

8.4.1: 图 8-10 是一个简单的矩阵乘法程序

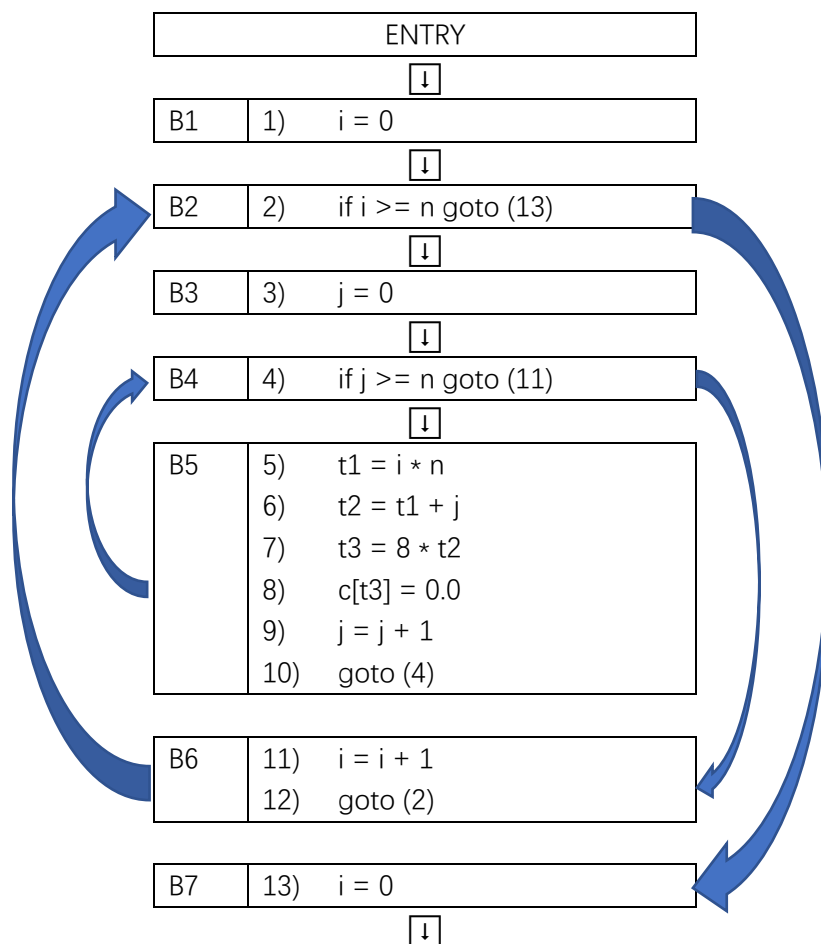
```
for (i=0; i<n; i++)
  for (j=0; j<n; j++)
    c[i][j] = 0.0;
for (i=0; i<n; i++)
  for (j=0; j<n; j++)
    for (k=0; k<n; k++)
      c[i][j] = c[i][j] + a[i][k]*b[k][j];
```

