冯诺依曼结构？？

class scanner: 读入处理器

每次处理一句话（以’\n’为结束标志）功能是把一句话划分成每个token

nextToken返回scanner的一个token，同时在scanner中删除此token

class token：标志类

enum tokentype{ Register, Label, Number, String, Pointer, Command}

gettype 确定这个token的type，去掉一些无用的字符（如”,” “：”等）

static map<string, int>key\_word 是关键字（指令）到int的映射//可改成到相应函数的映射

双目运算符：

计算步骤（binary\_operator）第一个参数传寄存器地址；

第二个参数如果是地址，则传地址，如果是常数，则传NULL，常数记在tmp中

memory：

1.栈空间(sp)

2.堆空间（.data 。word。。。。）（从0开始向上）(fp)

operation类

成员设计：两个register编号，一个number数据，一个运算编号 一个anser

双目运算符（存两个相对指针，若为立即数，则指针置为-1，数字记在number中）

乘除：双参、三参区别（reg1的编号是否为-1）

单目运算符 存两个

b系指令，dest存label编号，reg1 reg2同上

l s系指令 register(dest) address(reg1 + number)// 如果是内存address， reg1 = -1，label string存到一个临时vector里，number存vector下标，所有read\_in指令做完以后再解码一遍，reg1 = -1, number存内存相对地址

static成员vector<operation> text 存放程序 map<string, int>存放label

error:label 冲突？？

operation成员函数execute 算出anser，完成所有指令

state1:run\_

state2:对象内部，传回参数opp1, opp2

state3:run\_ 计算，填入ans

state4:对象内部

state5：run\_