

编译原理与技术实验报告

设计语法分析程序

- 一、实验环境
- 二、实验内容及要求
- 三、程序设计
 - 主要数据结构
 - 程序设计与算法
 - 错误处理
- 四、程序输入输出及执行结果
 - 输入形式
 - 输出形式
 - 执行结果

一、实验环境

- 实验平台：Windows10
- 编译环境：Visual Studio Code
- 编程语言：C++
- 版本号：1.0.0

二、实验内容及要求

编写语法分析程序，实现对算术表达式的语法分析。程序可以分析LL(1)文法，自行计算其FIRST集FOLLOW集，为给定文法自动构造预测分析表，并通过预测分析程序根据预测分析表进行逐步分析。根据同步信息的情况进行一定的错误处理。

三、程序设计

主要数据结构

```
multimap <string,string> grammar;    //文法
vector <string> nonter;               //非终结符
vector <string> ter;                 //终结符
map <string,set<string> > first;      //first集
map <string,set<string> > follow;     //follow集
pair <string,string> table[30][20];  //预测分析表
vector <string> stack;               //预测分析栈
```

由于输入文法时采取一个非终结符多个产生式时分别读入的策略，所以采用 `multimap` 容器来存储文法。终结符非终结符用 `vector` 来存储。对于FIRST集，FOLLOW集，由于集合与非终结符之间有对

应关系，所以采取了 `map` 结构。预测分析表采用 `pair` 数组的形式。考虑到方便输出，预测分析时用到的栈采用 `vector` 容器，而没有使用c++的 `stack` 结构。

程序设计与算法

```
class BuildSet
{
public:
    void buildfirstset();           //构建first集
    void ergodic(string a);        //递归构建first集
    void buildfollowset()          //构建follow集
}
```

构建FIRST集时首先将所有 `nonter` 中所有非终结符插入 `map<string,set> first` 中。然后调用 `ergodic` 函数，依次计算每一个非终结符的FIRST集。在 `ergodic` 函数中，传入的参数为非终结符。对于该符，首先遍历该符在文法中的所有产生式，使用 `while` 循环和一个 `pos` 指针遍历每个产生式的右部。如果字符是"@"或是终结符，则将其加入FIRST集，跳出循环，分析下一个产生式；如果是非终结符，则判断其FIRST集是否构造完成，没构造完则递归构造。构造完成后将其FIRST集所有非"@"元素加入本符的FIRST集。若含有"@"，则 `pos` 指针加一，继续遍历右部；不含空则跳出循环，继续处理下一个产生式。

构建FOLLOW集时，首先在初始符号的FOLLOW集中加入"\$"，然后用 `while` 循环和一个 `pos` 指针遍历每个产生式的右部。如果是终结符，则跳过；如果是非终结符，则进入循环将其后面的符号遍历，遇到终结符或FIRST集中不含有@的非终结符则结束这个遍历，并将后面字符FIRST集中所有元素加入当前非终结符的FOLLOW集。如果当前符号的后继符号均为非终结符且含有空，或无后继符号，则应将当前生成式左部的FOLLOW集加入当前的非终结符。此时可能遇到生成式左部的FOLLOW集未构建完全的情况，所以本程序采取的策略是在此时记录将左部加入当前非终结符的动作，写入 `multimap<string,string> tran` 中，在所有生成式都遍历完毕后处理。处理时首先消除自己给自己加入的情况，然后建立一个需要被加入FOLLOW集的符号集合进行判断。若一个符号只需要将它的FOLLOW集加到其他符号中而不需要被加入，则说明它的FOLLOW集已经构造完成，此时就可以完成加入FOLLOW集的动作并在 `tran` 中删除这个动作。此时又有新的非终结符的FOLLOW集构建完成。循环以上过程直到 `tran` 中记录的动作被全部删除，此时即构建完成全部非终结符的FOLLOW集。

```
class BuildTable
{
public:
    void input(string& str)         //读取文件输入文法
    void bt()                       //构建预测分析表
    void outputtable()             //输出预测分析表
}
```

输入部分将根据用户输入的文件名来读取文件，将生成式存入 `multimap <string,string> grammar` 中。同时将读到的终结符读入 `vector <string> ter` 中，非终结符读入 `vector <string> nonter` 中。构建预测分析表时，遍历每一个产生式，然后通过循环的方式计算每个产生式右部的FIRST集，遇到终结符循环停止；遇到非终结符就将其FIRST集作为该右部的FIRST集，若其中含有空符，则分析下一个字符，直至跳出循环或读到末尾。本程序采用一个 `pair` 数

组作为预测分析表的存储结构，每确定一个终结符在当前生成式的FIRST集中，就将该生成式的左部右部存入对应位置的 `pair` 中。考虑到错误处理的问题，遍历完成所有生成式后，分析表格中空的部分若该位置对应的终结符在对应非终结符的FOLLOW集中，则填入同步信息 `synch`。最后输出表格。

```
class Predict
{
public:
    void predict(string& str)          //进行非递归预测分析
    void error(int f1)                 //输出错误
}
```

