

学号：19373075

姓名：许天识

思路方法

语法分析部分依旧是递归下降辅助以算符优先。对于关键的变量与函数，由于部分知识尚未学到，因此自行设计了一个方法(可能要大幅重构)。在语义分析程序中，增加变量池与常量池，时刻维护已经出现的变量、常量，变量保存两个寄存器（初始寄存器及被load的寄存器），常量则保存数值大小。

具体细节

变量

变量的关键在于Exp的计算。首先将运算数栈内元素打包成为StackNumEle类，结构如下：

```
private boolean isNumber; //判断是常量还是变量
private Integer number; //是常量保存常量的值
private Var var; //是变量保存var
```

变量var结构如下：

```
//被分配空间的寄存器
private int true_register;
//值被加载出去的寄存器，一旦用到该变量并且该变量还没有load寄存器则赋予一个新的，同时只要变量的值
发生改变，load寄存器也要同时赋予新的。
private int load_register;
//变量类型，目前都是i32
private String type;
```

算符优先法流程不变，重点在于如何归约。

对于变量和常量的定义，我设计了两个不同的归约函数：

对于常量定义

此时运算数栈内只可能是数字(常量在之前已被转换为了数字)，因此可以直接计算出值。最后再翻译即可。

对于变量定义

此时我采用的方法是不进行任何实际计算，只翻译出运算代码，每归约一步翻译一句。

运算数栈内是数字就表示为：

```
"elem.getNumber()"
```

是变量就表示为：

```
" %"+var.getLoad_register();
```

同时将中间运算式子也当作是变量压入栈内，赋予新的寄存器。最后根据不同的运算符翻译出指令语句。

函数

函数本质上是Exp中的元素，因此我采用在Exp识别函数中添加对函数的识别，函数的参数也是Exp，只需要递归调用即可。

关键问题解决方案

1.如何判断函数调用结束？

显然原先的方法已经失效，函数调用结束的标志在于一对左右括号，因此我在调用Exp()识别函数参数时，记录了左右括号的相对数量，当左右括号完美匹配则不再读取token，转向规约，函数结束。

同时要注意部分函数如get类函数它的返回值也是Exp的成员之一，因此需要压入运算数栈中。