**8.6 总共可以分为9个基本块（最后的return 单独算一个block），因为有goto 15,goto 9,goto 13 ,goto 6,goto 3等语句，因此这些jump语句的目标语句前必须含有label, 这样划分后有9个block.**

1 m <- 0 **(block 1)**

2 v <- 0

3 if v >= n goto 15 (block 2)

4 r <- v (block 3)

5 s <- 0

6 if r < n goto 9 **(block 4)**

7 v <- v +1 (block 5)

8 goto 3

9 x <- M[r] （block 6）

10 s <- s + x

11 if s<= m goto 13

12 m <- s **(block 7)**

13 r <- r + 1 (block8)

14 goto 6

15 return m **(block 9)**

**8.7 Tree语句表示如下(省略block末尾多余的Jump语句)：**

**3个trace: 1->2->3->4->5 6->7->8 9**

**1、Label(Begin)**

Move (Temp m, Const 0)

Move (Temp v, Const 0)

**2、Label(TestConditionOfV)**

CJump( >=, v , n, Exit, BodyOfV)

**3、Label(BodyOfV)**

Move ( Temp r , Temp v)

Move ( Temp s, Const 0)

**4、Label(TestConditionOfR)**

CJump( <, r, n, BodyOfR,IncreaseV)

**5、Label(IncreaseV)**

Move (Temp v, Binop(+,Temp v, Const 1))

Jump(TestConditionOfV)

**6、Label(BodyOfR)**

Move ( Temp x, Mem(Temp r))

Move ( Temp s, Binop( +, Temp s, Temp x))

Cjump( <= , s, m, IncreaseR, DecreaseM)

**7、Label(DecreaseM)**

Move (Temp m, Temp s)

**8、Label(IncreaseR)**

Move (Temp r, Binop ( + ,Temp r, Const 1))

Jump(TestConditionOfR)

**9、Lable(Exit)**

Return m