;s;s end#	$PROGRAM \rightarrow begin\ d;\ S\ end$
2	# end S; d begin	begin d;s;s end#	'begin' 匹配
3	# end S; d	d;s;s end#	'd' 匹配
4	# end S;	;s;s end#	';' 匹配
5	# end S	s;s end#	$S \rightarrow s\ T$
6	# end T s	s;s end#	's' 匹配
7	# end T	;s end #	$T \rightarrow ;\ s\ T$
8	# end T s;	;s end #	';' 匹配
9	# end T s	s end #	's' 匹配
10	# end T	end #	$T \rightarrow \epsilon$
11	# end	end #	'end' 匹配
12	#	#	分析成功

40. 设有文法
- $G[S]: S \rightarrow Pab \mid bP \quad P \rightarrow \varepsilon \mid b$

根据文法 $G[S]$ 填写如下 LL(1) 分析表的内容。

	a	b	#
P	$\underline{P \rightarrow \varepsilon}$	$\underline{P \rightarrow b}$	$\underline{P \rightarrow \varepsilon}$

41. 设有文法
- $G[S]$
- :

 $S \rightarrow AB \mid bb \mid bAC \quad A \rightarrow \varepsilon \mid b \quad B \rightarrow \mid aC \quad C \rightarrow aSc$ 则 $FOLLOW(A) = \{ \underline{a, c, \#} \}$ 。对给出的文法 $G[S]$ 填写如下 LL(1) 分析表的内容。

	a	b	c	#
A	$\underline{A \rightarrow \varepsilon}$	$\underline{A \rightarrow b}$	$\underline{A \rightarrow \varepsilon}$	$\underline{A \rightarrow \varepsilon}$

42. 已知文法
- $G[S]$
- :

 $S \rightarrow aBc \mid bAB$ $A \rightarrow aAb \mid b$ $B \rightarrow b \mid \varepsilon$

- (1) 构造其 LL(1) 分析表;

- (2) 判断符号串
- baabbb**
- 是否为该文法的句子 (写出含有符号栈、输入串和规则的分析过程)。

解:

(1) $SELECT(S \rightarrow aBc) = \{a\}$ $SELECT(S \rightarrow bAB) = \{b\}$ $SELECT(A \rightarrow aAb) = \{a\}$ $SELECT(A \rightarrow b) = \{b\}$ $SELECT(B \rightarrow b) = \{b\}$ $SELECT(B \rightarrow \varepsilon) = \{c, \#\}$

由于文法中相同非终结符的不同产生式的选择集合交集为空, 所以该文法是 LL(1) 文法。根据每条产生式的选择集合构造 LL(1) 分析表如下:

	a	b	c	#
S	$S \rightarrow aBc$	$S \rightarrow bAB$		
A	$A \rightarrow aAb$	$A \rightarrow b$		
B		$B \rightarrow b$	$B \rightarrow \varepsilon$	$B \rightarrow \varepsilon$

- (2) 符号串
- baabbb**
- 的分析过程如下:

步骤	分析栈	剩余输入串	推导用产生式或匹配
1	#S	baabbb#	$S \rightarrow bAB$
2	#BAb	baabbb#	'b' 匹配
3	#BA	aabbb#	$A \rightarrow aAb$
4	#BbAa	aabbb#	'a' 匹配

5	#BbA	abbb #	$A \rightarrow aAb$
6	# BbbAa	abbb #	'a'匹配
7	# BbbA	bbb #	$A \rightarrow b$
8	# Bbbb	bbb #	'b'匹配
9	# Bbb	bb #	'b'匹配
10	# Bb	b#	'b'匹配
11	#B	#	$B \rightarrow \epsilon$
12	#	#	分析成功

43. 构造下面文法的 LL(1) 分析表。

$D \rightarrow TL$

$T \rightarrow \text{int} \mid \text{real}$

$L \rightarrow \text{id} R$

$R \rightarrow , \text{id} R \mid \epsilon$

解: $\text{SELECT}(D \rightarrow TL) = \{\text{int}, \text{real}\}$

$\text{SELECT}(T \rightarrow \text{int}) = \{\text{int}\}$

$\text{SELECT}(T \rightarrow \text{real}) = \{\text{real}\}$

$\text{SELECT}(L \rightarrow \text{id} R) = \{\text{id}\}$

$\text{SELECT}(R \rightarrow , \text{id} R) = \{,\}$

$\text{SELECT}(R \rightarrow \epsilon) = \{\#\}$

由于文法中相同非终结符的不同产生式的选择集合交集为空, 所以该文法是 LL(1) 文法。根据每条产生式的选择集合构造 LL(1) 分析表如下:

	int	real	id	,	#
D	$D \rightarrow TL$	$D \rightarrow TL$			
T	$T \rightarrow \text{int}$	$T \rightarrow \text{real}$			
L			$L \rightarrow \text{id} R$		
R				$R \rightarrow , \text{id} R$	$R \rightarrow \epsilon$

