**编译原理复习题**

**一、填空题：**

1、编译方式与解释方式的根本区别在于（ 是否生成目标代码 ）。

2、对编译程序而言，输入数据是（ 源程序 ），输出结果是（ 目标程序 ）。

3、如果编译程序生成的目标程序是机器代码程序，则源程序的执行分为两大阶段：（编译阶段 ）和（ 运行阶段 ）。

4、如果编译程序生成的目标程序是汇编语言程序，则源程序的执行分成三个阶段：（ 编译阶段）、（汇编阶段）和（运行阶段）。

5、自顶向下语法分析方法会遇到的主要问题有（ 回溯 ）和（ （左递归带来的）无限循环 ）。

6、LL(k)分析法中，第一个L的含义是（ 从左到右进行分析 ），第二个L的含义是（ 每次进行最左推导 ），“k”的含义是（向输入串中查看K个输入符号 ）。

7、LL(1)分析法中，第一个L的含义是（从左到右进行分析 ），第二个L的含义是（每次进行最左推导 ），“1”的含义是（向输入串中查看1个输入符号 ）。

8、自顶向下语法分析方法的基本思想是：从（识别符号）出发，不断建立（ 直接推导 ），试图构造一个推导序列，最终由它推导出与输入符号相同的（ 符号串 ）。

9、自底向上语法分析方法的基本思想是：从待输入的符号串开始，利用文法的规则步步向上进行（直接归约），试图（归约）到文法的（ 识别符号|开始符号 ）。

10、LR(0)分析法的名字中，“L”的含义是（从左到右进行分析），“R”的含义是（ 采用最右推导的逆过程---最左归约 ），“0”的含义是（向貌似句柄的符号串后查看0个输入符号）。

11、LR(1)分析法的名字中，“L”的含义是（从左到右进行分析），“R”的含义是（采用最右推导的逆过程---最左归约），“1”的含义是（向貌似句柄的符号串后查看1个输入符号）。

12、SLR(1)分析法的名字中，“S”的含义是（简单的 ），“L”的含义是（从左到右进行分析），“R”的含义是（采用最右推导的逆过程---最左归约），“1”的含义是（向貌似句柄的符号串后查看1个输入符号）。

13、在编译过程中，常见的中间语言形式有（逆波兰表示 ）、（三元式 ）、（四元式）和（树形表示）。

14、在编译程序中安排中间代码生成的目的是（便于代码优化）和（便于目标程序的移植 ）。

15、表达式-a+b\*(-c+d)的逆波兰表示为（ **a-bc-d+\*+**  ）。

16、表达式a+b\*(c+d/e)的逆波兰表示为（**abcde/+\*+** ）。

17、表达式a:=a+b\*c↑(d/e)/f的逆波兰表示为（ **aabcde/↑\*f/+:=** ）。

18、文法符号的属性有（ 继承属性 ）和（ 综合属性 ）两种。

19、一个文法符号的继承属性是通过语法树中它的（兄弟结点与父 ）结点的相应文法符号的属性来计算的。

20、一个文法符号的综合属性是通过语法树中它的（子 ）结点的属性来计算的。

21、语法制导的编译程序能同时进行（ 语法 ）分析和（ 语义 ）分析。

22、编译过程中扫描器所完成的任务是从（ 源程序 ）中识别出一个个具有（ 独立语法意义的单词 ）。

23、确定的有穷自动机是一个（ 五元组 ），通常表示为（ **DFA=（K，∑，M，S，Z）**）。

24、已知文法G(E):

E->T|E+T|E-T

T->F|T\*F|T/F

F->(E)|i

该文法的开始符号是（ E ），终结符号集合VT是（ **{ +,-,\*./,(,),i }** ），非终结符号集合VN是（ **{ E,T,F }** ），句型T+T\*F+i的短语有（ **T+T\*F+I ，T\*F第一个T，i**）。

25、已知文法G(E):

E->T|E+T|E-T

T->F|T\*F|T/F

F->(E)|i

改写该文法以消除直接左递归，改写后的文法为：E->（**+TE‘|-TE‘|ε** ），T->（ **FT‘** ），（**T‘-> \*FT‘|ε**），F->（ **（E）| i** ）。

26、已知文法G(Z):

Z->U0|V1

U->Z1|1

V->Z0|0

请写出全部由此文法描述的只含四个符号的句子：（ **0101，1010，1001，0110** ）。 该文法是Chomsky（  **3**  ）型文法。

27、Chomsky定义的四种形式语言文法分别为：（ 0型）文法--又称（短语文法 ）文法，（1型）文法--又称（上下文有关）文法，（2型 ）文法--又称（上下文无关 ）文法，（3型）文法--又称（正则）文法。

28、文法G(S):

