

# 编译原理课程设计



# 编译原理课程设计

一、课程介绍

二、课程设计任务

三、课程设计进度安排计划

# 一、编译原理课程设计课程介绍

课程性质：必修课

必修：每个同学必须修读并完成相应的设计要求。

## 二、课程设计任务

- 1.按课程设计任务书的具体要求进行完成。
- 2.课程设计实验报告书要按 《课程设计实验报告书 版头及正文格式》 文件中的格式书写。
- 3.完成方法：每个学生独立完成。
- 4.上交内容：课程设计源程序、课程设计实验报告书、系统使用说明书、课程设计系统的可执行程序、测试文本及测试源程序、测试报告、课程设计报告讲解视频
- 5.所设计的课程设计系统应该是一个窗口式图形界面的应用程序而且系统实现所采用的编程语言只能是C/C++语言。
- 6.课程设计进度安排说明。

# 实验回顾

第一个

# 实验2正则表达式转换为DFA图

## 一、实验内容:

设计一个应用软件, 以实现将正则表达式-->NFA--->DFA-->DFA最小化-->词法分析程序(选做内容)

## 二、必做实验内容及要求:

(1) 正则表达式应该可以支持命名操作, 运算符有: 转义符号(\)、连接、选择(|)、闭包(\*)、正闭包(+)、[], 可选(?), 括号()

(2) 要提供一个源程序编辑界面, 让用户输入一行(一个)或多行(多个)正则表达式(可保存、打开正则表达式文件)

(3) 需要提供窗口以便用户可以查看转换得到的NFA(用状态转换表呈现即可)

(4) 需要提供窗口以便用户可以查看转换得到的DFA(用状态转换表呈现即可)

(5) 需要提供窗口以便用户可以查看转换得到的最小化DFA(用状态转换表呈现即可)

(6) 要求应用程序为窗口式图形界面

(7) 书写完善的软件文档

## 三、选做实验内容及要求:

将最小化得到DFA图转换得到的词法分析源程序(该分析程序需要用C/C++语言描述, 而且只能采用讲稿中的转换方法一或转换方法二来生成源程序。), 系统需提供窗口以便用户可以查看转换得到的词法分析源程序

## 实验2

digit=[0-9]

letter=[A-Za-z]

\_num100=(\+|-)?digit+

\_ID101=letter(letter|digit)\*

\_specail2005=\+|-|\\*|/|=|<|<=|<<

说明：

(1) 通过命名中加下划线()来表示该正则表达式需要生成 DFA 图。

(2) 命名中的名字后的数值为对应单词的编码

(3) 命名中的名字中数值后加上 S 表示后面有多个单词，但对应的单词编码从这个数值开始编码。

(4) 由于整数中的正号 (+)，在系统已经被用作正比闭包运算符，所以需要通过引入转义符号 \ 来区分。同理，如算术运算符 (\*), 在系统已经被用作闭包运算符，所以也需要通过引入转义符号 \ 来区分。



## 实验2

1.输入一行(一个)或多行(多个)正则表达式:

letter=[A-Za-z]

digit=[0-9]

\_ID101=letter(letter|digit)\*

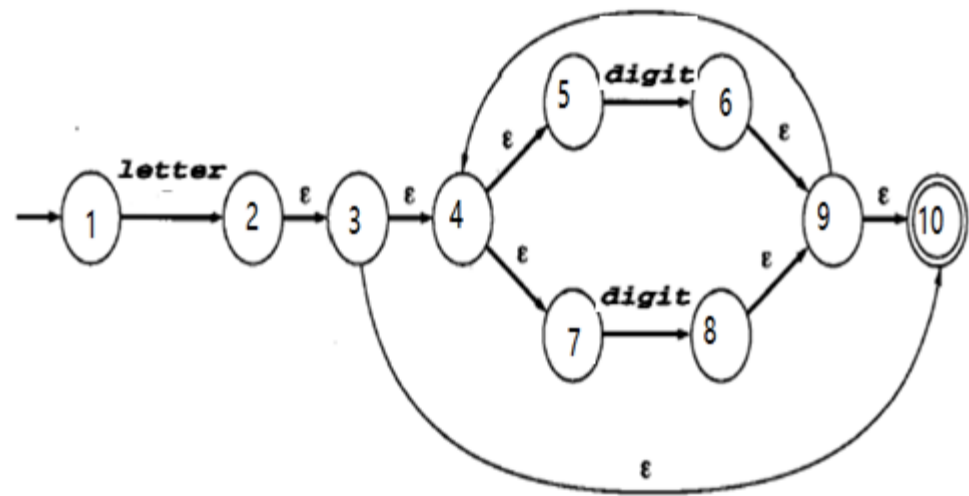
注意:

