**复习大纲**

记忆并理解

* 编译程序五阶段的划分与作用
* 各个编译阶段的输入与输出

掌握计算方法

* 乔姆斯基文法的分类；
* 根据文法定义进行推导与规约；
* 正则表达式
  + 正则表达式到NFA的转换
  + 使用子集构造法将NFA转DFA
  + DFA最小化
* LL分析法
  + First集和Follow集的计算
  + LL(1)分析表的计算
  + LL(1)文法的判别
  + 消除左递归，提取公共左子因子
* LR分析法
  + LR(0)自动机的计算
  + LR、SLR、LALR文法的判别与证明
* 中间代码表示
  + 中缀表达式、后缀表达式、三元式、四元式间的互相转换
* 优化技术
  + 消除公共表达式、复写传播、消除无用表达式等

掌握各种技术的使用场景与优劣

* 符号表实现方式的选择与比较
* 词法分析技术的选择与比较
* 语法分析技术的选择与比较
* 中间语言的选择与比较
* 内存管理技术的选择与比较
* 编译功能变化时，需要修改编译程序的哪个部分

**练习题**

**一、简答题**

1.源代码中括号不匹配的错误一般在编译的哪个阶段（采用五阶段划分模型）被检查出来，简述这一阶段的名称和主要任务。

2.请简述什么是文法的二义性。

**二、计算题**

已知文法G(S)的产生式集为：

请判断这个文法属于乔姆斯基几型文法，并判断下列符号串是否是这个文法的一个句子，如果是则给出这个句子的最左推导和语法树。

1.baa

2.aba

**三、计算题**

现有正则表达式

1.请将上述正则表达式转换为等价的NFA，画出所得NFA的状态转移图。

2.请使用子集构造法将1中得到的NFA转换成等价的DFA，画出所得DFA的状态转移图。

3.请使用划分法将2中得到的DFA转换成等价的最小DFA，画出所得最小DFA的状态转移图。

**四、计算题**

使用LR分析法对文法G(S)进行分析，G(S)的产生式集为

1.请给出这一文法的LR(0)自动机。

2.请证明这个方法是SLR(1)文法。

**五、计算题**

文法G(S)的产生式集为

1.请给出各非终结符的First集合和Follow集合。

2.请给出LL(1)分析表。

**六、计算题**

现有算术表达式1+2\*3\*4-5，四则运算的优先顺序按通常意义来理解

1.请转换为逆波兰表达式（后缀式）。

2.请转换为三地址代码中的四元式。

**七、计算题**

以下基本块中，均为内部使用的临时变量，在此基本块之外不再使用，请使用提取公共表达式、复写传播、消除无用表达式三种优化技术进行优化：

**八、设计题**

现在假设由你来设计一种语言，并给这种语言实现一个编译器，这个编译器采用标准的五阶段实现。

1.你希望程序运行尽可能快，因此准备使用多种优化技术，请写出五种你准备使用的不同的优化技术名称。

2.原来你的程序运行很好，但当你准备将程序移植到新的平台上时，你发现新平台的处理器比原来的处理器少3个寄存器，你最可能需要修改编译器的哪个部分，这个部分的功能是什么。

3.最开始你的语言只支持使用英文字母作为标识符，现在你准备支持英文字母和下划线作为标识符，请问最可能需要修改编译器的哪个部分，这个部分的功能是什么。

4.你设计的语言没有显式的变量定义语句，而是在首次赋值时定义，即不需要定义int a;可以直接通过a=1将变量a定义为int型变量并设置为初始值1。这个功能很方便，但有用户认为加入显式的变量定义语句可以在编译期检查出更多的错误，你决定采纳他的建议进行修改，请简述你会修改编译器的哪些部分，如何修改。