预测分析过程采用 `do-while` 循环，设置分析栈，标记栈顶符号和向前指针。读入用户输入的字符串，加入 `$` 符号标记结尾。循环中如果栈顶符号为空，则弹出空符；如果栈顶符号为终结符，则判断是否和向前指针所指向的符号相同，若相同则弹栈，同时向前指针前进一位，不相同则说明出错，此时需要弹出栈顶符号，向前指针不发生变化；若栈顶符号为非终结符，则确定其在分析表中的位置，找到相应的生成式，将当前栈顶符号弹出，然后将生成式右部倒序放入栈中；若没找到生成式而相应位置为同步信息，则弹出栈；若表中该项为空白，则向前移动指针。当读到结束标志 `$` 时循环结束。

错误处理

本程序在进行预测分析时，可以分析四种错误。当前栈顶符号和输入符号不匹配时，将预测分析栈中栈顶符号弹出；当读到预测分析表中无相应生成式，含有同步信息，从栈顶弹出非终结符；预测分析表中无相应生成式，也不含有同步信息，则向前移动指针；当栈顶符号为 `$`，但是指针指向的字符不为 `$` 时，则结束循环。

四、程序输入输出及执行结果

输入形式

将以文件形式输入，每行一个生成式，左部右部之间以 `->` 连接。同一个非终结符的多个产生式分多行存入文件。程序运行时将提示用户输入文件名和符号串。程序根据输入进行分析。

输出形式

首先提示用户输入文件名，程序分析读入的文法后将输入FIRST集、FOLLOW集和预测分析表。然后提示用户输入符号串，程序根据符号串逐步输出每步的栈中内容，符号串剩余未分析部分以及替换栈顶符号用到的产生式。如果遇到错误则输出相应的错误内容。

执行结果

t1:

请输入进行语法分析的文法

t1.txt

FIRST:

E:(n

F:(n

P:+ - @

Q:* / @

T:(n

FOLLOW:

E:\$)

F:\$) * + - /

P:\$)

Q:\$) + -

T:\$) + -

	+ - * / () n \$							
E					E->TP	synch	E->TP	synch
P	P->+TP	P->-TP				P->@	P->@	
T	synch	synch			T->FQ	synch	T->FQ	synch
Q	Q->@	Q->@	Q->*FQ	Q->/FQ		Q->@	Q->@	
F	synch	synch	synch	synch	F->(E)	synch	F->n	synch

请输入要进行语法分析的符号串

n+n-n*(n/n)

E->TP	\$PT	n+n-n*(n/n)\$
T->FQ	\$PQF	n+n-n*(n/n)\$
F->n	\$PQn	n+n-n*(n/n)\$
	\$PQ	+n-n*(n/n)\$
Q->@	\$P@	+n-n*(n/n)\$
	\$P	+n-n*(n/n)\$
P->+TP	\$PT+	+n-n*(n/n)\$
	\$PT	n-n*(n/n)\$
T->FQ	\$PQF	n-n*(n/n)\$
F->n	\$PQn	n-n*(n/n)\$
	\$PQ	-n*(n/n)\$
Q->@	\$P@	-n*(n/n)\$
	\$P	-n*(n/n)\$
P->-TP	\$PT-	-n*(n/n)\$
	\$PT	n*(n/n)\$
T->FQ	\$PQF	n*(n/n)\$
F->n	\$PQn	n*(n/n)\$
	\$PQ	*(n/n)\$
Q->*FQ	\$PQF*	*(n/n)\$
	\$PQF	(n/n)\$
F->(E)	\$PQ)E((n/n)\$
	\$PQ)E	n/n)\$
E->TP	\$PQ)PT	n/n)\$
T->FQ	\$PQ)PQF	n/n)\$
F->n	\$PQ)PQn	n/n)\$
	\$PQ)PQ	/n)\$
Q->/FQ	\$PQ)PQF/	/n)\$
	\$PQ)PQF	n)\$
F->n	\$PQ)PQn	n)\$
	\$PQ)PQ)\$
Q->@	\$PQ)P@)\$
	\$PQ)P)\$
P->@	\$PQ)@)\$
	\$PQ))\$
	\$PQ	\$
Q->@	\$P@	\$
	\$P	\$
P->@	\$@	\$
	\$	\$

```
请输入进行语法分析的符号串
nn++n

      E->TP          $PT          nn++n$
      T->FQ          $PQF         nn++n$
      F->n           $PQn         nn++n$
                        $PQ         n++n$
预测分析表中无相应生成式，不含有同步信息，向前移动指针
                        $PQ         ++n$
      Q->@           $P@         ++n$
                        $P         ++n$
      P->+TP         $PT+         ++n$
                        $PT         +n$
预测分析表中无相应生成式，含有同步信息，从栈顶弹出非终结符
                        $P         +n$
      P->+TP         $PT+         +n$
                        $PT         n$
      T->FQ          $PQF         n$
      F->n           $PQn         n$
                        $PQ         $
      Q->@           $P@         $
                        $P         $
      P->@           $@         $
                        $         $
```

t2:

```
请输入进行语法分析的文法
t2.txt
FIRST:
E:1
F:0
P:1 @
FOLLOW:
E:$
F:$
P:$

      1      0      $
E      E->1F      synch
F      F->0P      synch
P      P->1      P->@
```

```
请输入进行语法分析的符号串
101

      E->1F          $F1          101$
                        $F          01$
      F->0P          $P0          01$
                        $P          1$
      P->1          $1          1$
                        $          $
```

t3:

请输入要进行语法分析的文法

t3.txt

FIRST:

E:(i

F:(i

P:+ @

Q:* @

T:(i

FOLLOW:

E:\$)

F:\$) * +

P:\$)

Q:\$) +

T:\$) +

	+	*	()	i	\$
E			E->TP	synch	E->TP	synch
P	P->+TP			P->@		P->@
T	synch		T->FQ	synch	T->FQ	synch
Q	Q->@	Q->*FQ		Q->@		Q->@
F	synch	synch	F->(E)	synch	F->i	synch

请输入要进行语法分析的符号串

i+i*(i)

E->TP	\$PT	i+i*(i)\$
T->FQ	\$PQF	i+i*(i)\$
F->i	\$PQi	i+i*(i)\$
	\$PQ	+i*(i)\$
Q->@	\$P@	+i*(i)\$
	\$P	+i*(i)\$
P->+TP	\$PT+	+i*(i)\$
	\$PT	i*(i)\$
T->FQ	\$PQF	i*(i)\$
F->i	\$PQi	i*(i)\$
	\$PQ	*(i)\$
Q->*FQ	\$PQF*	*(i)\$
	\$PQF	(i)\$
F->(E)	\$PQ)E((i)\$
	\$PQ)E	i)\$
E->TP	\$PQ)PT	i)\$
T->FQ	\$PQ)PQF	i)\$
F->i	\$PQ)PQi	i)\$
	\$PQ)PQ)\$
Q->@	\$PQ)P@)\$
	\$PQ)P)\$
P->@	\$PQ)@)\$
	\$PQ))\$
	\$PQ	\$
Q->@	\$P@	\$
	\$P	\$
P->@	\$@	\$
	\$	\$

请输入要进行语法分析的符号串

i((i++*i)

E->TP

\$PT

i((i++*i)\$

T->FQ

\$PQF

i((i++*i)\$

F->i

\$PQi

i((i++*i)\$

\$PQ

((i++*i)\$

预测分析表中无相应生成式，不含有同步信息，向前移动指针

\$PQ

(i++*i)\$

预测分析表中无相应生成式，不含有同步信息，向前移动指针

\$PQ

i++*i)\$

预测分析表中无相应生成式，不含有同步信息，向前移动指针

\$PQ

++*i)\$

Q->@

\$P@

++*i)\$

\$P

++*i)\$

P->+TP

\$PT+

++*i)\$

\$PT

+*i)\$

预测分析表中无相应生成式，含有同步信息，从栈顶弹出非终结符

\$P

+*i)\$

P->+TP

\$PT+

+*i)\$

\$PT

*i)\$

预测分析表中无相应生成式，不含有同步信息，向前移动指针

\$PT

i)\$

T->FQ

\$PQF

i)\$

F->i

\$PQi

i)\$

\$PQ

)\$

Q->@

\$P@

)\$

\$P

)\$

P->@

\$@

)\$

\$

)\$

输入串长度不符合文法要求

t4:

请输入要进行语法分析的文法

t4.txt

FIRST:

A:a d

B:a d e

C:@ a d

D:@ a d

E:@ d

FOLLOW:

A:\$ a b c d e

B:b c

C:a d

D:b c e

E:a d

	a	b	e	d	c	\$
A	A->aBbDe	synch	synch	A->d	synch	synch
B	B->CAD	synch	B->e	B->CAD	synch	
C	C->ABc			C->dE		
D	D->Ae	D->@	D->@	D->Ae	D->@	
E	E->@			E->dE		

请输入要进行语法分析的符号串

请输入要进行语法分析的符号串
aebe

A->aBbDe	\$eDbBa	aebe\$
	\$eDbB	ebe\$
B->e	\$eDbe	ebe\$
	\$eDb	be\$
	\$eD	e\$
D->@	\$e@	e\$
	\$e	e\$
	\$	\$

t5:

请输入要进行语法分析的文法
t5.txt

FIRST:

A:a b c

B:@ a b

C:c

D:@ a

E:@ b

FOLLOW:

A:\$

B:\$ a b c

C:\$

D:\$

E:\$ a

	a	c	b	\$
A	A->BC	A->BC	A->BC	synch
B	B->Ea	B->@	B->Ea	B->@
C		C->cD		synch
D	D->aBED			D->@
E	E->@		E->b	E->@

请输入要进行语法分析的符号串

请输入要进行语法分析的符号串
baca

A->BC	\$CB	baca\$
B->Ea	\$CaE	baca\$
E->b	\$Cab	baca\$
	\$Ca	aca\$
	\$C	ca\$
C->cD	\$Dc	ca\$
	\$D	a\$
D->aBED	\$DEBa	a\$
	\$DEB	\$
B->@	\$DE@	\$
	\$DE	\$
E->@	\$D@	\$
	\$D	\$
D->@	\$@	\$
	\$	\$