S->AB

A->aAb|ab

B->Bc|ε

其描述的语言L(G[S])=（ **anbncm,n≥1，m≥0** ）。

29、文法G(S):

S->SAS|b|c

A->aaA|a

其描述的语言L(G[S])=（b,C,或者是以b开头、以c结尾的、中间是任意个由b或c间隔开的奇数个a组成的符号串）。

30、过程与过程引用中信息交换的方法是（全局变量的使用）和（参数传递 ）。

31、形式参数和实在参数之间的对应关系通常按（它们在源程序中的位置）来确定。

32、按传结果的方式进行形实参数结合，是一种把（传地址 ）和（传值）的特点相结合的参数传递方式。

33、在PASCAL中，由于允许用户动态申请与释放内存空间，所以必须采用（ 堆 ）存储分配方式。

34、编译程序进行数据流分析的目的是（ 为了进行全局优化 ）。

35、根据所涉及程序的范围，优化可分为（ 局部优化 ）、（ 循环优化 ）和（ 全局优化 ）三种。

36、局部优化是局限于一个（ 基本块 ）范围内的一种优化。

37、词法分析阶段的错误主要是（ 拼写错误 ），可通过（ 最小距离匹配 ）的办法去纠正错误。

38、对错误的处理方法一般有（ 校正法 ）和（ 局部优化法 ）。

39、源程序中的错误一般有（ 词法错误 ）、（ 语法错误 ）、（ 语义错误 ）和（ 违反环境限制的错误 ）四种。

40、翻译程序是这样一种程序，它能够将（ 用甲语言书写的程序 ）转换成与其等价的（ 用乙语言书写的程序 ）。

41、编译程序的工作过程一般可以划分成（ 词法分析、语法分析、语义分析、中间代码生成、代码优化 ）等五个基本阶段。

42、编译程序的工作过程还会伴有（ 表格处理 ）和（ 出错处理 ）。

**二、选择题：**

1、在使用高级语言编程时，首先可通过编译程序发现源程序的全部（ A ）错误和部分（ B ）错误。

A、语法 B、语义 C、语用 D、运行

2、程序语言的处理程序是一种（ A ）。

A、系统软件 B、应用软件 C、实时系统 D、分布式系统

3、（ B ）是两类程序语言处理程序，它们的主要区别在于是否生成目标代码。

A、高级语言和低级语言 B、解释程序和编译程序 C、编译程序和操作系统 D、系统程序和应用程序

4、汇编语言是将（ A ）翻译成（ B ）；编译程序是将（ C ）翻译成（ D ）。

A、汇编语言程序 B、机器语言程序 C、高级语言程序 D、汇编或机器语言程序 e、汇编或高级语言程序 F、机器或高级语言程序

5、编译程序与具体的机器（ A ），与具体的语言（ B ）。

A、有关 B、无关

6、编译程序生成的目标程序（ B ）是可执行的程序。

A、一定 B、不一定

7、编译程序生成的目标程序（ B ）是机器语言程序。

A、一定 B、不一定

8、把汇编语言程序翻译成机器可执行的目标程序的工作是由（ B ）完成的。

A、编译器 B、汇编器 C、解释器 D、预处理器

9、编译过程中，语法分析器的任务是（ B ）。

(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、(2)(3) B、(2)(3)(4) C、(1)(2)(3) D、(1)(2)(3)(4)

10、文法G所描述的语言是（ D ）的集合

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、由文法的识别符号推出的所有符号串； D、由文法的识别符号推出的所有终结符符号串；

11、描述语言L={ambn|n>m>1}的文法为（　D ）。

A、Z->Abb, A->aA|a, B->bB|b B、Z->ABb, A->Aa|a,B->aBb|b

C、Z->Ab, A->aAb|a D、Z->aAb, A->Ab|aAb|@

12、一个句型中的最左（　B 　）称为该句型的句柄。

　 A、短语 B、简单短语 C、素短语 D、终结符号

13、文法的二义性和语言的二义性是两个（　A　　）的概念。

　 A、不同 B、相同 C、无法判断

14、上下文无关文法（ A　）产生语言L={ambnci|i>1,n>1}

A、可以　　 B、不可以

15、（　　B　　　）正规文法能产生下面的语言：L={ambn|n>1}

　 A、存在一个　　 B、不存在任何　　　c、无法判断

16、编译程序中的语法分析器接受以（　　C　　　）为单位的输入，并产生有关信息供以后各阶段使用。

A、表达式　　 B、产生式　　　C、单词　　　D、语句

17、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、自左至右 B、自顶向下 C、自底向上 D、自右至左

