中山大学数据科学与计算机学院本科生实验报告

课程名称：编译器构造实验 任课教师：陈炬桦 教学助理（TA）：

|  |  |  |  |
| --- | --- | --- | --- |
| 学年度 | 2018-2019 | 学期 | 第二学期 |
| 年级 | 2016级 | 专业（方向） | 计算机科学与技术 |
| 学号 | 16337233 | 姓名 | 王凯祺 |
| 电话 | 13434386182 | Email | [Wangkq3@mail2.sysu.edu.cn](mailto:Wangkq3@mail2.sysu.edu.cn) |
| 开始日期 | 2019/3/8 | 完成日期 | 2019/3/8 |

1. 实验题目

**1.1 Description**

 输入开始符号，非终结符，终结符，产生式  
按非终结符顺序输出产生式；

**1.2 Input**

 输入开始符号；  
非终结符个数，非终结符，空格符分隔；  
终结符个数，终结符，空格符分隔；  
产生式的个数，各产生式的左边和右边符号，空格符分隔；

**1.3 Output**

G[开始符号]：  
按非终结符顺序输出各产生式；

2. 算法描述(介绍程序模块功能；流程图)

首先将产生式按产生式左边分类。那么每个非终结符都会对应多个产生式。对于每个终结符，输出他们对应的产生式即可。

3. 测试数据

样例输入

Z

8 Z E F P G T Q S

3 + \* i

18

Z E+T

E E

P G

F F

P G

G G

T T\*i

Q E

S i

E S+F

F FP

G GG

Q E+F

E T

F P

G F

Q T

Q S

样例输出

G[Z]:

Z::=E+T

E::=E | S+F | T

F::=F | FP | P

P::=G | G

G::=G | GG | F

T::=T\*i

Q::=E | E+F | T | S

S::=i

4. 程序清单

#include <bits/stdc++.h>

#define N 10020

using namespace std;

string s;

vector< string > vn, vt, v[200];

char st[N];

int main() {

scanf("%s", st);

s = st;

int n;

scanf("%d", &n);

for (int i = 1; i <= n; ++i) {

scanf("%s", st);

vn.push\_back((string)st);

}

scanf("%d", &n);

for (int i = 1; i <= n; ++i) {

scanf("%s", st);

vt.push\_back((string)st);

}

scanf("%d", &n);

while (n--) {

int x = -1;

scanf("%s", st);

for (int i = 0; i < (int)vn.size(); ++i)

if (vn[i] == (string)st) {

x = i;

break;

}

scanf("%s", st);

v[x].push\_back(st);

}

printf("G[%s]:\n", s.c\_str());

for (int i = 0; i < (int)vn.size(); ++i) {

printf("%s::=", vn[i].c\_str());

printf("%s", v[i][0].c\_str());

for (int j = 1; j < (int)v[i].size(); ++j)

printf(" | %s", v[i][j].c\_str());

puts("");

}

}