

《 编 译 原 理 》

课 程 设 计 指 导 书

李宏芒 编 写

适用专业：计算机科学与技术

合肥工业大学计算机与信息学院

2022 年 3 月

《编译原理》是计算机专业的一门重要的专业课程，其中包含大量软件设计思想。大家通过课程设计，实现一些重要的算法，或设计一个完整的编译程序模型，能够进一步加深理解和掌握所学知识，对提高自己的软件设计水平具有十分重要的意义。课程设计的内容主要包括：

- ✓ 词法分析。利用状态转换图设计词法分析器。从正规式构造非确定有限自动机（NFA）。用子集法把非确定有限自动机（NFA）确定化为确定有限自动机（DFA）。确定有限自动机（DFA）状态最少化。
- ✓ 语法分析。自上而下分析。递归子程序分析和预测分析。自下而上分析。算符优先分析法和 LR 分析程序。
- ✓ 语义分析和中间代码产生。基于一遍扫描的语法制导翻译方法。算术表达式、赋值语句、布尔表达式、控制语句等语法单位的翻译模式。

大家在进行课程设计时，可以选择本指导书提供给大家的一些参考选题，或者可以从上述课程设计的内容中选择某个主题，抽象成一个模型，以确定自己的设计题目，上机完成软件开发。软件开发可选择 C++、C#、Java 等语言（也可以是你熟悉的任何语言）。

最后每位同学都要认真撰写课程设计报告，**格式要规范，内容要详尽**，主要包括：

- ✓ 设计题目（封面含姓名、学号等），
- ✓ 设计目的及设计要求
- ✓ 设计内容（问题的描述及解决的方法）、主要算法描述（流程图），
- ✓ 设计的输入和输出形式
- ✓ 程序运行（测试、模拟）的结果（屏幕拷贝、生成结果的打印输出），
- ✓ 总结（体会）
- ✓ 源程序清单（部分核心代码）作为报告的附件。

希望每个同学尽可能不要都选择完全一样的题目（同一个班学生不重复选题）。同学原则上需要选择本指导书提供的题目，也可以自主选题（注：自主命题的知识点难度系数需要达到“***”级别，也鼓励同学尝试一些更难的课题），

鼓励选择有一定难度、较大工作量、综合性较强的题目，在评定成绩时将会给予好的成绩。

学生原则上需要按照《编译原理课程设计指导书》中的具体要求独立完成课题的

设计和软件开发。

现场验收：针对每个学生完成课程设计的实现细节进行现场验收，主要根据完成的难度（包括知识点难度和开发技术难度）、设计思想、代码量、独立性、开发平台选择、用户友好性等进行综合评判，现场验收的过程与结果统一登记记录，为最后成绩的评判提供依据。

实验的最终成绩是根据“现场验收”和“课程设计报告”综合评判而得出的，现场验收往往更能考查出学生完成的真实情况，因此现场验收是主要的衡量标准，但是课程设计报告具有一票否决权，不管现场验收成绩多好，只要课程设计报告不合格，也将不会获得好成绩，如没有提交课程设计报告，课程设计成绩必定做 0 分计算。

注：关于课题难度系数的说明，课题难度包括知识点难度和开发技术难度，同学们即使选择了知识点难度较低的课题，也可以通过软件设计和开发技术难度的增加提高整个课题的难度系数。本课程设计指导书中只给出了每一个课题的知识点难度系数供同学参考，软件设计和开发技术的难度系数由于涉及面较广（如开发平台、软件模式、设计理念、用户友好性等），同学可充分发挥主观能动性自我把握，鼓励技术创新，现场验收时进行评估。

编译原理课程设计部分参考选题：

1. 题目： FORTRAN 语言实型常数识别程序设计

设计内容及要求： 将教材 P. 41 的图 3. 2 (d) 识别 FORTRAN 实型常数的状态转换图用程序实现。程序能够从用户输入的任意一个字符串中识别出 FORTRAN 实型常数，显示输出。（**知识点难度系数 ***）

2. 题目： 简化的 FORTRAN 语言词法分析程序设计

设计内容及要求： 将教材 P. 42 上的表 3. 1 的词法分析器构造出来，限制条件如教材所述。保留字的识别按标识符一样识别，通过查找保留字表区分是保留字还是标识符。程序能够从用户输入的源程序中，识别出的单词符号，并用二元式表示，显示输出或输出到文件中。（**知识点难度系数 ***）