18、算符优先分析法每次都是对（ E ）进行归约，简单优先分析法每次都是对（ C ）进行归约。

A、最左短语 B、简单短语 C、句柄 D、素短语 E、最左素短语

19、文法G(S)：S—>aTb|,，T->R，R—>R/S|S的句型aR/aSb/aTb,b的最左素短语为（ B ）。

A、aTb B、aSb C、S D、R/ E、,

20、LR语法分析栈中存放的状态是识别（ B ）的DFA状态。

A、前缀 B、可归前缀 C、项目 D、句柄

21、已知文法G[S]：s->LaR|R，L->bR|c，R->L 该文法是（ B ）。

(1)LR(0)文法；(2)SLR(1)文法；(3)LR(1)文法；(4)LALR(1)文法；(5)都不是

A、(1)(2) B、(3)(4) C、(1)(2)(3)(4) D、(5) E、(6)

22、若一个句型中出现了某一产生式的右部，则此右部（ B ）是该句型的句柄。

A、一定 B、不一定

23、LR(K)方法是（ D ）。

A、从左到右分析，每次走K步的一种编译方法 B、从左到右分析，共经过K步的一种编译方法 C、从左到右分析，每次向前预测K步的一种编译方法 D、从左到右分析，每次向貌似句柄的符号串后看K个输入符号的一种编译方法

24、文法G[S]：S->AA，A->Aa|a 不是LR(1)文法的理由是（ D ）

A、FIRST(S)NFIRST(A)≠ф B、FIRST(A)NFOLLLOW(A) ≠ф C、FIRST(Aa)NFIRST(a) ≠ф D、都不是

25、下列关于标识符和名字的叙述中，正确的是（ C ）。

A、标识符有一定的含义 B、名字是一个没有意义的字符序列 C、名字有确切的属性 D、都不对

26、编译程序在其工作过程中使用最多的数据结构是（ C ）。它记录着源程序中的各种信息，以便查询或修改。

A、线性表 B、链表 C、表 D、符号表

27、在编译程序在其工作过程中使用的各种表中，以（ D ）最重要，其生存期最长，使用也最频繁。

A、线性表 B、链表 C、表 D、符号表

28、编译程序使用（ B ）区别标识符的作用域。

A、说明标识符的过程或函数名 B、说明标识符的过程或函数的静态层次 C、说明标识符的过程或函数的动态层次 D、标识符的行号

29、动态存储分配时，可以采用的分配方法有（ C ）。

(1)以过程为单位的栈式动态存储分配；(2)堆存储分配；(3)最佳分配方法

A、(1) B、(2) C、(1)(2) D、(1)(2)(3)

30、过程调用时，参数的传递方法通常有（ D ）。

(1)传值；(2)传地址；(3)传结果；(4)传名

A、(1)(2) B、(1)(2)(3) C、(1)(2)(4) D、(1)(2)(3)(4)

31、在编译方法中，动态存储分配的含义是（ A ）。

A、在运行阶段对源程序中的量进行分配 B、在编译阶段对源程序中的量进行分配

C、在编译阶段对源程序中的量进行分配，在运行时这些量的地址可以根据需要改变

D、以上都不对

32、PASCAL中过程说明的局部量地址分配在（ B ）。

A、调用者的数据区中 B、被调用者的数据区 C、主程序的数据区 D、公共数据区

33、表达式-a+b\*c+d+(e\*f)/d\*e，如果优先级由高到低依次为-、+、\*、/，且均为左结合，则其后缀式为（ D ）。

A、abc\*+d+ef\*d/e\*+- B、a-bc\*def\*d/e\*+++ C、a-bc\*+def\*d/e\*++ D、a-b+cd+ef\*+\*de\*/

34、表达式a\*b-c-d$e$f-g-h\*i中，运算符的优先级由高到低依次为-、\*、$，且均为右结合，则其后缀式为（ D ）。

A、ab\*c-d-e$fg-h-i\*$ B、$\*a-b-cd$e\*-f-ghi C、bcd--a\*efgh--i\*$$ D、abcd--\*efgh--i\*$$ E、ab\*c-d-e$fg-h-i\*$