44. 文法 $G[A]$ 如下:

$G(A): A \rightarrow aABe \mid a \quad B \rightarrow Bb \mid d$

(1). 该文法是否为 LL(1) 文法。

(2). 若不是请改写为等价的 LL(1) 文法 $G'[A]$ 。

(3). 构造 $G'[A]$ 的 LL(1) 分析表。

(4). 给出符号串 $aade\#$ 的分析过程。

解:

(1) 由于关于 A 的产生是含有左公共因子, 关于 B 的产生是含有左递归, 所以该文法不

是 LL(1) 文法。

(2) 改写后的文法为：

$$A \rightarrow a A'$$

$$A' \rightarrow ABe \mid \varepsilon$$

$$B \rightarrow dB'$$

$$B' \rightarrow bB' \mid \varepsilon$$

验证：对 A' 有 $FIRST(ABe) = \{a\} \cap FOLLOW(A') = \{d, \#\} = \emptyset$

对 B' 有 $FIRST(bB') = \{b\} \cap FIRST(B') = \{e\} = \emptyset$

所以改写后的文法是 LL(1) 文法。

(3) 该文法的 LL(1) 分析表。

	a	d	b	e	#
A	$A \rightarrow a A'$				
A'	$A' \rightarrow ABe$	$A' \rightarrow \varepsilon$			$A' \rightarrow \varepsilon$
B		$B \rightarrow dB'$			
B'			$B' \rightarrow bB'$	$B' \rightarrow \varepsilon$	

(4) 符号串 aade# 的分析过程如下

步骤	分析栈	剩余输入串	推导用产生式或匹配
1	#A	aade#	$A \rightarrow a A'$
2	#A'a	aade#	'a' 匹配
3	#A'	ade#	$A' \rightarrow ABe$
4	#eBA	ade#	$A \rightarrow a A'$
5	#eBA'a	ade#	'a' 匹配
6	#eBA'	de#	$A' \rightarrow \varepsilon$
7	#eB	de#	$B \rightarrow dB'$
8	#eB'd	de#	'd' 匹配
9	#eB'	e#	$B' \rightarrow \varepsilon$
10	#e	e#	'e' 匹配
11	#	#	分析成功

45. 已知文法 $G[S]$:

$$S \rightarrow aH$$

$$H \rightarrow aMd \mid d$$

$$M \rightarrow Ab \mid \varepsilon$$

$$A \rightarrow aM \mid e$$

(1) 证明该文法是否为 LL(1) 文法。

证明: $\text{SELECT}(H \rightarrow aMd) \cap \text{SELECT}(H \rightarrow d) = \{a\} \cap \{d\} = \emptyset$

$\text{SELECT}(M \rightarrow Ab) \cap \text{SELECT}(M \rightarrow \varepsilon) = \{a, e\} \cap \{d, b\} = \emptyset$

$\text{SELECT}(A \rightarrow aM) \cap \text{SELECT}(A \rightarrow e) = \{a\} \cap \{e\} = \emptyset$

\therefore 该文法是 LL(1) 文法。

(2) 构造该文法的 LL(1) 分析表。

	a	d	b	e	#
S	$S \rightarrow aH$				
H	$H \rightarrow aMd$	$H \rightarrow d$			
M	$M \rightarrow Ab$	$M \rightarrow \varepsilon$	$M \rightarrow \varepsilon$	$M \rightarrow Ab$	
A	$A \rightarrow aM$			$A \rightarrow e$	

(3) 给出输入串 aaabd# 的预测分析过程, 并判断该输入串是否为该文法的句子。

步骤	分析栈	剩余输入串	推导用产生式或匹配
1	#S	aaabd#	$S \rightarrow aH$
2	#Ha	aaabd#	'a' 匹配
3	#H	aabd#	$H \rightarrow aMd$
4	#dMa	aabd#	'a' 匹配
5	#dM	abd#	$M \rightarrow Ab$
6	#dbA	abd#	$A \rightarrow aM$
7	#dbMa	abd#	'a' 匹配
8	#dbM	bd#	$M \rightarrow \varepsilon$
9	#db	bd#	'b' 匹配
10	#d	d#	'd' 匹配
11	#	#	分析成功

\therefore 输入串 aaabd# 是该文法的句子。

46. 编写下列文法 G[E] 的递归下降分析程序(伪代码即可)。

G[E]:

$E \rightarrow Aa \mid Bb$

$A \rightarrow cA \mid eB$

$B \rightarrow bd$