3. 题目： ϵ -CLOSURE (I) 构造算法的程序实现

设计内容及要求： 将 ϵ -CLOSURE (I) 构造算法用程序实现。要求：对任意给定的一个 NFA M (其状态转换矩阵及初态、终态信息保存在指定文件中) 的某一个状态子集 I，显示输出构造出的 ϵ -CLOSURE (I)。（**知识点难度系数 ***）

4. 题目： 从右线性文法构造与之等价的有限自动机的程序实现

设计内容及要求： 构造一转换程序，实现将用户任意给定的右线性文法，

转换为与之等价的有限自动机 FA M，输出其状态转换矩阵（显示输出或输出到文件中）。（知识点难度系数 *）

5. 题目： 从有限自动机构造与之等价的右线性文法的程序实现

设计内容及要求：构造一转换程序，实现将用户任意给定的有限自动机 FA M，转换为与之等价的右线性文法，显示输出或输出到文件中。（知识点难度系数 *）（知识点难度系数 *）

6. 题目： 有限自动机的状态转换图显示程序的实现

设计内容及要求：构造一程序，实现：将任一给定的有限自动机 M(其状态转换矩阵及初态、终态信息保存在指定文件中)，在屏幕上显示输出 M 的状态转换图。程序应具有通用性，状态节点在屏幕上的分布应合理、美观。（知识点难度系数 *）

7. 题目： 从 NFA 构造与之等价的正规式 r 的程序实现

设计内容及要求：对给定的任意 NFA M(其状态转换矩阵及初态、终态信息分别保存在指定文件中)。构造一程序，从 NFA 构造与之等价的正规式 r，并显示输出。（知识点难度系数 *）

8. 题目： 构造正规式 $r_1|r_2$ （或运算）的 NFA 的程序实现

设计内容及要求：对给定的正规式 r_1 、 r_2 ，已知它们的 NFA 分别为 M_1 、 M_2 (其状态转换矩阵及初态、终态信息分别保存在指定文件中)。构造一程序，由此程序构造正规式 $r_1|r_2$ （或运算）的 NFA(将其状态转换矩阵及初态、终态信息保存在指定文件中)。（知识点难度系数 *）

9. 题目： 构造正规式 r_1r_2 （连接运算）的 NFA 的程序实现

设计内容及要求：对给定的正规式 r_1 、 r_2 ，已知它们的 NFA 分别为 M_1 、 M_2 (其状态转换矩阵及初态、终态信息分别保存在指定文件中)。（知识点难度系数 *）

10. 题目： 构造正规式 r^* （闭包运算）的 NFA 的程序实现

设计内容及要求：对给定的正规式 r，已知其 NFA 为 M(其状态转换矩阵及初态、终态信息保存在指定文件中)。构造一程序，由此程序构造正规式 r^* （闭包运算）的 NFA(将其状态转换矩阵及初态、终态信息保存在指定文件中)。（知识点难度系数 *）

11. 题目： 基于语法制导构造正规式的 NFA

设计内容及要求：首先构造一个语法分析程序，实现对任意正规式的语法分析。语法分析方法采用自下而上的分析方法（如算符优先分析，或 LR 分析）。

在此语法分析器的基础上，按照语法制导的思想，增加构造 NFA 的功能。生成的 NFA 将其状态转换矩阵及初态、终态信息保存在指定文件中。进一步实现把 NFA 确定化为 DFA 的算法(其状态转换矩阵及初态、终态信息保存在指定文件中)。(知识点难度系数 ***)

12. 题目： DFA M 状态最少化的程序实现

设计内容及要求：构造一程序，实现：将给定的 DFA M(其状态转换矩阵及初态、终态信息保存在指定文件中)的有限状态集 S 划分成若干互不相交的子集，使得：任何不同的两个子集中的状态都是可区别的，而同一子集中的任何两个状态都是等价的(要利用 Ia 函数，但并不需要构造 ϵ -CLOSURE 函数，因这是 DFA)。输出化简后的 DFA M' (其状态转换矩阵及初态、终态信息保存在指定文件中)。(知识点难度系数 **)

13. 题目： 把 NFA 确定化为 DFA 的算法实现

设计内容及要求：构造一程序，实现：将给定的 NFA M(其状态转换矩阵及初态、终态信息保存在指定文件中)，确定化为 DFA M'。(要先实现 ϵ -CLOSURE 函数和 Ia 函数)。输出 DFA M' (其状态转换矩阵及初态、终态信息保存在指定文件中)。(知识点难度系数 **)