35、a:= a+b\*c↑(d/e)/f的逆波兰记号表示是（ C ）。

A、aabc\*+↑de/f/:= B、aabcde↑/\*f/:= C、aabcde/↑\*f/+:= D、以上都不对。

三、简答题：

1、什么是语法制导翻译？

答：所谓语法制导翻译，是指在语法规则的制导下，通过计算语义规则，完成对输入符号的翻译。

由于使用属性文法时把语法规则和语义规则分开，但在使用语法规则进行推导或归约的同时又使用这些语义规则来指导翻译与最终产生目标代码，所以称为语法制导翻译。

2、写出算术表达式A+B\*(C-D)+E/(C-D)\*\*N 的三元式、四元式序列。

答：四元式： 三元式：

（-，C，D，T1） （1） （-，C，D）

（\*，B，T1，T2） （2）（\*，B，（1））

（+，A，T2，T3） （3） （+，A，（2））

（-，C，D，T4） （4） （-，C，D）

（\*\*，T4，N，T5） （5） （\*\*，（4），N）

（/，E，T5，T6） （6）（/，E，（5））

（+，T3，T6，T7） （7） （+，（3），（6））

3、写出条件语句 IF a<0 THEN x:=x+1 ELSE x:=4\*（x-1）的四元式形式。

答：

（1） （>，a，0，T1）

（2） （BMZ，T1， ，（6））

（3） （+，x，1，T2）

（4） （：=，T2， ，x）

（5） （BR， ， ，（9））

（6） （-，x，1，T3）

（7） （\*，4，T3，T4）

（8） （：=，T4， ，x）

（9）

4、把语句 IF A∨B<D then S1 else S2 翻译成一串四元式。

答：（1） （《，B，D，T1》

（2） （∨，A，T1 ，T2）

（3） （BMZ，T2， ，（6））

（4） S1的四元式序列

（5） （JMP， ， ，（7））

（6） S2的四元式序列

（7）

5、给出表达式A+B\*(C-D)-E/F↑G 的三元式、逆波兰和四元式表示。

答： A、三元式 B、四元式

（1）（-，C，D） （1）（-。C。D。T1）

（2）（\*，B，（1）） （2）（\*，B，T1，T2）

（3）（+，A，（2）） （3）（+，A，T2，T3）

（4）（↑，F，G） （4）（↑，F，G，T4）

（5）（/，E，（4）） （5）（/，E，T4，T5）

（6）（-，（3），（5）） （6）（-，T3，T5，T6）

C、逆波兰： ABCD-\*+EFG↑/-

6、给出算术表达式 -(a+b)\*(c+d)-(a+b+c)的四元式序列。

答：（1） （+，a，b，T1）

（2） （neg，T1， ，T2）

（3） （+，c，d，T3）

（4） （\*，T2，T3，T4）

（5） （+，a，b，T5）

（6） （+，T5，c，T6）

（7） （-，T4，T6，T7）

7、写出当型语句

while x+y>3 do

begin

a:=a+3\*b;

b:=a+e-f\*e;

end;

的四元式序列。

答： （1） （+，x，y，T1）

（2） （>，T1，3，T2）

（3） （BMZ，T2， ,(12)）

（4） （\*，3，b，T3）

（5） （+，a ，T3，T4）

（6） （：=，T4， ，a）

（7） （+，a，e，T5）

（8） （\*，f，e，T6）

（9） （-，T5，T6，T7）

（10） （：=，T7， ，b）

（11） （BR， ， ，（1））

（12）

8、将语句

if A V B>0 then while C>0 do C:=C+D

翻译成四元式。

答：

100 (jnz， A， -， 104)

101 (j， -， -， 102)

102 (j>， B， 0， 104)

103 (j， -， -， 109)

104 (j>， C， 0， 106)

105 (j， -， -， 109)

106 (+， C， D， T1)

107 (:=， T1， -， C)

108 (j， -， -， 104)

109

9、对下列文法G:

S'->#S#

S->D(R)

R->R;P|P

P->S|i

D->i

（1）计算文法G中每个非终结符的FIRSTVT集；

（2）计算文法G中每个非终结符的LASTVT集；

答：（1）各非终结符的FIRSTVT集合为：

FIRSTVT（S’）={ #} FIRSTVT（P）={（，i）

FIRSTVT（S）={（，i） FIRSTVT（D）={i }

FIRSTVT（R）={；，（，i ）

（2）各非终结符的LASTVT集合为：

LASTVT（S’）={#} LASTVT（P）={ }，i}

LASTVT（S）={} }

LASTVT（D）={ i}

LASTVT（R）={；，），i}

10、对下列文法G:

