

一、单项选择题(共 10 小题, 每小题 2 分) (题分 20 分)

1. 语言是
 - A. 句子的集合
 - B. 产生式的集合
 - C. 符号串的集合
 - D. 句型的集合
2. 编译程序前三个阶段完成的工作是
 - A. 词法分析、语法分析和代码优化
 - B. 代码生成、代码优化和词法分析
 - C. 词法分析、语法分析、语义分析和中间代码生成
 - D. 词法分析、语法分析和代码优化
3. 一个句型中称为句柄的是该句型的最左
 - A. 非终结符号
 - B. 短语
 - C. 句子
 - D. 直接短语
4. 下推自动机识别的语言是
 - A. 0 型语言
 - B. 1 型语言
 - C. 2 型语言
 - D. 3 型语言
5. 扫描器所完成的任务是从字符串形式的源程序中识别出一个个具有独立含义的最小语法单位即
 - A. 字符
 - B. 单词
 - C. 句子
 - D. 句型
6. 对应 Chomsky 四种文法的四种语言之间的关系是
 - A. $L_0 \subset L_1 \subset L_2 \subset L_3$
 - B. $L_3 \subset L_2 \subset L_1 \subset L_0$
 - C. $L_3 = L_2 \subset L_1 \subset L_0$
 - D. $L_0 \subset L_1 \subset L_2 = L_3$
7. 词法分析的任务是
 - A. 识别单词
 - B. 分析句子的含义
 - C. 识别句子
 - D. 生成目标代码
8. 常用的中间代码形式不含
 - A. 三元式
 - B. 四元式
 - C. 逆波兰式
 - D. 语法树
9. 代码优化的目的是
 - A. 节省时间
 - B. 节省空间
 - C. 节省时间和空间
 - D. 把编译程序进行等价交换
10. 代码生成阶段的主要任务是
 - A. 把高级语言翻译成汇编语言
 - B. 把高级语言翻译成机器语言
 - C. 把中间代码变换成依赖具体机器的目标代码
 - D. 把汇编语言翻译成机器语言

二、填空题(本大题共 5 小题, 每小题 2 分) (题分 10 分)

1. 编译程序首先要识别出源程序中每个(), 然后再分析每个()并翻译其意义。
2. 编译器常用的语法分析方法有()和()两种。
3. 通常把编译过程分为分析前端与综合后端两大阶段。词法、语法和语义分析是对源程序的(), 中间代码生成、代码优化与目标代码的生成则是对源程序的()。
4. 程序设计语言的发展带来了日渐多变的运行时存储管理方案, 主要分为两大类, 即: ()方案和()方案。

5. 对编译程序而言, 输入数据是(), 输出结果是()。

三、名词解释题(共 5 小题, 每小题 4 分) (题分 20 分)

1. 词法分析
2. LL(1)文法
3. 语法树
4. LR(0)分析器
5. 语言和文法

四、简答题(共 4 小题, 每小题 5 分) (题分 20 分)

1. 编译程序和高级语言有什么区别?
2. 编译程序的工作分为那几个阶段?
3. 简述自下而上的分析方法。
4. 简述代码优化的目的和意义。

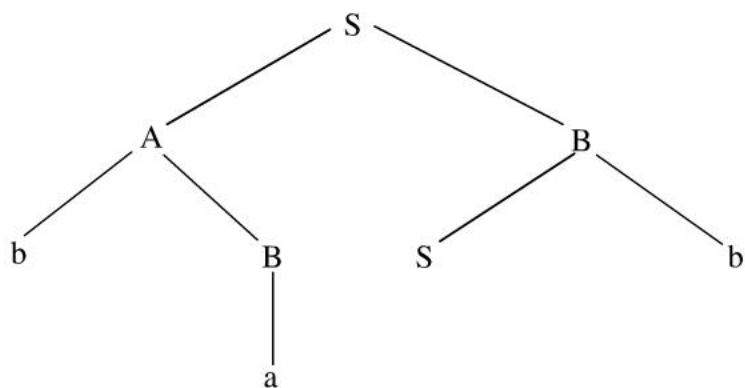
五、综合应用题(共 3 小题, 每小题 10 分) (题分 30 分)

