

词法分析实验题

1. 正规式转化为 NFA

【问题描述】正规表达式→NFA 问题的一种描述是：

编写一个程序，输入一个正规表达式，输出与该文法等价的有穷自动机。

【基本要求】设置 FA 初始状态 X，终态 Y，过程态用数字表示：0 1 2 3…………

【测试用例】

测试数据： (a|b)*abb

输出结果应为：X X - ~ ->3

Y

0 0-a->1

1 1-b->2

2 2-b->Y

3 3-~->0 3-a->3 3-b->3

2. NFA 确定化为 DFA

【问题描述】NFA→DFA 问题的一种描述是：

编写一个程序，输入一个非确定有穷自动机（NFA），输出与 NFA 等价的确定有穷自动机（DFA）。

【基本要求】设置 FA 初始状态 X，终态 Y，过程态用数字表示：0 1 2 3…………

【测试用例】

测试数据： X X-~->3

Y

0 0-a->1

1 1-b->2

2 2-b->Y

3 3-~->0 3-a->3 3-b->3

输出结果应为：X X-a->0

Y Y-a->0 Y-b->1

0 0-a->0 0-b->2

1 1-a->0 1-b->1

2 2-a->0 2-b->Y

3. DFA 的化简

【问题描述】DFA 化简问题的一种描述是：

编写一个程序，输入一个确定的有穷自动机（DFA），输出与 DFA 等价的最简的确定有穷自动机（DFA）。

【基本要求】设置 DFA 初始状态 X，终态 Y，过程态用数字表示：0 1 2 3…………

【测试用例】

测试数据： X X-a->0 X-b->1

Y Y-a->0 Y-b->1

0 0-a->0 0-b->2

1 1-a->0 1-b->1

2 2-a->0 2-b->Y

输出结果应为：X X-a->0 X-b->X

Y Y-a->0 Y-b->X
 0 0-a->0 0-b->2
 2 2-a->0 2-b->Y

4. 基于 DFA 的词法分析程序的实现

【问题描述】基于 DFA 的词法分析程序的实现问题的一种描述是：
 编写一个程序，输入一个确定的有穷自动机 (DFA)，输出词法分析程序代码，
 该代码功能与 DFA 识别功能等价。

【基本要求】设置 DFA 初始状态 X，终态 Y，过程态用数字表示：0 1 2 3……

【测试用例】

测试数据：

(1)DFA:
 X X-a->0 X-b->X
 Y Y-a->0 Y-b->X
 0 0-a->0 0-b->2
 2 2-a->0 2-b->Y
 (2) abb;ba

5. 词法分析程序的设计与实现

【问题描述】

请根据给定的文法设计并实现词法分析程序，从源程序中识别出单词，记录其单词类别和单词值，输入输出及处理要求如下：

(1) 数据结构和与语法分析程序的接口请自行定义；类别码需按下表格式统一定义；

(2) 为了方便进行自动评测，输入的被编译源文件统一命名为 **testfile.txt**；输出的结果文件统一命名为 **output.txt**，结果文件中每行按如下方式组织：

单词类别码 单词的字符/字符串形式(中间仅用一个空格间隔)

单词的类别码请统一按如下形式定义：

单词名称	类别码	单词名称	类别码	单词名称	类别码	单词名称	类别码
标识符	IDENF R	if	IFTK	-	MINU	=	ASSIGN
整型常量	INTCO N	else	ELSETK	*	MULT	;	SEMICN
字符常量	CHARC ON	do	DOTK	/	DIV	,	COMMA
字符串	STRCO N	while	WHILETK	<	LSS	(LPARENT
const	CONST TK	for	FORTK	<=	LEQ)	RPARENT
int	INTTK	scanf	SCANFTK	>	GRE	[LBRACK

char	CHART K	printf	PRINTF TK	>=	GEQ]	RBRACK
void	VOIDT K	return	RETURNTK	==	EQL	{	LBRACE
main	MAINT K	+	PLUS	!=	NEQ	}	RBRACE