**练习题参考答案**

**一、简答题**

1.源代码中括号不匹配的错误一般在编译的哪个阶段（采用五阶段划分模型）被检查出来，简述这一阶段的名称和主要任务。

答：名称：语法分析。

主要任务：读取词法符号，进行语法分析，判断源程序是否符合程序语法，并生成抽象语法树或语法符号。

2.请简述什么是文法的二义性。

答：文法的一个句子有两个不同的语法树/最左推导/最右推导，则称这一文法具有二义性。

**二、计算题**

已知文法G(S)的产生式集为：

请判断这个文法属于乔姆斯基几型文法，并判断下列符号串是否是这个文法的一个句子，如果是则给出这个句子的最左推导和语法树。

1.baa

2.aba

答：1.baa是这个文法的一个句子。

最左推导为：。

语法树为：

黑色的钟表

中度可信度描述已自动生成

2. aba不是这个文法的一个句子，因为该文法的句子一定以b开头。

**三、计算题**

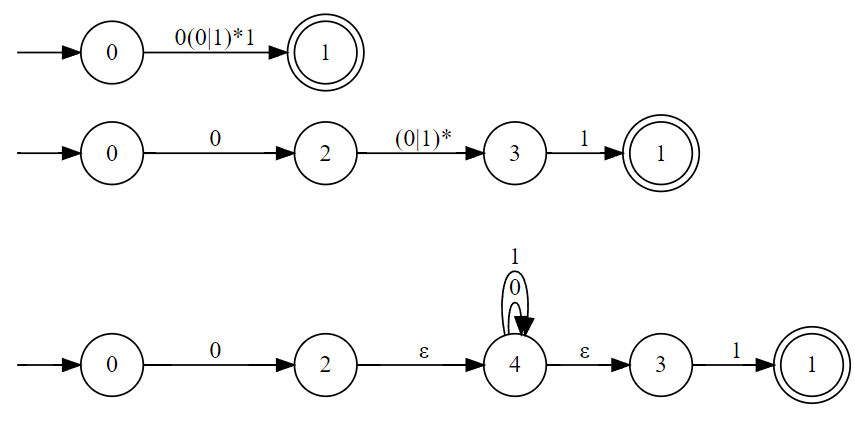
现有正则表达式

1.请将上述正则表达式转换为等价的NFA，画出所得NFA的状态转移图。

2.请使用子集构造法将1中得到的NFA转换成等价的DFA，画出所得DFA的状态转移图。

3.请使用划分法将2中得到的DFA转换成等价的最小DFA，画出所得最小DFA的状态转移图。

答：1.上述正则表达式转换为NFA的过程如下图：

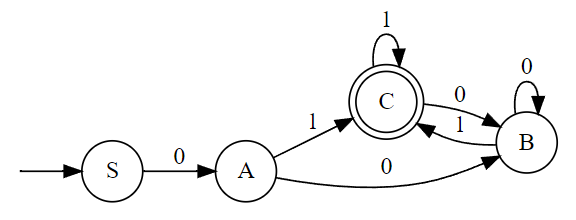


2.使用子集法，得到与NFA状态集对应的DFA状态及其转换关系，如下表：

|  |  |  |
| --- | --- | --- |
| 状态 | 0 | 1 |
| {0}S | {2,4,3}A | / |
| {2,4,3}A | {4,3}B | {4,3,1}C |
| {4,3}B | {4,3}B | {4,3,1}C |
| {4,3,1}C | {4,3}B | {4,3,1}C |