1. 证明下述文法 G:

$$S \rightarrow aSbSlaSld$$

是二义性文法。

2. 对于文法 $G[S]: S \rightarrow AB, A \rightarrow AalbB, B \rightarrow alSb$ 求句型 $baSb$ 的全部短语、直接短语和句柄? 句型 $baSb$ 的语法树如图五(2)所示。



图五(2) 句型 $baSb$ 的的语法树

3. 设有非确定的有自限动机 NFA $M = (\{A, B, C\}, \{0, 1\}, \delta, \{A\}, \{C\})$, 其中:
 $\delta(A, 0) = \{C\}$ $\delta(A, 1) = \{A, B\}$ $\delta(B, 1) = \{C\}$ $\delta(C, 1) = \{C\}$ 。请画出状态转换矩阵和状态转换图。

参考答案

一、单项选择题(共 10 小题, 每小题 2 分, 共 20 分)

1. 语言是

- A. 句子的集合 B. 产生式的集合
C. 符号串的集合 D. 句型的集合
2. 编译程序前三个阶段完成的工作是
A. 词法分析、语法分析和代码优化
B. 代码生成、代码优化和词法分析
C. 词法分析、语法分析、语义分析和中间代码生成
D. 词法分析、语法分析和代码优化
3. 一个句型中称为句柄的是该句型的最左
A. 非终结符号 B. 短语 C. 句子 D. 直接短语
4. 下推自动机识别的语言是
A. 0 型语言 B. 1 型语言
C. 2 型语言 D. 3 型语言
5. 扫描器所完成的任务是从字符串形式的源程序中识别出一个个具有独立含义的最小语法单位即
A. 字符 B. 单词 C. 句子 D. 句型
6. 对应 Chomsky 四种文法的四种语言之间的关系是
A. $L_0 \subset L_1 \subset L_2 \subset L_3$ B. $L_3 \subset L_2 \subset L_1 \subset L_0$
C. $L_3 = L_2 \subset L_1 \subset L_0$ D. $L_0 \subset L_1 \subset L_2 = L_3$
7. 词法分析的任务是
A. 识别单词 B. 分析句子的含义
C. 识别句子 D. 生成目标代码
8. 常用的中间代码形式不含
A. 三元式 B. 四元式 C. 逆波兰式 D. 语法树
9. 代码优化的目的是
A. 节省时间 B. 节省空间
C. 节省时间和空间 D. 把编译程序进行等价交换
10. 代码生成阶段的主要任务是
A. 把高级语言翻译成汇编语言
B. 把高级语言翻译成机器语言
C. 把中间代码转换成依赖具体机器的目标代码
D. 把汇编语言翻译成机器语言

二、填空题（本大题共 5 小题，每小题 2 分，共 10 分）

- 编译程序首先要识别出源程序中每个(单词)，然后再分析每个(句子)并翻译其意义。
- 编译器常用的语法分析方法有(自底向上)和(自顶向下)两种。
- 通常把编译过程分为分析前端与综合后端两大阶段。词法、语法和语义分析是对源程序的(分析)，中间代码生成、代码优化与目标代码的生成则是对源程序的(综合)。
- 程序设计语言的发展带来了日渐多变的运行时存储管理方案，主要分为两大类，即(静态存储分配)方案和(动态存储分配)方案。
- 对编译程序而言，输入数据是(源程序)，输出结果是(目标程序)。

三、名词解释题(共 5 小题，每小题 4 分，共 20 分)

- 词法分析

词法分析的主要任务是从左向右扫描每行源程序的符号，按照词法规则从构成源程序的字符串中识别出一个个具有独立意义的最小语法单位，并转换成统一的内部表示(token)，送给语法分析程序。

2. LL(1)文法

若文法的任何两个产生式 $A \rightarrow \alpha \mid \beta$ 都满足下面两个条件：

(1) $\text{FIRST}(\alpha) \cap \text{FIRST}(\beta) = \emptyset$;

(2) 若 $\beta \Rightarrow^* \varepsilon$ ，那么 $\text{FIRST}(\alpha) \cap \text{FOLLOW}(A) = \emptyset$ 。

