

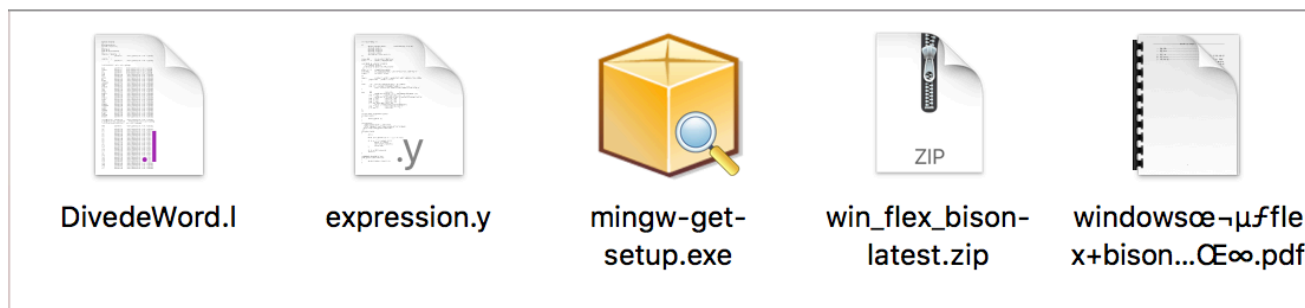
Linux环境下使用flex与bison

获取.l与.y文件

- 从FTP路径中找到文件

/李冬梅/授课材料/编译原理/实验/tool/Windows下的flex+bison_gcc_by_李伟.zip

- 下载之后将文件解压，我们只需要里面含有.l与.y的文件



- 之后，打开命令行，并进入到解压文件的目录，可以看到相应的文件：

```
Windows下的flex+bison+gcc_by_李伟 ls
DivedeWord.l          mingw-get-setup.exe  windowsæ-µfflex+bison+gcc_by_200æ.pdf
expression.y          win_flex_bison-latest.zip
```

安装flex与bison

- 执行如下命令：

```
sudo apt-get install flex
```

```
sudo apt-get install bison
```

文件转码

- 由于文件来自于Windows，国内Windows系统默认文件编码为GBK，而Linux编码为utf-8，因此我们要将文件转化成utf-8
- 使用iconv命令，将文件转化成utf-8格式，命令如下：

```
iconv -f GBK -t utf8 DivedeWord.l > divide.l
```

我们将使用新生成的**divide.l**文件

- 使用iconv命令，将文件转化成utf-8格式，命令如下：

```
iconv -f GBK -t utf8 expression.y > expr.y
```

我们将使用新生成的**expr.y**文件

flex使用

- **flex**是**lex**在GNU中的实现，因此在Linux系统中自带，可用 `man flex` 查看其功能(看不看都行)。我们可以直接使用**flex**命令对.l文件进行处理：

```
flex divide.l
```

执行完此命令后，会生成一个.c文件：**lex.yy.c**，这个文件是一个**C语言**的文件，我们将使用**gcc**编译并执行此文件

- 执行如下命令：

```
gcc lex.yy.c -o divide
```

这个命令生成的divide文件为可执行文件，执行它：

```
./divide
```

之后会进入等待输入的状态，随意输入字符，程序将会输出该字符所属类别

bison使用

- **bison**是**yacc**在GNU中的实现，因此在Linux系统中自带，可用 `man yacc` 查看其功能(看不看都行)。我们可以直接使用**bison**命令对.l文件进行处理：

```
bison expr.y
```

执行完此命令后，会生成一个.c文件：**expr.tab.c**，这个文件是一个C语言的文件，我们将使用gcc编译并执行此文件

- 执行如下命令：

```
gcc expr.tab.c -o expression
```

(若此命令执行失败，可能是因为未关联math库，可用此命令替代：

```
gcc expr.tab.c -o expression -lm )
```

这个命令生成的divide文件为可执行文件，执行它：

```
./expression
```

之后会进入命令行状态，随意输入表达式，程序将会计算该表达式