S'->#S#

S->fStS

S->i=E E->E+T|T

T->P↑T|P

P->(E)|i

（1）计算文法G中每个非终结符的FIRSTVT集；

（2）计算文法G中每个非终结符的LASTVT集；

答：（1）各非终结符的FIRSTVT集合为：

FIRSTVT（S’）={ #} FIRSTVT（T）={ ↑，（，i）

FIRSTVT（S）={f，i} FIRSTVT（E）={+，↑，（，i ）

FIRSTVT（P）={（，i ）

（2）各非终结符的LASTVT集合为：

LASTVT（S’）={#} LASTVT（T）={ ↑，}，i}

LASTVT（S）={ t，=，+，↑，)，i}

LASTVT（E）={ +，↑，}，i}

LASTVT（P）={ }，i}

11、文法G1：P->PaP|PbP|cP|Pe|f

证明文法G1是二义文法。

答：因为文法存在句型：fbfbf ，此句型有两棵不同的语法树，所以文法是二义的。

或存在2种最右推导：

1. P=>PbP=>PbPbP=>PbPbf=>Pbfbf=>fbfbf
2. P=>PbP=>Pbf=>PbPbf=>Pbfbf=>fbfbf

12、有文法G[N]：

N->SE|E

S->SD|D

E->0|2|4|6|8|10

D->0|1|2|3|4|5|6|7|8|9

证明该文法是二义的；此文法描述的语言是什么？并试写出另一文法，使L（G‘）=L（G），且G‘是无二义的。

答：对于该文法，存在句型110，有两棵不同的语法树或两种不同的最右推导，因此文法具有二义性。

A、 N=>SE=>S10=>D10=>110

B、 N=>SE=>S0=>SD0=>S10=>D10=>110

该文法描述的语言是偶数集合。

等价的无二义文法是：

G‘[N]：

N->SE|E

S->SD|D

E->0|2|4|6|8

D->0|1|2|3|4|5|6|7|8|9

13、写出不能被5整除的偶整数的文法。

答：G[Z]:

Z->(+|-)AB

A->0|1|2|3|4|5|6|7|8|9|AA

B->1|2|3|4|5|6|7|8|9

14、对于文法G=（VN，VT，S，P）：

VN={ S，A，B，}；VT={a，b}，开始符号为S

P：

S->aB|bA

A->aS|bAA|a

B->bS|aBB|b

给出串aaabbabbba的

1. 最左推导；（2）最右推导；（3）推导树。

答：最左推导：

S=>aB=>aaBB=>aaaBBB=>aaabSBB=>aaabbABB=>aaabbaBB=>aaabbabB=>aaabbabbS=>aaabbabbbA=>aaabbabbba

最右推导：

S=>aB=>aaBB=>aaBbS=>aaBbbA=>aaBbba=>aaaBBbba=>aaaBbbba=>aaabSbbba=>aaabbAbbba=>aaabbabbba

15、什么是句柄？

答：一个句型德最左直接短语称为句柄。

16、什么是最左素短语？

**17、**　上下文无关文法

*答：若一个形式文法 G = (N, Σ, P, S) 的产生式规则都取如下的形式：V -> w，则称之为上下文无关的，其中 V∈N ，w∈(N∪Σ)\* 。上下文无关文法取名为“上下文无关”的原因就是因为字符 V 总可以被字串 w 自由替换，而无需考虑字符 V 出现的上下文。上下文无关文法重要的原因在于它们拥有足够强的表达力来表示大多数程序设计语言的语法；*

**四、问答题：**

1、给出将赋值语句翻译成四元式的语法制导定义，允许右部表达式含有加法、乘法、取负、括号运算。生成赋值语句x:=b\*(c+d)+a的四元式。

2、出条件赋值语句 i:=if b then e1 else e2 的语义子程序。其中b是布尔表达式，e1和e2 是算术表达式，I代表与e1和e2类型相同的左部变量。按写出的语义子程序生成条件赋值语句 z:=if a>c then x+y else x-y+0.5 的四元式序列。

3、试写出PASCAL循环语句 for I:=1 to n do s的语义子程序，假定该语句的文法为：

F1:=for i:=1 to n

S:= F1 do S1

4、给出做为条件控制的布尔表达式翻译为四元式的语法制导定义，允许布尔表达式中有与、或、非及括弧运算。（如在翻译过程中使用了自定义的函数，可以不写函数过程，但请注明函数的功能和出入口的参数）。

5、按语法制导的定义将下面的后缀表达式翻译成中缀表达式。注意：不允许出现冗余括号。

E->E1E2+

E->E1E2\*

E->id

6、某程序设计语言说明部分的语法制导定义如下所示：

D->TL

T->int|real

L->L1,id|id

给出其语法制导定义及自底向上的翻译方案，并比较两者的不同。

7、设有文法G[S]：

S->E(1)

E->Aa(2)|Bb(3)

A->cA(4)|d(5)

B->cB(6)|d(7)