(1) 通过命名中加下划线(\_)来表示该正则表达式需要生成DFA图。

(2) 命名中的名字后的数值为对应单词的编码

## 2.正则表达式→NFA

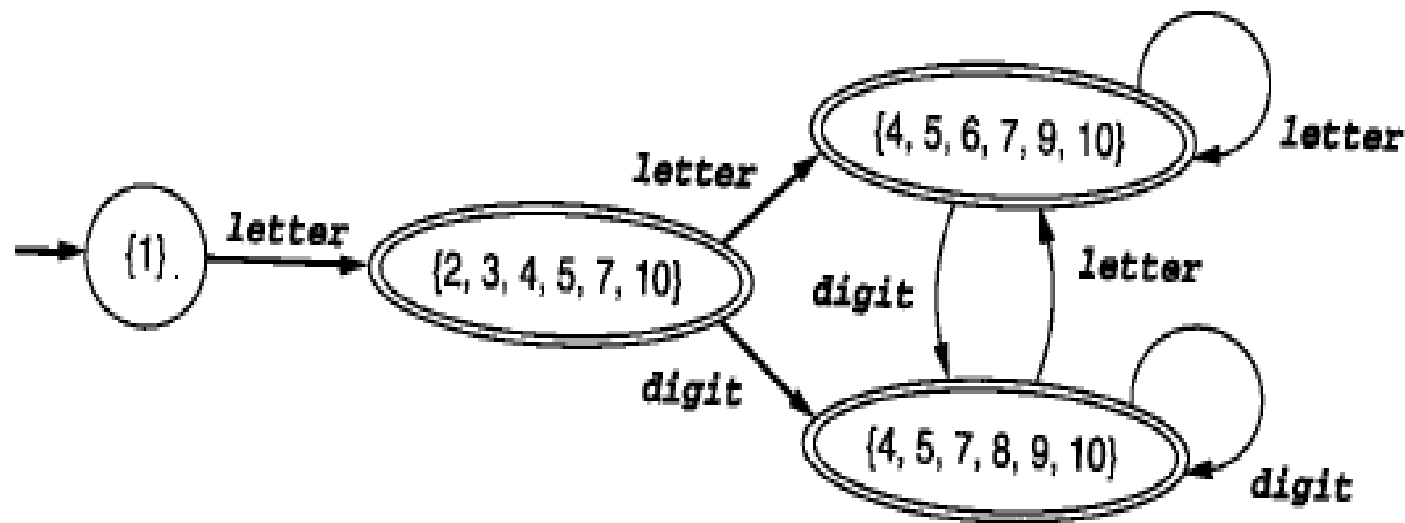
输出： 如下图的NFA或该图对应的状态转换表



		letter	digit	#
—	1	2		
	2			3
	3			4, 10
	4			5, 7
	5	6		
	6			9
	7		8	
	8			9
	9			4, 10
+	10			

### 3.NFA→DFA

输出：如下图的DFA或对应的状态转换表(如下表)