我们把满足这两个条件的文法叫做 **LL(1)文法**，其中的第一个 L 代表从左向右扫描输入，第二个 L 表示产生最左推导，1 代表在决定分析器的每步动作时向前看一个输入符号。除了没有公共左因子外，LL(1)文法还有一些明显的性质，它不是二义的，也不含左递归。

3. 语法树

句子的树结构表示法称为语法树(语法分析树或语法推导树)。

给定文法 $G=(V_N, V_T, P, S)$ ，对于 G 的任何句型都能构造与之关联的语法树。这棵树具有下列特征：

(1)根节点的标记是开始符号 S。

(2)每个节点的标记都是 V 中的一个符号。

(3)若一棵子树的根节点为 A，且其所有直接子孙的标记从左向右的排列次序为 $A_1 A_2 \dots A_R$ ，那么 $A \rightarrow A_1 A_2 \dots A_R$ 一定是 P 中的一条产生式。

(4)若一标记为 A 的节点至少有一个除它以外的子孙，则 $A \in V_N$ 。

(5)若树的所有叶节点上的标记从左到右排列为字符串 w，则 w 是文法 G 的句型；若 w 中仅含终结符号，则 w 为文法 G 所产生的句子。

4. LR(0)分析器

所谓 LR(0)分析，是指从左至右扫描和自底向上的语法分析，且在分析的每一步，只须根据分析栈当前已移进和归约出的全部文法符号，并至多再向前查看 0 个输入符号，就能确定相对于某一产生式左部符号的句柄是否已在分析栈的顶部形成，从而也就可以确定当前所应采取的分析动作 (是移进还是按某一产生式进行归约等)。

5. 语言和文法

文法就是语言结构的定义和描述，是有穷非空的产生式集合。

文法 G 定义为四元组的形式：

$$G=(V_N, V_T, P, S)$$

其中： V_N 是非空有穷集合，称为非终结符号集合； V_T 是非空有穷集合，称为终结符号集合；P 是产生式的集合(非空)；S 是开始符号(或识别符号)。这里， $V_N \cap V_T = \emptyset$ ， $S \in V_N$ 。 $V = V_N \cup V_T$ ，称为文法 G 的字母表，它是出现文法产生式中的一切符号的集合。

文法 G 所描述的语言用 $L(G)$ 表示，它由文法 G 所产生的全部句子组成，即

$$L(G)=\{x \mid S \Rightarrow^* x, \text{ 其中 } S \text{ 为文法开始符号, 且 } x \in V_T^+ \}$$

简单的说，文法描述的语言是该文法一切句子的集合。

四、简答题(共 4 小题, 每小题 5 分, 共 20 分)

1. 编译程序和高级语言有什么区别?

用汇编语言或高级语言编写的程序, 必须先送入计算机, 经过转换成用机器语言表示的目标程序(这个过程即编译), 才能由计算机执行。执行转换过程的程序叫编译程序。汇编程序是指没有编译过的汇编语言源文件。编译程序转换过的叫目标程序, 也就是机器语言。

编译程序的工作情况有三种: 汇编型、解释型和编译型。汇编型编译程序用来将汇编语言编写的程序, 按照一一对应的关系, 转换成用机器语言表示的程序。解释型编译程序将高级语言程序的一个语句, 先解释成为一组机器语言的指令, 然后立即执行, 执行完了, 取下一组语句解释和执行, 如此继续到完成一个程序止。用解释型编译程序, 执行速度很慢, 但可以进行人和计算机的"对话", 随时可以修改高级语言的程序。BASIC 语言就是解释型高级语言。编译型编译程序将高级语言编写的程序, 一次就会全部翻译成机器语言表示的程序, 而且过程进行很快, 在过程中, 不能进行人机对话修改。FORTRAN 语言就是编译型高级语言。

2. 编译程序的工作分为那几个阶段?

