int a,b; //不允许

int a; //允许

float c;

char d;

string e; (也就是char \* 类型)

func main()void {

int a,b,c;

int[5] arr;

a = 5;

b = 6;

arr[5] = 10;//不允许 arr$5+5 = arr[5+5]

c = fff(a,b);

e = alloc(“aaa”);

if(a < b){

int c = 0;

c = c - 10;

a = c;

}else{

b = 12;

}

while(b != 100){

b = b+1;

if(b % 2 == 0) break;

}

return;

}

func fff(int a, int b)int{

int c = a +b;

return c;

}

关键字：

T\_int : int类型

T\_float : float类型

T\_char: char类型

T\_string : string类型

T\_intArr: int数组类型

T\_floatArr: float数组类型

T\_func : 函数标识

T\_alloc: alloc，字符串分配

T\_free: 字符串释放（暂时不加）

运算符：

T\_add +

T\_sub -

T\_mul \*

T\_div /

T\_mod %

T\_equal ==

T\_big >

T\_bige >=

T\_small <

T\_smalle <=

T\_nequal !=

T\_leftmove <<

T\_rightmove >>

T\_land &

T\_lor |

T\_lnot ~

T\_lxor ^

T\_and &&

T\_or ||

T\_not !

T\_assign =

T\_getarr $

控制流：

T\_if if

T\_else else

T\_while while

T\_break break

T\_continue continue

T\_return return

常量：

T\_cst\_int int常量

T\_cst\_float float常量

T\_cst\_char char 常量

注释：

T\_begin /\*

T\_end \*/

标识符：

T\_identifier 标识符

T\_ls (

T\_rs )

T\_lb {

T\_rb }

T\_semi ;

T\_comma ,

~~T\_lm [~~

~~T\_rm ]~~

非终结符：

Program 程序

EXTDEFLIST 定义列表，包括函数定义、变量定义

FUNC 函数定义

FUNC\_VAR 函数形参

FUNC\_VAR\_LIST 函数形参列表

FUNC\_ARG 函数实参

FUNC\_ARG\_LIST 函数实参列表

T\_TYPE 类型

DECLAREVAL 变量定义

DECLAREVALLIST 变量定义列表

BODY 语句体

STMTLIST 语句列表

STMT 语句

IF\_STMT if语句

IF\_ELSE\_STMT if\_else语句

WHILE\_STMT while语句

CONST 常量类型

EXP 表达式

T\_ID 标识符

G[program]:

Program -> EXTDEFLIST

EXTDEFLIST -> FUNC EXTDEFLIST | DECLAREVAL EXTDEFLIST | ε

//函数部分

FUNC -> T\_func T\_identifier( FUNC\_VAR\_LIST ) T\_TYPE {BODY}

FUNC\_VAR\_LIST -> ε | T\_TYPE T\_identifier FUNC\_VAR

FUNC\_VAR -> ε | T\_comma T\_TYPE T\_identifier

T\_TYPE -> T\_int | T\_float |T\_char |T\_string | T\_intArr |T\_floatArr

//语句部分

BODY-> { DECLAREVAL STMTLIST }

DECLAREVAL -> ε | T\_TYPE T\_identifier T\_semi DECLAREVAL

| T\_TYPE T\_identifier = EXP T\_semi DECLAREVAL

STMTLIST -> ε | STMT STMTLIST

STMT -> ε | EXP T\_semi |BODY | T\_return EXP T\_semi| T\_if(EXP) STMT

| T\_if (EXP) STMT T\_else STMT; | T\_while(EXP) STMT

//表达式部分

CONST -> T\_cst\_int | T\_cst\_float | T\_cst\_char

EXP -> CONST | T\_identifier | (EXP)

| EXP T\_and EXP | EXP T\_assign EXP | EXP T\_or EXP | T\_not EXP

| T\_not EXP | EXP T\_land EXP | EXP T\_lor EXP | T\_lnot EXP | EXP T\_lxor EXP| EXP T\_add EXP

| EXP T\_sub EXP | EXP T\_mul EXP | EXP T\_div EXP | T\_sub EXP | EXP T\_equal EXP

| EXP T\_big EXP | EXP T\_bige EXP | EXP T\_small EXP |EXP T\_smalle EXP | EXP T\_nequal EXP

| EXP T\_leftmove EXP | EXP T\_rightmove EXP | T\_alloc (“xxx”) | T\_free(EXP)

| T\_identifier T\_getarr EXP | T\_TYPE(EXP) | EXP T\_mod EXP

| T\_identifier T\_ls FUNC\_ARG\_LIST T\_rs

//函数调用

FUNC\_ARG\_LIST -> ε | EXP FUNC\_ARG

FUNC\_ARG -> ε | T\_comma EXP FUNC\_ARG

优先级及结合性

1. $(获得数组元素，如a$1和C语言中的a[1]相等)
2. 强制转换，T\_sub（单目），T\_not, T\_lnot,
3. T\_div T\_mul, T\_mod
4. T\_sub, T\_add
5. T\_leftmove, T\_rightmove
6. T\_big, T\_bige, T\_small, T\_smalle
7. T\_equal, T\_nequal
8. T\_land
9. T\_lxor
10. T\_lor
11. T\_and
12. T\_or
13. T\_assign

函数调用返回值：$v0 - $v1

函数传参寄存器：$a0-$a3

数据存储器：$t0-$t9

保存数据：$s0-$s7

返回地址：$ra

EXP

1、判断是否是左值，如果是则它在符号表中的位置（用来找数据段的偏移量也即变量地址）

2、判断运算符类型

3、判断操作数个数及类型

编译的时候，默认EXP的计算结果存放在$t0。首先将左子树的值，然后将$t0移动到 $t1，然后计算右子树的值，然后计算($t0 op $t1)并保存在$t0中。

STMTLIST

1、根据STMTLIST来