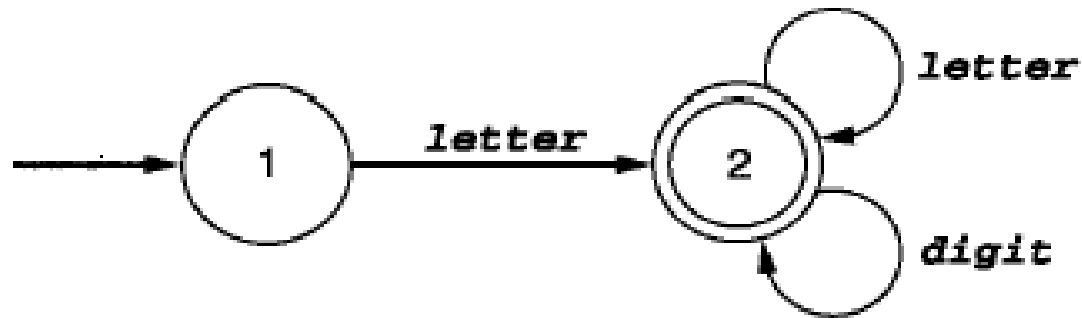


字符 \ 状态集合	letter	digit
{ 1 }	{ 2, 3, 4, 5, 7, 10 }	
{ 2, 3, 4, 5, 7, 10 }	{ 6, 9, 4, 7, 10, 5 }	{ 8, 9, 4, 7, 5, 10 }
{ 6, 9, 4, 7, 10, 5 }	{ 6, 9, 4, 7, 10, 5 }	{ 8, 9, 4, 7, 5, 10 }
{ 8, 9, 4, 7, 5, 10 }	{ 6, 9, 4, 7, 10, 5 }	{ 8, 9, 4, 7, 5, 10 }

—  
+  
+  
+

## 4.DFA最小化

输出：如下图的最小化DFA或对应的状态转换表



—

+

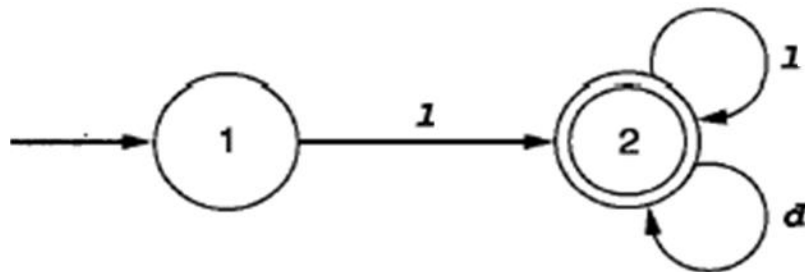
	letter	digit
1	2	
2	2	2

## 5.DFA→词法分析程序【注意：实验要求生成的分析程序是C/C++语言的表达方法】

输出：生成对应的词法分析程序(方法一)。

```
int Judgechar(char ch)
{ if (isalpha(ch)) return 1; else if (isdigit(ch)) return 0;
  return -1; }
char GetNext(){ return buffer[pos++]; }

void GetToken()
{
  if (Judgechar(ch)==1)
  { ch=GetNext();
    while ((Judgechar(ch)==1) || (Judgechar(ch)==0))
      ch=GetNext();
    cout<<"ID";
  }
  else   cout<<"Error";
}
```



## 输出：生成对应的词法分析程序(方法二)

```
int Judgechar(char ch)
```

```
{ if (isalpha(ch)) return 1; else if (isdigit(ch)) return 0;  
  return -1; }
```

```
char GetNext(){ return buffer[pos++]; }
```

```
void GetToken()
```

```
{ int state=1; bool done=false;
```

```
  while (!done)
```

```
  { switch(state)
```

```
    { case 1: if (Judgechar(ch)==1)
```

```
      { state=2; ch=GetNext(); }
```

```
    else { cout<<"Error!";return; }
```

```
    case 2: if ((Judgechar(ch)==1) || (Judgechar(ch)==0))
```

```
      ch=GetNext();
```

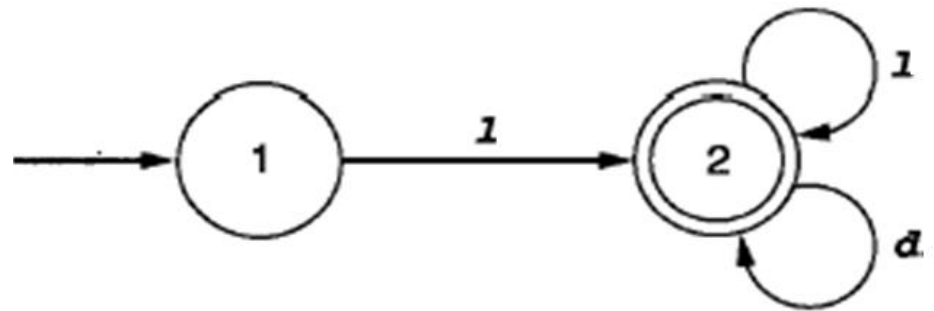
```
      else done=true;
```

```
    }
```

```
  }
```

```
  cout<<"ID";
```

```
}
```