构造其LR（0）分析表并利用此分析表判断符号串acccd是否是该文法的句子。

8、对下列文法：

S’->S

S->bRST

S->bR

R->dSa

R->e

T->fRa

T->f

(1)出非终结符的FIRST和FOLLOW的集合。

（2）构造该文法的SLR（1）分析表。

9、给定文法 G[S] ：

S → SaA | a

A → AbS | b

⑴ 请构造该文法的以 LR(0) 项目集为状态的识别规范句型活前缀的 DFA 。

⑵ 请构造该文法的 LR(0) 分析表。

⑶ 什么是 LR(0) 文法？该文法是 LR(0) 文法吗？为什么？

⑷ 什么是 SLR(1) 文法？该文法是 SLR(1) 文法吗？为什么？

10、构造下列文法的LR（1）分析表，其中S’ 是文法的开始符号。

S’->S

S->Aa|dAb|Bb|dBa

A->c

B->c

11、对下面的文法G：

S’->bAAB|bA

A->dSa|b

B->cAa|c

（1）判断此文法是下列文法中的哪一种？并说明理由。

LR（0）、SLR（1）、LALR（1）、LR（1）

（2）构造相应文法的项目集族和分析表。

12、对下面的文法G[S]：

S->aBc|bAB

A->aAb|b

B->b|ε

构造其LL（1）分析表，并利用符号栈分析符号串baabbb是否是该文法的句子。

13、对下面的文法G：

E->E+T|T

T->T\*F|F

F->(E)|I

1. 构造其消除直接左递归后的文法G’；

（2）构造文法G’ 的LL（1）分析表，并利用符号栈分析符号串i1\*i2+i3是否是该文法的句子。

14、对下面的文法G：

P->begind;Xend

X->d;X|sY

Y->;sY|ε

构造文法G’ 的LL（1）分析表

15、对下面的文法G：

E->TE’

E’->+E|ε

T->FT’

T’->T|ε

F->PF’

F’->\*F’|ε

P->(E)|a|b|Λ

(1)计算这个文法的每个非终结符的FIRST和FOLLOW集合；

（2）证明这个文法是LL（1）的；

（3）构造它的预测分析表；

（4）构造它的递归下降分析程序。

16、对文法G(S)：

S → S ∨ a T | a T | ∨ a T

T → ∧ a T | ∧ a

(1) 消除该文法的左递归和提取左公因子；

(2) 构造各非终结符的FIRST和FOLLOW集合；

(3) 构造该文法的LL(1)分析表，并判断该文法是否是LL(1)的。

17、设字母表∑={ a，b}，给出∑上的一个正则式b\*abb\*(abb\*)。

（1）构造该正则式所对应的NFA（画出转换图）；

（2）将所求的NFA确定化（画出DFA的转换图）；

（3）将所求出的DFA最小化（画出极小化后的转换图）；

（4）该正则式所表示的正则集是什么？试用自然语言描述之。

18、设字母表∑={ a，b}，给出∑上的一个正则式（a \*|b\*）b(ba)\*。

（1）构造该正则式所对应的NFA（画出转换图）；

（2）将所求的NFA确定化（画出DFA的转换图）；

（3）将所求出的DFA最小化（画出极小化后的转换图）；

19、设有语言 L= {α | α ∈ {0,1} + ，且α不以 0 开头，但以 00 结尾 } 。

⑴ 试写出描述 L 的正规表达式；

⑵ 构造识别 L 的 DFA （要求给出详细过程，并画出构造过程中的 NDFA、DFA 的状态转换图，以及 DFA 的形式化描述 ) 。

20、设字母表∑={ a，b}，给出∑上的一个正则式（a|b）\*（aa|bb）（a|b）\*。

（1）构造该正则式所对应的NFA（画出转换图）；

（2）将所求的NFA确定化（画出DFA的转换图）；

（3）将所求出的DFA最小化（画出极小化后的转换图）；

21、设字母表∑={ a，b}，给出∑上的一个正则式（a|b）\*a（a|b）。

（1）构造该正则式所对应的NFA（画出转换图）；

（2）将所求的NFA确定化（画出DFA的转换图）；

（3）将所求出的DFA最小化（画出极小化后的转换图）；

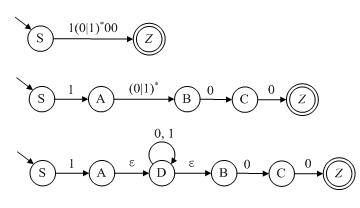
22、设有语言 L={ α | α∈ {0,1} + ，且α不以 0 开头，但以 00 结尾 } 。

⑴试写出描述 L 的正规表达式；

⑵构造识别 L 的 DFA （要求给出详细过程，并画出构造过程中的 NDFA 、 DFA 的状态转换图，以及 DFA 的形式化描述 ) 。

答：（ 1 ）正规表达式： 1(0|1) \* 00

（ 2 ）第一步：将正规表达式转换为 NDFA

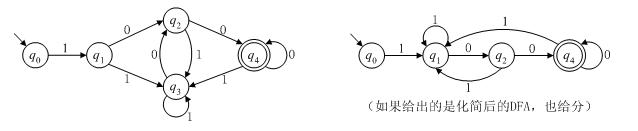


第二步：将 NDFA 确定化为 DFA ：

造表法确定化（ 3 分） 确定化后 DFA M 的状态转换表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 状态 输入 | I 0 | I 1 |  | t | 0 | 1 |
| [S] | — | [A,D,B] |  | q 0 | — | q 1 |
| [A,D,B] | [D,B,C] | [D,B] | 重新命名 | q 1 | q 2 | q 3 |
| [D,B,C] | [D,B,C,Z] | [D,B] |  | q 2 | q 4 | q 3 |
| [D,B] | [D,B,C] | [D,B] |  | q 3 | q 2 | q 3 |
| [D,B,C,Z] | [D,B,C,Z] | [D,B] |  | q 4 | q 4 | q 3 |