14. 题目： 基于贪心算法的 DFA 的程序实现

设计内容及要求：采用贪心算法实现教材 P. 62 表 3.5 的 DFA，要求从输入串中匹配最长的子串。输出所有识别出的符号串及其词形。(知识点难度系数 *)

15. 题目： 根据句型的推导构造其语法分析树的程序实现

设计内容及要求：构造一程序，实现：接受用户任意输入的一个句型的推导序列，生成该句型的语法分析树并显示输出。程序应具有通用性，语法分析树的节点在屏幕上的分布要合理、美观。(知识点难度系数 *)

16. 题目： 从语法分析树构造句型所有的推导的程序实现

设计内容及要求：构造一程序，实现：接受用户任意输入的一个句型的语法分析树(其表示存于指定文件中)，生成该语法分析树中包含的该句型的所有推导(显示输出)。(知识点难度系数 *)

17. 题目： 递归下降分析程序的实现

设计内容及要求： (知识点难度系数 *)

对文法 $G: E \rightarrow E+T \mid T$ 构造出 G 的递归下降分析程序。程序显示输出
 $T \rightarrow T * F \mid F$ 匹配过程(即自上而下生成语法分析树的步骤，
 $F \rightarrow (E) \mid i$ 输出各匹配产生式序号即可)。

18. 题目： 集合 FIRST(X) 构造算法的程序实现

设计内容及要求：构造一程序，实现教材 P. 78 的 FIRST(X) 集合的构造算法。对任一给定的文法 G，程序输出所有非终结符 P 的 FIRST(P)。(知识点难度系数 *)

19. 题目： 集合 FOLLOW(A) 构造算法的程序实现

设计内容及要求：首先，构造一程序，实现教材 P. 78 的 FIRST(X) 集合的构造算法。对任一给定的文法 G，程序输出所有非终结符 P 的 FIRST(P)。在此基础上，构造一程序，实现教材 P. 79 的 FOLLOW(A) 集合的构造算法。对任一给定的文法 G，程序输出所有非终结符 A 的 FOLLOW (A)。(知识点难度系数 *)

20. 题目： 预测分析表构造算法的程序实现

设计内容及要求：对于给定的一个 LL(1) 文法，假定所有非终结符号 P 的集合 FIRST(P) 和集合 FOLLOW(P) 都已知，构造其预测分析表（实现教材 P. 79 给出的预测分析表构造算法）。对教材 P. 79 给出的例 4.7 构造出预测分析表。程序显示输出预测分析表或输出到指定文件中。(知识点难度系数 *)

21. 题目： 预测分析表自动构造程序的实现

设计内容及要求：对于任意输入的一个 LL(1) 文法，构造其预测分析表。要求：首先实现集合 FIRST(X) 构造算法和集合 FOLLOW(A) 构造算法，再实现教材 P. 79 给出的预测分析表构造算法。程序显示输出预测分析表或输出到指定文件中。(知识点难度系数 **)

22. 题目： 预测分析程序的实现

设计内容及要求：(知识点难度系数 *)

对文法 G: $E \rightarrow E+T \mid T$ 按教材 P. 76 表 4.1 构造出 G 的预测分析程序，
 $T \rightarrow T * F \mid F$ 程序显示输出如 P. 78 那样的匹配过程。
 $F \rightarrow (E) \mid i$

23. 题目： 集合 FIRSTVT(P) 构造算法的程序实现

设计内容及要求：构造一程序，实现教材 P. 91 的 FIRSTVT(P) 集合的构造算法。对任一给定的算符文法 G，程序输出所有非终结符 P 的 FIRSTVT(P)。(知识点难度系数 *)

24. 题目： 集合 LASTVT(P) 构造算法的程序实现

设计内容及要求：构造一程序，实现教材 P. 91 的 LASTVT(P) 集合的构造算法。对任一给定的算符文法 G，程序输出所有非终结符 P 的 LASTVT(P)。(知识点难度系数 *)

