

实验 1-1 词法分析器设计与实验设计说明

一、运行和开发环境

- ①无图形界面：Win10 下 Visual Studio 开发，通过.exe 可执行文件在 cmd 窗口下运行。
- ②图形化界面：Win10 下 Qt5 开发，通过.exe 可执行文件由窗体程序运行。

二、功能

能识别的单词：

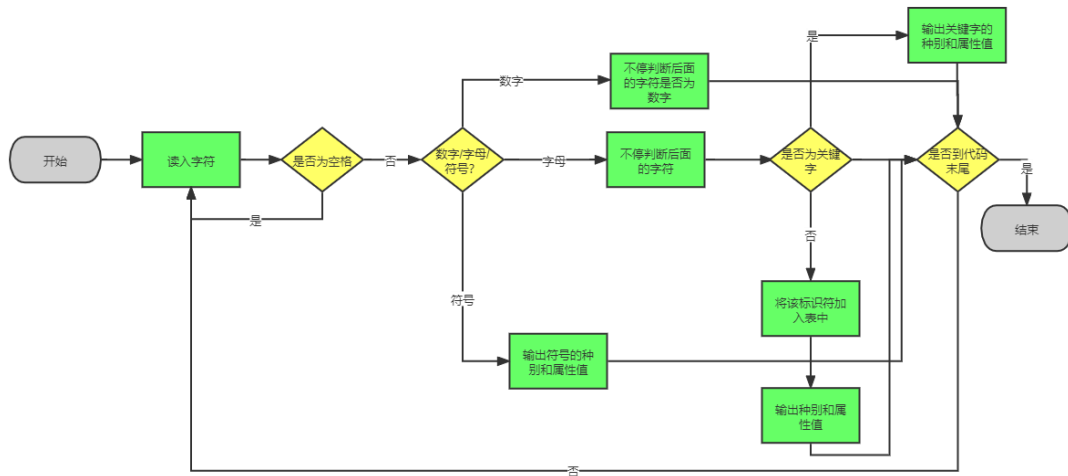
- ✓ 关键字：int | void | if | else | while | return
- ✓ 标识符：字母（字母|数字）* （注：不与关键字相同）
- ✓ 数值：数字（数字）*
- ✓ 赋值号： =
- ✓ 算符： + | - | * | / | = | == | > | >= | < | <= | !=
- ✓ 界符： ;
- ✓ 分隔符： ,
- ✓ 注释号： /* */ | //
- ✓ 左括号： (
- ✓ 右括号：)
- ✓ 左大括号： {
- ✓ 右大括号： }
- ✓ 字母：| a |...| z | A |...| Z |
- ✓ 数字：0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
- ✓ 结束符： #

扩充单词：

除上述作业要求单词以外，还扩充了如下单词：

- ✓ 关键字：char | const | unsigned | bool | true | false
- ✓ 左方括号： [
- ✓ 右方括号：]
- ✓ 单引号： '
- ✓ 双引号： "

三、主程序框图



四、运行结果

①无图形界面

没有使用图形界面，打开可执行文件（作业目录下的 lexical_analyzer_console.exe），可以看见输入代码的提示。

```

Please input the code(end with #):

```

通过控制台键盘输入代码，另起一行顶格输入#结束。（同目录下有 test.txt 的测试用例）

```

Please input the code(end with #):
int main()
{
    int a=0;
    int b=3;
    char c='A';
    bool d[3]={true,false,false};
    a=a+b+c;
    return 0;
}
#

```

输出结果如下所示：

```

<$char, ->
<$ID, 4>
<$ASSIGN, ->
<$SINGLE_QUOTE, ->
<$ID, 5>
<$SINGLE_QUOTE, ->
<$BOUND, ->
<$bool, ->
<$ID, 6>
<$LSQUARE_BRACKET, ->
<$VALUE, 3>
<$RSQUARE_BRACKET, ->
<$ASSIGN, ->
<$LBRACE, ->
<$true, ->
<$false, ->
<$false, ->
<$BOUND, ->
<$ID, 2>
<$ASSIGN, ->
<$ID, 2>
<$PLUS, ->
<$ID, 3>
<$PLUS, ->
<$ID, 4>
<$BOUND, ->
<$return, ->
<$VALUE, 0>
<$BOUND, ->
<$RBRACE, ->

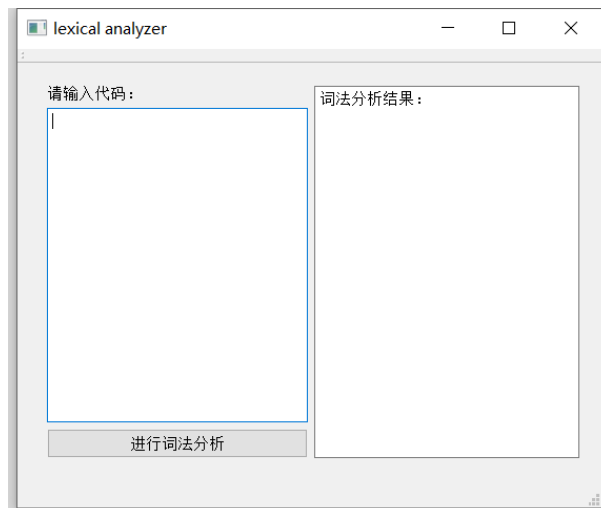
```

②图形化界面

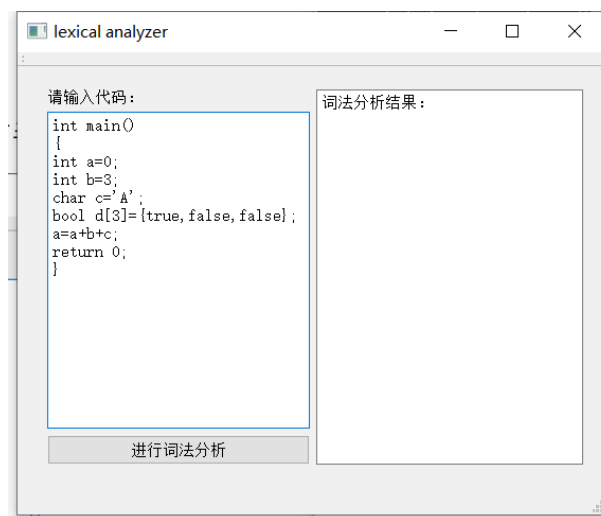
打开可执行文件 lexical_anlayzer_gui（作业目录下的 lexical_analyzer_gui/lexical_analyzer_gui.exe），可运行图形化的词法分析器。

图形化界面如图所示

左半部分输入代码，右半部分输出词法分析的结果。



输入代码：



按下“进行词法分析”的按钮，右方输出词法分析结果。