DFA 的状态转换图



第三步：给出 DFA 的形式化描述

DFA M = （ { q 0 , q 1 , q 2 , q 3 , q 4 }, {0,1}, t, q 0 , { q 4 } ）

t 的定义见 M 的状态转换表。

23、给定文法 G[S] ：

S → SaA|a

A → AbS|b

⑴请构造该文法的以 LR(0) 项目集为状态的识别规范句型活前缀的 DFA 。

⑵请构造该文法的 LR(0) 分析表。

⑶什么是 LR(0) 文法？该文法是 LR(0) 文法吗？为什么？

⑷什么是 SLR(1) 文法？该文法是 SLR(1) 文法吗？为什么？

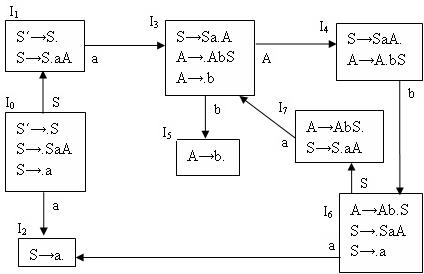
答： （1）拓广文法：

G[S′]: S′→ S ⑴ S → SaA ⑵ S → a ⑶

A → AbS ⑷ A → b ⑸

该文法的以 LR(0) 项目集为状态的识别规范句型活前缀的 DFA ：

⑵ 该文法的 LR(0) 分析表：



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 状态 | ACTION | | | GOTO | |
| a | b | # | S | A |
| 0 | S 2 |  |  | 1 |  |
| 1 | S 3 |  | acc |  |  |
| 2 | r 3 | r 3 | r 3 |  |  |
| 3 |  | S 5 |  |  | 4 |
| 4 | r 2 | r 2 /S 6 | r 2 |  |  |
| 5 | r 5 | r 5 | r 5 |  |  |
| 6 | S 2 |  |  | 7 |  |
| 7 | r 4 /S 3 | r 4 | r 4 |  |  |

⑶ LR(0) 文法：该文法的以 LR(0) 项目集为状态的识别规范句型活前缀的 DFA 中没有冲突状态。

该文法不是 LR(0) 文法，因为存在冲突状态： I 4 和 I 7。

⑷ SLR(1) 文法：该文法的以 LR(0) 项目集为状态的识别规范句型活前缀的 DFA 中有冲突状态，冲突可用 FOLLOW 集解决。

该文法不是 SLR(1) 文法。因为 FOLLOW(S)={a,b,#} ，所以无法解决冲突。

24、对下面的文法G：

S->aBc|bAB

A->aAb|b

B->b|ε

(1)计算这个文法的每个非终结符的FIRST和FOLLOW集合；

（2）证明这个文法是LL（1）的；

（3）构造它的预测分析表。

答：（1）first(B)={b, ε},first(A)={a,b},first(S)={a,b}

Follow(S)={#},follow(A)={b,#},follow(B)={c,#}

(2)因为只有FIRST(B)含有ε , 所以只考虑：

FOLLOW(B)=FIRST(c) ∪FOLLOW(S)={c, #}

∴该文法的LL(1)表

(3)该文法的LL(1)分析表如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a | b | c | # |
| S | aBc | bAB |  |  |
| A | aAb | b |  |  |
| B |  | b | ε | ε |
|  |  |  |  |  |