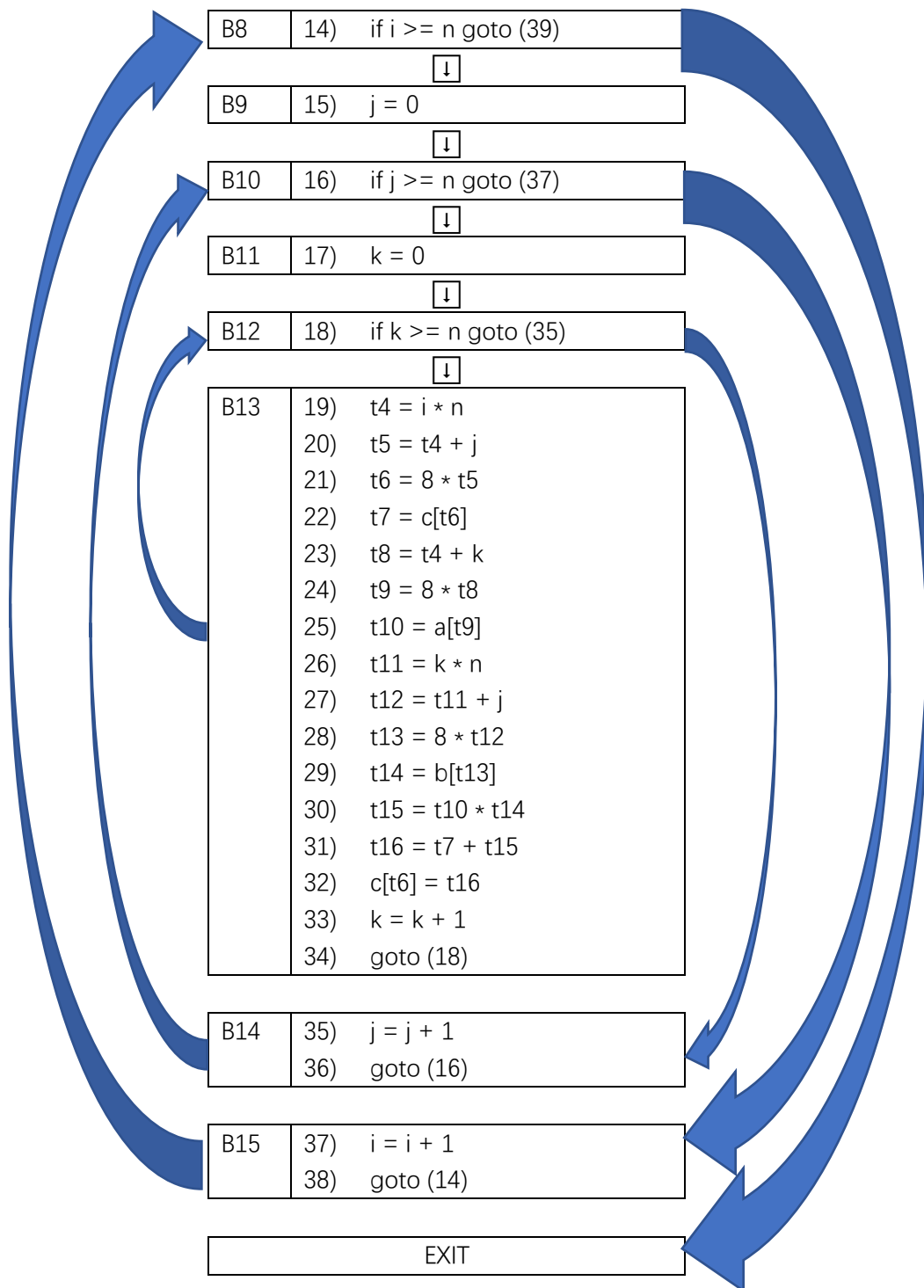
图 8-10 一个矩阵相乘算法

- 1) 假设矩阵的元素是需要 8 个字节的数值, 而且矩阵按行存放。把程序翻译成我们在本节中一直使用的那种三地址语句
- 2) 为 1) 中得到的代码构造流图
- 3) 找出在 2) 中得到的流图的循环

答:

1) & 2)





3) 循环:

{B4 , B5}

{B2 , B3 , B4 , B5 , B6}

{B12 , B13}

{B10 , B11 , B12 , B13 , B14}

{B8 , B9 , B10 , B11 , B12 , B13 , B14 , B15}

8.5.1 & 2: 为下面的基本块构造 DAG，并假设只有 a 在基本块出口活跃，简化下述三地址代码

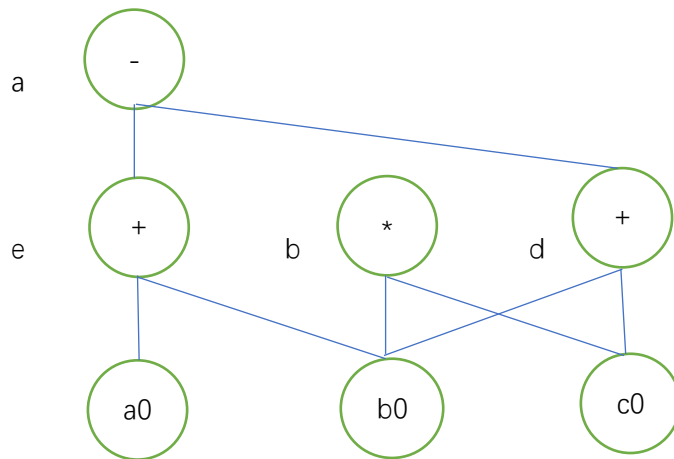
$d = b + c$

$e = a + b$

$b = b * c$

$a = e - d$

答:



简化:

$d = b + c$

$e = a + b$

$a = e - d$

删除 $b = b * c$