25. 题目： 算符优先分析算法的程序实现

设计内容及要求：（知识点难度系数 *）

对文法 $G: E \rightarrow E+T \mid T$ 采用教材 P. 90 表 5. 1，实现 P. 93 描述的算符优先

$T \rightarrow T * F \mid F$ 分析算法。程序显示输出“移进-归约”的步骤。

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

$P \rightarrow (E) \mid i$

26. 题目： 带出错处理的算符优先分析算法的程序实现

设计内容及要求：（知识点难度系数 *）

对文法 $G: E \rightarrow E+T \mid T$ 采用教材 P. 98 表 5. 3，实现 P. 93 描述的算符优先

$T \rightarrow T * F \mid F$ 分析算法。程序显示输出“移进-归约”的步骤。

$F \rightarrow (E) \mid i$ 要编制各出错处理子程序。

27. 题目： 优先表构造算法的程序实现

设计内容及要求： 实现教材 P. 92 优先表构造算法。对任一给定的算符优先文法 G ，假定所有非终结符 P 的 $FIRSTVT(P)$ 、 $LASTVT(P)$ 均已知。以教材 P. 90 例 5. 4 文法为例，程序生成表 5. 1 优先表。（知识点难度系数 *）

28. 题目： 优先表自动构造程序的实现

设计内容及要求：对任一给定的算符优先文法 G ，构造其优先表。要求：首先实现对于非终结符 P 的 $FIRSTVT(P)$ 构造算法和 $LASTVT(P)$ 构造算法，再实现教材 P. 92 给出的优先表构造算法。以教材 P. 90 例 5. 4 文法为例，程序生成表 5. 1 优先表。（知识点难度系数 **）

29. 题目： 算符优先分析自动构造程序的实现

设计内容及要求：对任一给定的算符优先文法 G ，构造其优先表。要求：首先实现对于非终结符 P 的 $FIRSTVT(P)$ 构造算法和 $LASTVT(P)$ 构造算法，再实现教材 P. 92 给出的优先表构造算法。再实现 P. 93 描述的算符优先分析算法。程序显示输出“移进-归约”的步骤。可以以教材 P. 90 例 5. 4 文法为例，描述程序实现的过程（知识点难度系数 **）

30. 题目： 优先函数构造程序的实现

设计内容及要求：构造一程序，实现：根据优先关系表构造优先函数。输入的优先表假定保存在指定文件中，构造出的优先函数可显示输出，或输出到指定文件中。（知识点难度系数 *）

31. 题目： 消除左递归算法的程序实现

设计内容及要求：构造一程序，实现教材 P. 70 消除左递归算法。对于用户任意输入的文法 G ，输出一个无左递归的等价文法，可显示输出，或输出到指定文件中。（**知识点难度系数 ***）

32. 题目： 消除回溯算法的程序实现

设计内容及要求：构造一程序，实现：消除文法每一条产生式候选式的公共左因子。对于用户任意输入的文法 G ，输出一个无回溯的等价文法，可显示输出，或输出到指定文件中。（**知识点难度系数 ***）

33. 题目： LR 分析器总控程序的实现

设计内容及要求：对 P. 101 中的文法，按图 5. 5LR 分析表构造 LR 分析器。要求程序按 P. 102 例 5. 7 那样，对于输入串 $i*i+i$ ，输出 LR 分析器的工作过程。（**知识点难度系数 ***）

34. 题目： 识别文法活前缀的 NFA 构造程序的实现

设计内容及要求：构造一程序，对任意给定的文法 G ：①构造并输出 G 的所有 LR(0) 项目；②用这些 LR(0) 项目构造并输出识别文法活前缀的 NFA (输出其状态转换矩阵)。（**知识点难度系数 ****）

35. 题目： LR(0) 项目集规范族构造程序的实现

设计内容及要求：构造一程序，对任意给定的文法 G ，构造识别文法活前缀的 DFA，输出 DFA 的状态转化矩阵及 LR(0) 项目集规范族。要求按教材 P. 107 所给的 $ITEMSETS(G')$ 构造，要实现 $CLOSURE(I)$ 、 $GO(I, X)$ 函数。按 P. 105 例 5. 8 给出测试结果，可输出到指定文件中。（**知识点难度系数 ****）

36. 题目： LR(0) 分析表构造算法的程序实现

设计内容及要求： 构造一程序，实现 LR(0) 分析表构造算法。根据教材 P. 106 图 5. 7 的识别文法活前缀的 DFA，输出如 P. 109 表 5. 4 的 LR(0) 分析表，可输出到指定文件中。（**知识点难度系数 ***）