以上状态中，S是起始状态，C是终止状态。

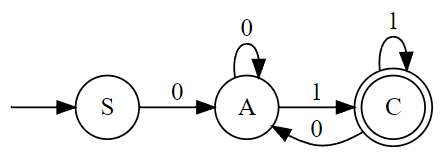
状态转换图如下：



3.初始划分为{S, A, B}, {C}

最终划分为{S}, {A, B}, {C}

对应的最小DFA状态转移图为：



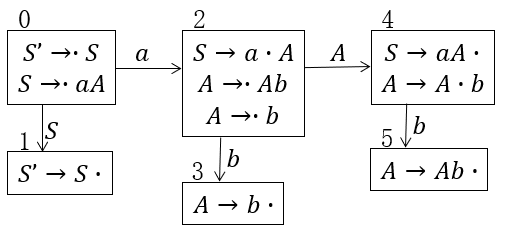
**四、计算题**

使用LR分析法对文法G(S)进行分析，G(S)的产生式集为

1.请给出这一文法的LR(0)自动机。

2.请证明这个方法是SLR(1)文法。

答：1.LR(0)自动机如下图所示：



2. 由于项目集4有移进归约冲突，因此不是LR(0)文法。

移进部分接收输入字符b，归约部分归约到S，S的Follow集为{#}，不包含b，这一冲突可用SLR(1)解决。其他项目集均没有移进归约冲突。综上，其为SLR(1)文法。

**五、计算题**

文法G(S)的产生式集为

1.请给出各非终结符的First集合和Follow集合。

2.请给出LL(1)分析表。

答：1.集合如下表所示：

|  |  |  |
| --- | --- | --- |
|  | First | Follow |
| S | {a, b, c} | {#} |
| A | {b, c} | {#, c} |

2.LL(1)分析表如下所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a | b | c | # |
| S |  |  |  |  |
| A |  |  |  |  |

**六、计算题**

现有算术表达式1+2\*3\*4-5，四则运算的优先顺序按通常意义来理解

1.请转换为逆波兰表达式（后缀式）。

2.请转换为三地址代码中的四元式。

答：1.转换为逆波兰表达式为：

123\*4\*+5-

2.转换为四元式为：

|  |  |  |  |
| --- | --- | --- | --- |
| op | arg1 | arg2 | result |
| \* | 2 | 3 |  |
| \* |  | 4 |  |
| + |  |  |  |
| - |  | 5 |  |

**七、计算题**

以下基本块中，均为内部使用的临时变量，在此基本块之外不再使用，请使用提取公共表达式、复写传播、消除无用表达式三种优化技术进行优化：

答：依次使用三种优化技术，结果如下：

|  |  |  |
| --- | --- | --- |
| 提取公共表达式： | 复写传播： | 消除无用代码 |

**八、设计题**

现在假设由你来设计一种语言，并给这种语言实现一个编译器，这个编译器采用标准的五阶段实现。

1.你希望程序运行尽可能快，因此准备使用多种优化技术，请写出五种你准备使用的不同的优化技术名称。

2.原来你的程序运行很好，但当你准备将程序移植到新的平台上时，你发现新平台的处理器比原来的处理器少3个寄存器，你最可能需要修改编译器的哪个部分，这个部分的功能是什么。

3.最开始你的语言只支持使用英文字母作为标识符，现在你准备支持英文字母和下划线作为标识符，请问最可能需要修改编译器的哪个部分，这个部分的功能是什么。

4.你设计的语言没有显式的变量定义语句，而是在首次赋值时定义，即不需要定义int a;可以直接通过a=1将变量a定义为int型变量并设置为初始值1。这个功能很方便，但有用户认为加入显式的变量定义语句可以在编译期检查出更多的错误，你决定采纳他的建议进行修改，请简述你会修改编译器的哪些部分，如何修改。

答：（本题答案不唯一）

1.常量合并、复写传播、无用代码消除、循环展开、函数内联、公共子表达式消除、强度削弱、删除归纳变量等。

2.寄存器为机器相关的部分，寄存器分配在目标代码生成阶段执行，因此需要修改目标代码生成。这一阶段的功能是将中间代码转换成目标机器上的汇编或者机器语言代码，并执行机器相关优化。

3.标识符的识别过程涉及到词法符号的解析，需要修改语言的词法，对应的分析过程是词法分析。这一阶段的功能是将将源代码解析为词法符号。

4.需要修改程序的语法分析、语义分析和中间代码生成部分。语法中，加入对变量定义的支持，因而语法分析器需要对应进行修改；在语义分析时，增加变量是否已经定义的检查部分。