**实验一、无符号数的识别**

姓名：马佳敏 学号：201906021316

1. 问题描述

设计识别Pascal的无符号数的算法。如2021、11.28、63.6E8、1.9E6这样的数。

1. 算法设计

Pascal的无符号数的正规式如图1所示。

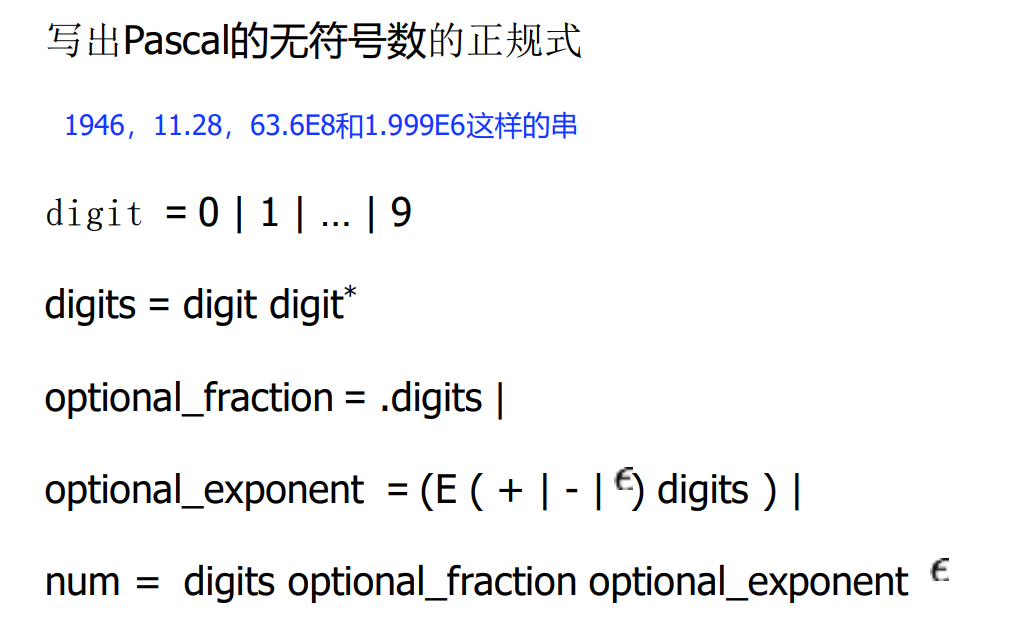


图1无符号数的正规式

其中应该符合的规范（此处不考虑0开头的情况）：

1. 输入合法字符（数字、‘.’、‘E’、‘+|-’）；
2. 若同时存在'.'和'E'，'.'必须出现在'E'之前；
3. 若存在'+|-'，则必须存在'E'；
4. 若同时存在'E'和'+|-'，'E'必须出现在'+|-'前一位；
5. 三个符号都不能位于第一位和最后一位。

其对应的状态自动机如图2所示。

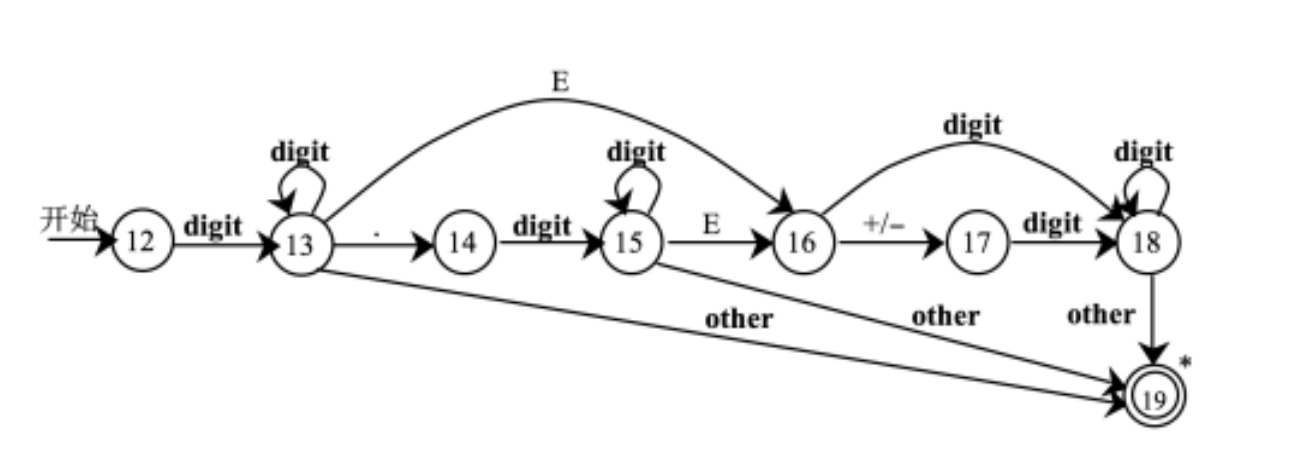


图2 无符号数的状态自动机

1. 源代码

#include<iostream>

using namespace std;

bool isNumeric(string s) {

int index1 = 0; // 记录'.'位置

int index2 = 0; // 记录'E'位置

int index3 = 0; // 记录'+|-'位置

for (int i = 0; i < s.length(); i++) {

// 对于不该出现的字符，直接返回false

if ((s[i] > '9' || s[i] < '0') && s[i] != 'E' && s[i] != '.' && s[i] != '+' && s[i] != '-') {

cout << "this is not a number" << endl << endl;

return false;

}

// 遇到'.'符号，若其不在第一位且index1为0，表示'.'未出现过，将位置赋值给index1

if (s[i] == '.') {

if (index1 != 0 || i == 0) return false;

else index1 = i;

}

// 遇到'E'符号，若其不在第一位且index2为0，表示'E'未出现过，将位置赋值给index2

else if (s[i] == 'E') {

if (index2 != 0 || i == 0) return false;

else index2 = i;

}

// 遇到'+|-'符号，若其不在第一位且index3为0，表示'+|-'未出现过，将位置赋值给index3

else if (s[i] == '+' || s[i] == '-') {

if (index3 != 0 || i == 0) return false;

else index3 = i;

}

// 若同时存在'.'和'E'，'.'必须出现在'E'之前

if (index1 != 0 && index2 != 0 && index1 > index2) {

cout << "false, 'E' is not behind '.'" << endl << endl;

return false;

}

// 若存在'+|-'，则必须存在'E'

else if (index2 == 0 && index3 != 0) {

cout << "false, only exits '+|-'" << endl << endl;

return false;

}

// 若同时存在'E'和'+|-'，'E'必须出现在'+|-'前一位

else if (index2 != 0 && index3 != 0 && index2 + 1 != index3) {

cout << "false, '+|-' must be after with 'E'" << endl << endl;

return false;

}

// 三个符号都不能位于最后一位

else if (index1 == s.length() - 1) {

cout << "false, '.' in the last" << endl << endl;

return false;

}

else if (index2 == s.length() - 1) {

cout << "false, 'E' in the last" << endl << endl;

return false;

}

else if (index3 == s.length() - 1) {

cout << "false, '+|-' in the last" << endl << endl;

return false;

}

}

cout << "true" << endl << endl;

return true;

}

int main() {

string s;

while (cin >> s)

isNumeric(s);

return 0;

}

1. 测试结果