# 课程设计的第一项任务

- 设计一个应用软件，以实现输入某一高级程序设计语言的所有单词的正则表达式-->NFA--->DFA-->DFA最小化-->词法分析程序。

# 课程设计的第一项任务要求

(1) 以文本文件的方式输入某一高级程序设计语言的所有单词对应的正则表达式，系统需要提供一个操作界面，让用户打开某一语言的所有单词对应正则表达式文本文件，该文本文件的具体格式可根据自己实际的需要进行定义。

(2) 正则表达式应该可以支持命名操作，运算符有：转义符号(\)、连接、选择(|)、闭包(\*)、正闭包(+)、[]、可选(?)、括号()

(3) 要提供一个源程序编辑界面，让用户输入一行（一个）或多行（多个）正则表达式（可保存、打开正则表达式文件）

(4) 需要提供窗口以便用户可以查看转换得到的NFA（用状态转换表呈现即可）

(5) 需要提供窗口以便用户可以查看转换得到的DFA（用状态转换表呈现即可）

(6) 需要提供窗口以便用户可以查看转换得到的最小化DFA（用状态转换表呈现即可）

(7) 将最小化得到DFA图转换得到的词法分析源程序（该分析程序需要用C/C++语言描述，而且只能采用讲稿中的转换方法一或转换方法二来生成源程序），系统需提供窗口以便用户可以查看转换得到的词法分析源程序

(8) 对要求(7)得到的源程序进行编译生成一个可执行程序，并以该高级程序设计语言的一个源程序进行测试，输出该该源程序的单词编码。需要提供窗口以便用户可以查看该单词编码。

(9) 对系统进行测试：(A) 先以TINY语言的所有单词的正则表达式作为文本来测试，生成一个TINY语言的词法分析源程序；(B) 接着对这个词法分析源程序利用C/C++编译器进行编译，并生成可执行程序；(C) 以sample.tny来测试，输出该TINY语言源程序的单词编码文件sample.lex。

(10) 要求应用程序为窗口式图形界面

(11) 书写完善的软件文档



# 第一项任务要求的测试数据

- Tiny语言

# TINY 语言的单词分类

(1). 关键词:

if then else end repeat until read write 不区分大小写

(2). 专用符号: + - \* / % ^ < <> <= >= > = { } ; :=

(3). identifier和*number*, 则定义:

identifier: 字母开头, 后面可跟若干个字母或数字符号, 且不区分大小写

*number*: 数字符号开头, 后面可跟若干个数字

也可以抽象地表示为:

identifier = letter(letter|digit)\*

*number* = digit+

letter = [a-zA-Z]

digit = [0-9]

(4). 空格由空白、换行符和制表符组成。

(5). 用表示{ }注释, 注释可以放在任何空白出现的位置(即注释不能放在标记内)上, 且可以超过一行。注释不能嵌套。

# 1.TINY语言词法测试结果

- 1.打开包含有所有TINY语言单词的正则表达式的文本文件
- 2.生成对应的NFA图，需要界面可以查看该图
- 3.生成对应的DFA图，需要界面可以查看该图
- 4.生成对应的最小DFA图，需要界面可以查看该图
- 5.把最小DFA图转换为对应的词法分析源程序(C++源程序)，该源程序需要在C/C++编译器环境下成功编译，需要界面可以查看词法分析源程序
- 6.利用这个词法分析源程序进行词法分析，并生成单词编码文件，需要界面可以查看单词编码文件内容。

第二个

# 实验4 LR(1)分析生成器

## 一、实验内容:

设计一个应用软件, 以实现LR(1)分析生成器。

## 二、实验要求:

### 1.必做功能:

(1)要提供一个文法输入编辑界面, 让用户输入文法规则 (可保存、打开存有文法规则的文件)

(2)求出文法各非终结符号的first集合与follow集合, 并提供窗口以便用户可以查看这些集合结果。【可以采用表格的形式呈现】

(3)需要提供窗口以便用户可以查看文法对应的LR(0)DFA图。(可以用画图的方式呈现, 也可用表格方式呈现该图点与边数据)