37. 题目： LR(0) 分析器自动构造程序的实现

设计内容及要求：对任意给定的文法 G ，构造识别文法活前缀的 DFA，输出 DFA 的状态转化矩阵及 LR(0) 项目集规范族；实现 LR(0) 分析表构造算法；实现 LR 分析器总控程序。程序输出一个完整的 LR(0) 分析器源程序，可输出到指定文件中。（**知识点难度系数 ****）

38. 题目： SLR(1) 分析表构造算法的程序实现

设计内容及要求：构造一程序，实现 SLR(1) 分析表构造算法 (假定所给文法识别文法活前缀的 DFA、LR(0) 项目集族、所有非终结符 FOLLOW 集合均已构造

出来了)。根据教材 P. 111 例 5.11 文法为输入, 构造其 SLR(1) 分析表。(知识点难度系数 *)

39. 题目: LR(1) 项目集规范族构造程序的实现

设计内容及要求: 构造一程序, 对任意给定的文法 G 构造 LR(1) 项目集规范族 (按教材 P. 115 所述方法构造, 要实现 $CLOSURE(I)$ 、 $GO(I, X)$, 集合 FIRST 的构造方法参见教材 P. 78)。(知识点难度系数 **)

40. 题目: LR(1) 分析表构造算法的程序实现

设计内容及要求: 构造一程序, 实现 LR(1) 分析表构造算法 (假定所给文法识别文法活前缀的 DFA、LR(1) 项目集族已构造出来了)。根据教材 P. 116 图 5.10 的 LR(1) 项目集族和 GO 函数为输入, 构造并输出其 LR(1) 分析表 5.5。(知识点难度系数 *)

41. 题目: LR(1) 分析表自动构造程序的实现

设计内容及要求: 对任意给定的文法 G 构造 LR(1) 项目集规范族 (按教材 P. 115 所述方法构造, 要求实现 $CLOSURE(I)$ 、 $GO(I, X)$ 、FIRST (集合 FIRST 的构造方法参见教材 P. 78); 然后实现 LR(1) 分析表构造算法。以教材 P. 115 例 5.13 为输入, 构造并输出其 LR(1) 分析表 5.5。(知识点难度系数 **)

42. 题目: LALR(1) 项目集规范族构造程序的实现

设计内容及要求: 假设对于给定文法, 识别文法活前缀的 DFA、LR(1) 项目集族已构造出来了。构造一程序, 检查两个 LR(1) 项目集是否为同心集 (可任意输入), 若是, 则输出合并后的同心集, 并检查合并后的集合是否含有冲突项目 (指出存在何种冲突), 输出合并同心集后的识别文法活前缀的 DFA, 及 LALR(1) 项目集规范族。(知识点难度系数 *)

43. 题目: LALR(1) 分析表自动构造程序的实现

设计内容及要求: 对任意给定的文法 G 构造 LR(1) 项目集规范族 (按教材 P. 115 所述方法构造, 要求实现 $CLOSURE(I)$ 、 $GO(I, X)$ 、FIRST (集合 FIRST 的构造方法参见教材 P. 78); 然后构造 LALR(1) 项目集规范族; 再实现 LALR(1) 分析表构造算法。以教材 P. 115 例 5.13 为输入, 构造并输出其 LALR(1) 分析表 5.6。

(知识点难度系数 **)

44. 题目: 带出错处理的 LR 分析器总控程序的实现

设计内容及要求: 按教材 P. 127 表 5.9 分析表构造一 LR 分析器, 输出语法分析过程 (如 P. 128 所述), 要构造各出错处理子程序。(知识点难度系数 *)

45. 题目: 算术表达式从中缀式翻译成后缀式的程序实现

设计内容及要求：设计一个语法制导翻译器，将算术表达式从中缀式翻译成后缀式。要求：先确定一个定义算术表达式的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的算术表达式，程序将其转换成后缀式输出。（**知识点难度系数 *****）

46. 题目：将算术表达式转换成抽象语法树的程序实现

设计内容及要求：设计一个语法制导翻译器，将算术表达式转换成抽象语法树。要求：先确定一个定义算术表达式的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的算术表达式，程序将其转换成抽象语法树输出(可按一定格式输出到指定文件中)。（**知识点难度系数 *****）

47. 题目：将算术表达式转换成 DAG 的程序实现

设计内容及要求：设计一个语法制导翻译器，将算术表达式转换成 DAG。要求：先确定一个定义算术表达式的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的算术表达式，程序将其转换成 DAG 输出(可按一定格式输出到指定文件中)。（**知识点难度系数 *****）