25、设字母表∑={ a，b}，对于以aa或ab结尾的字的正规集。

（1）请写出描述该语言的正规式。

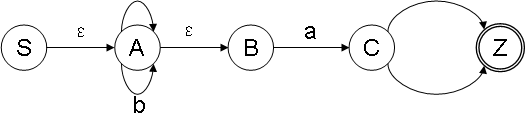
（2）构造该正规式所对应的NFA（画出转换图）；

（3）将所求的NFA确定化（画出DFA的转换图）；

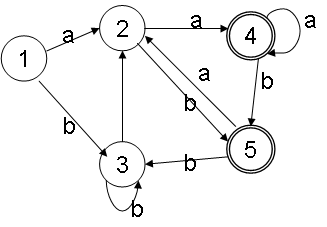
（4）将所求出的DFA最小化（画出极小化后的转换图）；

答： （1）正则式为：（a|b）\*a(a|b）。 （1分）

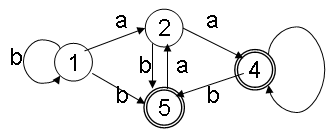
（2）由正则式得到的转换系统如下图： （1分）



（3）由子集法构造的DFA如下图：



（4）DFA最小化如下图：



26、设文法G为 S 🡪 E

E🡪 TE | ε

T 🡪 eT | t

(1)证明它是LR(1)文法；

(2)构造它的LR(1)分析表；

(3)给出输入符号串etet的分析过程。

答：(1)拓广文法G’：（1分）

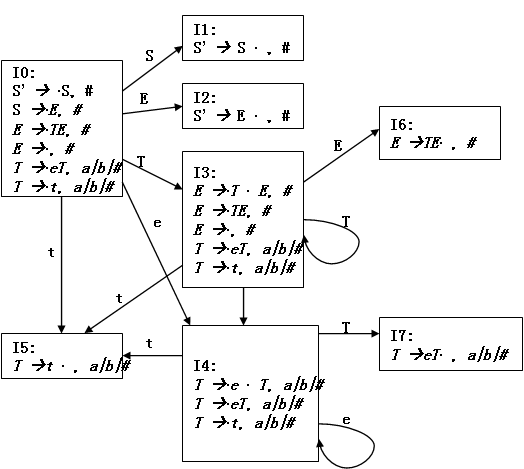
(0) S' 🡪S (1) S 🡪 E (2) E🡪 TE

(3) E 🡪ε (4) T 🡪 eT (5) T 🡪 t

FIRST(A) = {ε, e,t}

FIRST(B) = {e, t}

构造的DFA如下:



由项目集规范族看出，不存在冲突动作。

∴该文法是LR(1)文法。

(2)

LR(1)分析表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 状态 | Action | | | Goto | | |
| e | t | # | S | E | T |
| 0 | S4 | S5 | r3 | 1 | 2 | 3 |
| 1 |  |  | acc |  |  |  |
| 2 |  |  | r1 |  |  |  |
| 3 | S4 | S5 | r3 |  | 6 | 3 |
| 4 | S4 | S5 |  |  |  | 7 |
| 5 | r5 | r5 | r5 |  |  |  |
| 6 |  |  | r2 |  |  |  |
| 7 | r4 | r4 | r4 |  |  |  |

(3)输入串abab的分析过程为：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 步骤 | 状态栈 | 符号栈 | 当前字符 | 剩余字符串 | 动作 |
| (1) | 0 | # | e | tet# | 移进 |
| (2) | 04 | #e | t | et# | 移进 |
| (3) | 045 | #et | e | t# | 归约 T🡪t |
| (4) | 047 | #eT | e | t# | 归约 T🡪et |
| (5) | 03 | #T | e | t# | 移进 |
| (6) | 034 | #Te | t | # | 移进 |
| (7) | 0345 | #TeT | # |  | 归约 T🡪t |
| (8) | 0347 | #TeT | # |  | 归约 T🡪et |
| (9) | 033 | #TT | # |  | 归约 E🡪ε |
| (10) | 0336 | #TTE | # |  | 归约 E🡪TE |
| (11) | 036 | #TE | # |  | 归约 E🡪TE |
| (12) | 02 | # E | # |  | 归约 S🡪E |
| (13) | 01 | #S | # |  | acc |

27、对文法G(S)：

S → S ∨ a T | a T | ∨ a T

T → ∧ a T | ∧ a

(1) 消除该文法的左递归和提取左公因子；

(2) 构造各非终结符的FIRST和FOLLOW集合；

(3) 构造该文法的LL(1)分析表，并判断该文法是否是LL(1)的。

答：（1）消除左递归

S → a T S’ | ∨ a T S’

S’ → ∨ a T S’ | ε

T → T → ∧ a T | ∧ a

提取左公因子

S → a T S’ | ∨ a T S’

S’ → ∨ a T S’ | ε

T → ∧ a T’

T’ → T | ε

(2) 各非终结符的FIRST和FOLLOW集合：

FIRST(S)={a，∨} FIRST(S')={ ∨，ε} FIRST(T)={ ∨ }

FIRST(T')={ ∨，ε} FOLLOW(S)={#} FOLLOW(S')={#}

FOLLOW(T)={∨，#} FOLLOW(T')={∨，#}

(3) LL(1)分析表如下(3分) ，该文法是LL(1)文法。（1分）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a | ∨ | ∧ | # |
| S | S→aTS' | S→∨aTS' |  |  |
| S' |  | S'→∨aTS' |  | S’→ε |
| T |  |  | T→∧aT' |  |
| T' |  | T'→ε | T'→T | T'→ε |