(4)判断该文法是否为SLR(1)文法。(应提供窗口呈现判断的结果, 如果不是SLR(1)文法, 需要在窗口中显示其原因)

(5)需要提供窗口以便用户可以查看文法对应的LR(1)DFA图。(可以用画图的方式呈现, 也可用表格方式呈现该图点与边数据)

(6)需要提供窗口以便用户可以查看文法对应的LR(1)分析表。【LR(1)分析表采用表格的形式呈现】

(7)应用程序应为窗口式图形界面。

### 2.选做功能。

(1)需要提供窗口以便用户输入需要分析的句子。

(2)需要提供窗口以便用户查看使用LR(1)分析该句子的过程。【可以使用表格的形式逐行显示分析过程】

# 课程设计的第二项任务要求

- 设计一个应用软件，以实现输入某一高级程序设计语言的所有语法对应的BNF文法-->求出文法的每个非终结符号的First集合和Follow集合-->生成LR(0)DFA图-->判断是否为SLR(1)文法-->生成LR(1)DFA图-->生成LR(1)分析表-->以LR(1)语法分析方法进行源程序各语句的语法分析并显示其分析过程→生成相应的语法树并显示。

# 课程设计的第二项任务

具体要求:

- (1) 以文本文件的方式输入某一高级程序设计语言的所有语法对应的BNF文法, 因此系统需要提供一个操作界面, 让用户打开某一语言的所有语法对应的BNF文法的文本文件, 该文本文件的具体格式可根据自己实际的需要进行定义。
- (2) 求出文法的每个非终结符号的First集合和Follow集合, 并需要提供窗口以便用户可以查看该结果 (可用两张表格的形式分别进行呈现)
- (3) 需要提供窗口以便用户可以查看文法对应的LR(0)DFA图。 (可以用画图的方式呈现, 也可用表格方式呈现该图点与边数据)
- (4) 判断该文法是否为SLR(1)文法。 (应提供窗口呈现判断的结果, 如果不是SLR (1) 文法, 需要在窗口中显示其原因)
- (5) 需要提供窗口以便用户可以查看文法对应的LR(1)DFA图。 (可以用画图的方式呈现, 也可用表格方式呈现该图点与边数据)
- (6) 需要提供窗口以便用户可以查看文法对应的LR(1)分析表。 (LR(1)分析表采用表格的形式呈现)
- (7) 打开一个与上述所输入BNF文法对应的程序设计语言的源程序 (即任务一的输出文件), 并采用LR(1)语法分析方法对这个源程序中的每个语句进行语法分析, 需要提供窗口以便用户可以查看对应的语法分析过程。 【可以使用表格的形式逐行显示分析过程】
- (8) 在 (7) 的分析过程中同时生成相应的语法树, 需要提供窗口以便用户可以查看对应的语法分析过程 (语法树需要采用树状形式进行呈现)。 【每个语句的语法树结构可根据实际的需要进行定义。】
- (9) 以TINY语言的所有语法以及第一项任务的测试结果sample.lex作为测试, 并生成对应的语法树并呈现出来。
- (10) 要求应用程序为窗口式图形界面
- (11) 书写完善的软件文档
- (12) 选做内容: 可以生成某种中间代码 【具体的中间代码形式可以自定】。

# 第二项任务要求的测试数据

- Tiny语言



## 2.TINY语言的所有语法

program->stmt-sequence

stmt-sequence-> stmt-sequence ;statement | statement

statement->if-stmt | repeat-stmt | assign-stmt | read-stmt | write-stmt

if-stmt->if exp then stmt-sequence end | if exp then stmt-sequence  
else stmt-sequence end

repeat-stmt->repeat stmt-sequence until exp

assign-stmt->identifier := exp

read-stmt->read identifier

write-stmt->write exp

exp->simple-exp comparison-op simple-exp | simple-exp

comparison-op-> < | > | = | >= | <= | <>

simple-exp->simple-exp addop term | term

addop-> + | -

term->term mulop factor | factor

mulop-> \* | / | % | ^

factor->(exp) | number | identifier

# 课程设计的第三项任务

前两项任务都完成后...

