## **10.2 (1)**：

### 一、三地址代码

LABEL l0 // while入口  
t1 = 1  
t2 = x  
IF t1 < t2 THEN l1 ELSE l6 // 1 < x ? l1 : l6  
  
LABEL l1  
t3 = y  
t4 = 1  
IF t3 > t4 THEN l2 ELSE l6 // y > 1 ? l2 : l6  
  
LABEL l2  
t5 = -n // -n  
t6 = t5 + 2 // -n + 2  
t7 = x  
t8 = y  
t9 = t7 \* t8 // x \* y  
IF t6 < t9 THEN l3 ELSE l4 // 条件判断  
  
LABEL l3  
a = 100 // if 真分支  
GOTO l5  
  
LABEL l4  
b = a // else 分支  
  
LABEL l5  
GOTO l0 // 回到while入口  
  
LABEL l6 // while结束

### 二、语法结构与注释语法树

对应产生式与主要属性如下：

S → while (B) S1  
B → B1 && B2  
 B1 → E1 < E2 (E1 = 1, E2 = x)  
 B2 → E3 > E4 (E3 = y, E4 = 1)  
  
S1 → if (B3) S2 else S3  
B3 → E5 < E6  
 E5 → -n + 2  
 E6 → x \* y  
  
S2 → a = 100  
S3 → b = a

属性说明：

* B.tc = l2（while条件为真时的跳转入口）
* B.fc = l6（while条件为假时跳转出口）
* S.code 是结构化代码块，入口 l0，出口 l6

## **习题 10.3**

### 文法：

S → for (S1; B; S2) S3  
 code = S1.code ++  
 gen[LABEL l1] ++  
 B.code ++  
 gen[LABEL B.tc] ++  
 S3.code ++  
 S2.code ++  
 gen[GOTO l1] ++  
 gen[LABEL B.fc]

### 输入句子：

for(i=0; i<100; i=i+1) print i;

### 分解：

* S1: i = 0
* B : i < 100
* S2: i = i + 1
* S3: print i

### 三地址代码：

t1 = 0  
i = t1  
LABEL l1  
t2 = 100  
IF i < t2 THEN l2 ELSE l3  
LABEL l2  
PRINT i  
t3 = i + 1  
i = t3  
GOTO l1  
LABEL l3

## 习题 10.5：给定函数的符号表和三地址代码

### 原代码片段：

int x; float z;  
int a[10,20], b[6];  
  
float bar(int brr[6]) {  
 float x;  
 x = brr[0] + brr[5];  
 return x;  
}  
  
float foo(int x; float boo(); int arr[10,10];) {  
 if (x==0) z = sqrt(boo(arr[0][0]));  
 else return boo(arr[x][x]);  
}

### ➤ 符号表构造（使用全局命名）：

#### @table:

entry: x INT offset: 4  
entry: z FLO offset: 8  
entry: a ARRAY base: 812 dims:2 dim[0]:10 dim[1]:20 etype:INT  
entry: b ARRAY base: 892 dims:1 dim[0]:6 etype:INT  
entry: bar FUNC offset: 916 mytab: bar@table  
entry: foo FUNC offset: 924 mytab: foo@table

#### bar@table:

outer: @table width: 8 argc:1 arglist: (brr)  
entry: brr ARRPTT offset: 4 etype: INT  
entry: x FLO offset: 8  
rtype: FLO

#### foo@table:

outer: @table width: 28 argc:3 arglist: (x boo arr)  
entry: x INT offset: 4  
entry: boo FUNPTT offset: 8 rtype: FLO  
entry: arr ARRPTT offset: 16 etype: INT  
rtype: FLO

### ➤ 三地址代码：

#### bar@code:

t1 = brr[0]  
t2 = brr[5]  
t3 = t1 + t2  
RETURN t3

#### foo@code:

IF x == 0 THEN l1 ELSE l2  
LABEL l1  
t4 = 0  
t5 = arr[t4] // arr[0][0] 平铺访问  
PAR t5  
t6 = CALL boo, 1  
t7 = CALL sqrt, 1  
z = t7  
GOTO l3  
LABEL l2  
t8 = x  
t9 = arr[t8]  
PAR t9  
t10 = CALL boo, 1  
RETURN t10  
LABEL l3