词法分析、语法分析和语义分析是对源程序进行的分析(称为编译程序的前端), 而中间代码生成、代码优化和代码生成三个阶段合称为对源程序进行综合(称为编译程序的后端), 它们从源程序的中间表示建立起和源程序等价的目标程序。

3. 简述自下而上的分析方法。

所谓自下而上分析法就是从输入串开始, 逐步进行“归约”, 直至归约到文法的开始符号; 或者说从语法树的末端开始, 步步向上“归约”, 直到根节点。

4. 简述代码优化的目的和意义。

代码优化是尽量生成“好”的代码的编译阶段。也就是要对程序代码进行一种等价变换, 在保证变换前后代码执行结果相同的前提下, 尽量使目标程序运行时所需要的时间短, 同时所占用的存储空间少。

五、综合应用题(共 3 小题, 每小题 10 分, 共 30 分)

1. 证明下述文法 G:

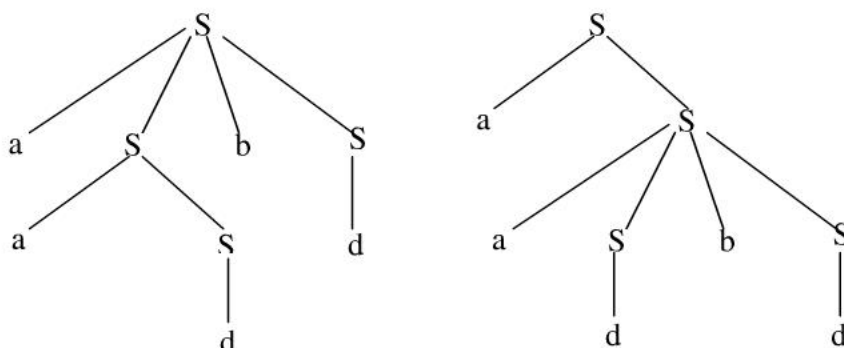
$$S \rightarrow aSbSlaSld$$

是二义性文法。

解:

一个文法, 如果存在某个句子有不只一棵语法分析树与之对应, 那么称这个文法是二义性文法。

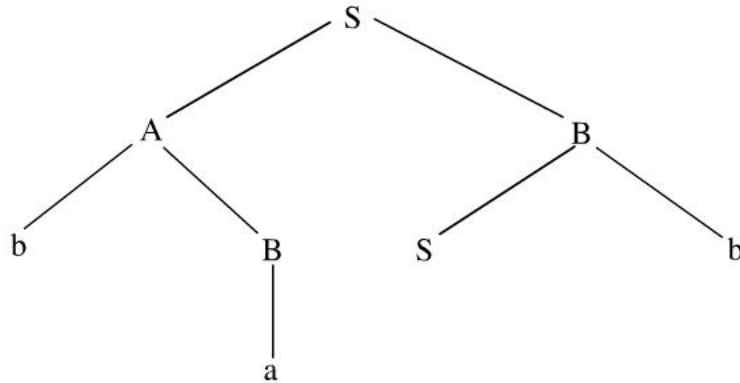
句子 aadbld 有两棵语法树。如下图:



(1) _____ (2)

由此可知， $S \rightarrow aSbSlaSld$ 定义的文法是二义性文法。

2. 对于文法 $G[S]: S \rightarrow AB, A \rightarrow AalbB, B \rightarrow alSb$ 求句型 $baSb$ 的全部短语、直接短语和句柄？
句型 $baSb$ 的语法树如图五(2)所示。



图五(2) 句型 $baSb$ 的的语法树

解：

$baSb$ 为句型 $baSb$ 的相对于 S 的短语， ba 为句型 $baSb$ 的相对于 A 的短语， Sb 为句型 $baSb$ 的相对于 B 的短语，且为直接短语， a 为句型 $baSb$ 的相对于 B 的短语，且为直接短语和句柄。

3. 设有非确定的有自限动机 NFA $M = (\{A, B, C\}, \{0, 1\}, \delta, \{A\}, \{C\})$ ，其中：

$\delta(A, 0) = \{C\}$ $\delta(A, 1) = \{A, B\}$ $\delta(B, 1) = \{C\}$ $\delta(C, 1) = \{C\}$ 。请画出状态转换矩阵和状态转换图。

解：

状态转换矩阵为：

δ	0	1
A	C	A, B
B	\emptyset	C
C	\emptyset	C

状态转换图为：

