

编译原理 实验一

171860578 张梓悦 基础班

a) 我的程序实现了的功能：

完成全部选做内容（8进制数、16进制数、指数形式的浮点数的报错均按照词法错误处理）

完成全部必做内容

b) 选做内容的实现：

识别8进制和16进制数（按照词法错误处理）：

通过正则表达式识别8进制和16进制数

```
int      0|[1-9]+{digit}*
int8     0[0-7]+
int16    0[xX][0-9a-fA-F]+
```

同时也通过正则表达式识别错误的8进制、16进制数，比如09,0x1G。

```
wint8    0({digit}|[letter])+
wint16   0[xX]({digit}|{letter})+
wint     {digit}*|{digit}+({digit}|[a-zA-Z])+
```

错误的8进制、16进制数按照词法错误处理

```
{printf("Error type A at Line %d: Illegal hexadecimal number '%s'\n",yylineno, yytext);
yylval.node_type = NULL;
eflag=true;
return INT;}
{printf("Error type A at Line %d: Illegal octal number '%s'\n",yylineno, yytext);
yylval.node_type = NULL;
eflag=true;
```

最终按照如下的形式输出词法错误：

```
Error type A at Line 3: Illegal octal number '09'.
Error type A at Line 4: Illegal hexadecimal number '0x3G'.
```

识别指数形式的浮点数（按照词法错误处理）：

通过正则表达式识别指数形式的浮点数

```
float2      ([0-9]*[\\.] [0-9]+|[0-9]+[\\.] ) [eE] [-+]? {digit}+
```

同时也通过正则表达式识别错误的指数形式的浮点数，比如1.05e, .e5。

```
wfloat2     \. [eE] [-+]? {digit}+| ([0-9]*[\\.] [0-9]+|[0-9]+[\\.] ) [eE] [-+]?
wfloat3     [0-9]+[\\.] [eE] ({digit}|[a-zA-Z])+
```

错误的指数形式的浮点数照词法错误处理

```
{printf("Error type A at Line %d: Illegal floating point number \"%s\".\n",yylineno, yytext);
yyval.node_type = NULL;|
eflag=true;
return FLOAT;}
```

最终按照如下的形式输出词法错误：

```
Error type A at Line 2: Illegal floating point number "1.05e".
Error type A at Line 3: Illegal floating point number ".e5".
```

识别“//”和“/*...*/”形式的注释（按照词法错误处理）：

通过正则表达式识别注释，/**/形式的注释参考了网上的博客

<http://www.voidcn.com/article/p-qxyozbrt-bud.html>

```
"/*" [^*]* [*] + ([^* /] [^*]* [*] +)* "/"
```

```
"//".*
```

同时也通过正则表达式识别错误的单独出现的注释。

对于单独出现的“/*”输出词法错误：

```
Error type A at Line 9: unterminated comment '/*'.
```

对于单独出现的“*/”，由于连续识别出乘号和除号，不符合文法的定义，故输出语法错误：

```
Error type B at Line 8: syntax error.
```

c) 语法错误的实现：

对于所有的语法错误，错误信息给出syntax error：

```
Error type B at Line 4: syntax error.  
Error type B at Line 5: syntax error.
```

d) 我的程序通过makefile进行编译

```
ziyuezhang@ziyuezhang-virtual-machine:~/Desktop/compiler/Code$ make  
flex -o ./lex.yy.c ./lexical.l  
bison -o ./syntax.tab.c -d -v ./syntax.y  
gcc -c ./syntax.tab.c -o ./syntax.tab.o  
gcc -o parser ./tree.o ./syntax.tab.o ./main.o -lfl -ly
```