48. 题目：将算术表达式转换成三元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将算术表达式翻译成三元式。要求：先确定一个定义算术表达式的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的算术表达式，程序将其转换成三元式输出(可按一定格式输出到指定文件中)。（**知识点难度系数 *****）

49. 题目：将算术表达式转换成间接三元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将算术表达式翻译成间接三元式。要求：先确定一个定义算术表达式的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的算术表达式，程序将其转换成间接三元式输出(可按一定格式输出到指定文件中)。（**知识点难度系数 *****）

50. 题目：将算术表达式转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将算术表达式翻译成四元式。要求：先确定一个定义算术表达式的文法，为其设计一个语法分析程序，为每条

产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的算术表达式，程序将其转换成四元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

51. 题目：将布尔表达式转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将布尔表达式翻译成四元式。要求：先确定一个定义布尔表达式的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的布尔表达式，程序将其转换成四元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

52. 题目：将条件语句转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将条件语句翻译成四元式。要求：先确定一个定义条件语句的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的条件语句，程序将其转换成四元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

53. 题目：将 WHILE 语句转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将 WHILE 语句翻译成四元式。要求：先确定一个定义 WHILE 语句的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的 WHILE 语句，程序将其转换成四元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

54. 题目：将 FOR 语句转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将 FOR 语句翻译成四元式。要求：先确定一个定义 FOR 语句的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的 FOR 语句，程序将其转换成四元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

55. 题目：将 DO-WHILE 语句转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将 DO-WHILE 语句翻译成四元式。要求：先确定一个定义 DO-WHILE 语句的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的 DO-WHILE 语句，程序将其转换成四

元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

56. 题目：将 SWITCH 语句转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将 SWITCH 语句翻译成四元式。要求：先确定一个定义 SWITCH 语句的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的 SWITCH 语句，程序将其转换成四元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

57. 题目：将包含数组引用的赋值语句转换成四元式的程序实现

设计内容及要求：设计一个语法制导翻译器，将包含数组引用的赋值语句翻译成四元式。要求：先确定一个定义包含数组引用的赋值语句的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一个正确的包含数组引用的赋值语句，程序将其转换成四元式输出(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

58. 题目：嵌套过程中的说明语句翻译的程序实现

设计内容及要求：设计一个语法制导翻译器，完成对说明语句的翻译，即构造每个过程的符号表，填写所有名字在符号表中的有关信息。要求：先确定一个定义允许嵌套过程的说明语句的文法，为其设计一个语法分析程序，为每条产生式配备一个语义子程序，按照一遍扫描的语法制导翻译方法，实现翻译程序。对用户输入的任意一组正确的说明语句，程序将输出相应的符号表(可按一定格式输出到指定文件中)。(知识点难度系数 ***)

59. 题目：基本块划分算法的程序实现

设计内容及要求：根据基本块划分算法，构造一个基本块划分程序，实现：对于任意输入的一个四元式程序，将其划分为基本块，输出各基本块，并输出程序流程图。以 P. 279 例 10.1 为输入，输出 P. 281 图 10.8. (知识点难度系数 **)

60. 题目：将基本块转换成 DAG 的算法的程序实现

设计内容及要求：根据将基本块转换成 DAG 的算法，实现：对于任意输入的一个基本块（四元式程序），将其转换成 DAG 并输出(可按一定格式输出到指定文件中)。以 P. 283 例 10.4 为输入，输出 P. 284 图 10.10 构造过程。(知识点难度系数 **)

61. 题目：由 DAG 重构基本块的程序实现

设计内容及要求：按照 DAG 节点构造顺序，重构基本块四元式代码。输入的 DAG 按一定格式存于指定文件中，输出的基本块四元式代码按一定格式输出到指定文件中。以 P. 284 图 10.10 为输入，输出基本块四元式代码。（**知识点难度系数 ****）

62. 题目：局部优化程序的实现

设计内容及要求：根据将基本块转换成 DAG 的算法，实现：对于任意输入的一个基本块（四元式程序），将其转换成 DAG；然后按照 DAG 节点构造顺序，重构基本块四元式代码。以 P. 283 例 10.4 为输入，完成并输出局部优化。（**知识点难度系数 *****）

（待续）

。。。。。。（大家也可以自行设计一个设计题目）