- mini-c语言作为测试
- (1) 以mini-c的词法进行测试，并以至少一个mini-c源程序进行词法分析的测试。
- (2) 以mini-c的语法进行测试，并以测试步骤(1)的源程序所生成的单词编码文件进行语法分析，生成对应的语法树。

### 第三部分 课程设计进度安排

时间	进度安排
第1周	任课老师进行课程设计介绍
第1周~第5周	完成 <b>第一项课程设计任务</b> 的软件设计及实验报告书的书写要求，并以Tiny语言的词法进行测试及书写测试报告
第6周~第10周	完成 <b>第二项课程设计任务</b> 的软件设计及实验报告书的书写要求，并以Tiny语言的语法进行测试及书写测试报告
第11周~第13周	<ol style="list-style-type: none"><li>1.以Mini-C进行整个课程设计系统功能的测试（包含词法和语法的测试），并书写测试报告</li><li>2.完成最后课程设计实验报告的整合和自评</li><li>3.完成系统使用说明书的书写</li><li>4.<b>课程设计报告讲解视频的录制（15分钟）</b></li></ol>
第14周	完成如下内容： <ol style="list-style-type: none"><li>1.整理好课程设计源程序、测试文本及测试源程序、测试报告、课程设计的可执行程序、使用说明书、课程设计报告书和讲解视频。</li><li>2.<b>第14周周五晚上12点之前</b>提交上述所有资料，在砺儒云课堂进行提交。</li></ol>
第15~17周	<ol style="list-style-type: none"><li>1.抽查、评分。</li><li>2.抽查到的同学需到学院实验室解释相关的工作。</li></ol>

# 编译原理课程设计讲解视频录制的注意事项

- 课程设计的设计过程的讲解视频录制。时间为15分钟，但如果  
有系统演示的话，只能有2分钟的系统演示，但你可以录制20  
分钟的视频，这样你就可以有7分钟的系统演示，但视频的时  
长最长为20 分钟。

- **注意事项：**

1.视频录制的开头和结尾都要有本人露面，开头就露面讲解课程设计的标题，汇报人的学号、姓名、年级和班级，后面就露面说课程设计讲解完毕。【最好是视频中有全程的小视窗露面】

2.需要先制作汇报讲解用的PPT讲稿才进行讲解视频的录制

3.为了方便在砺儒云课堂上提交，整个课程设计的讲解视频最大容量不超过100M。如果超过这个容量，请使用视频编辑软件进行压缩处理。

# 课程设计评分标准

主要根据学生所制作的前两项任务要求的软件系统设计、运行及测试情况、课程设计实验报告书的书写情况以及课程设计报告讲解视频的情况进行评分（**系统和文档各占45%，课程设计报告讲解视频占10%，且均需要达到及格标准以上**）。

评分标准主要依据如下：

**E<60:**在规定时间内上交实验源程序、实验报告书、测试数据与测试结果、测试报告、讲解视频，在**必做内容的任务功能已经全部完成的情况下**，各任务功能累计**未能完成超过3/5**。

**60<D<70:**在规定时间内上交所有的课程设计材料，在**必做内容的任务功能已经全部完成的情况下**，各任务功能累计**完成超过3/5**，但没能全部完成。讲解视频有完整的讲解思路，编程风格好，实验报告以及测试材料基本符合规范，设计思想基本清晰，界面基本符合要求。

**70<C<80:**在规定时间内上交所有的课程设计材料，并完成了各任务功能的全部内容（除选做内容外），讲解视频讲解思路符合要求，实验报告以及测试材料等符合规范，编程风格好，设计思想基本清晰，界面美观大方，使用方便。

**80<B<90:**在规定时间内上交所有的课程设计材料，并完成了各任务功能的全部内容以及选做内容中的部分要求，讲解视频讲解思路清晰，实验报告以及测试材料等规范清晰，编程风格好，设计思想清晰，界面美观大方，使用方便。

**90<A<100:**在规定时间内上交所有的课程设计材料，并完成了各任务功能的全部内容以及选做内容的全部内容，且功能完善，讲解视频讲解思路十分清晰，实验报告以及测试材料等十分规范清晰，设计思想十分清晰，编程风格好，界面美观大方，使用方便。