高等计算机体系结构，2020年春季

作业4： 流水线2

主讲教师：栾钟治

助讲教师：杨海龙；助教：孙庆骁

作业下发时间：2020年4月24日

作业回收时间：2020年5月8日

# 分支预测 35分

考察以下高级语言代码段:

int array[1000] = { /\* random values \*/ };

int sum1 = 0, sum2 = 0, sum3 = 0, sum4 = 0;

for (i = 0; i < 1000; i ++) // 分支 1: 循环分支

{

if (i % 4 == 0) // 分支 2: If 条件分支 1

sum1 += array[i]; // 发生分支的路径

else

sum2 += array[i]; // 不发生分支的路径

if (i % 2 == 0) // 分支 3: If 条件分支 2

sum3 += array[i]; // 发生分支的路径

else

sum4 += array[i]; // 不发生分支的路径

}

(a) 当使用last-time预测器时三个分支的预测准确率分别是多少？(假设每一个分支的last-time计数器起始状态是‘不发生’) 请写出你的计算过程和依据。

|  |  |
| --- | --- |
| **分支1：循环分支** | **循环判断语句一共执行了1001次（0-999正确执行，1000停止执行），其真实的分支序列为：TTTT…(1000个T)F。所以使用last-time计数器时，第一次和最后一次预测错误，因此准确率为：999/1001 = 99.9%** |
| **分支2：If条件分支1** | **If条件分支1一共执行了1000次，其真实的分支序列为：TFFF TFFF(250个TFFF)。所以使用last-time计数器时，预测的分支序列为：FTFF FTFF(250个FTFF)，故预测准确率为：500/1000 = 50%** |
| **分支3：If条件分支2** | **If条件分支2一共执行了1000次，其真实的分支序列为：TFTF TFTF(250个TFTF)。所以使用last-time计数器时，预测的分支序列为：FTFT FTFT(250个FTFT)，故预测准确率为：0/1000 = 0%** |

(b) 当使用基于2-bit饱和计数器的预测器时三个分支的预测准确率分别是多少？(假设每一个分支的2-bit计数器起始状态是‘强不发生’) 请写出你的计算过程和依据。

|  |  |
| --- | --- |
| **分支1：循环分支** | **循环判断语句一共执行了1001次（0-999正确执行，1000停止执行），其真实的分支序列为：TTTT…(1000个T)F。所以使用2-bit饱和计数器时，第一次、第二次和最后一次预测错误，因此准确率为：998/1001 = 99.7%** |
| **分支2：If条件分支1** | **If条件分支1一共执行了1000次，其真实的分支序列为：TFFF TFFF(250个TFFF)。所以使用2-bit饱和计数器时，预测的分支序列为：FFFF FFFF(250个FFFF)，故预测准确率为：750/1000 = 75%** |
| **分支3：If条件分支2** | **If条件分支2一共执行了1000次，其真实的分支序列为：TFTF TFTF(250个TFTF)。所以使用2-bit饱和计数器时，预测的分支序列为：FFFF FFFF(250个FFFF)，故预测准确率为：500/1000 = 50%** |

(c) 当分支2和分支3的2-bit计数器起始状态分别是 (i)‘弱不发生’；(ii)‘弱发生’ 时，预测准确率分别是多少？请写出你的计算过程和依据。

**a) 弱不发生**

|  |  |
| --- | --- |
| **分支2：If条件分支1** | **If条件分支1一共执行了1000次，其真实的分支序列为：TFFF TFFF(250个TFFF)。所以使用last-time计数器时，预测的分支序列为：FTFF FFFF(249个FFFF)，故预测准确率为：749/1000 = 74.9%** |
| **分支3：If条件分支2** | **If条件分支2一共执行了1000次，其真实的分支序列为：TFTF TFTF(250个TFTF)。所以使用last-time计数器时，预测的分支序列为：FTFT FTFT(250个FTFT)，故预测准确率为：0/1000 = 0%** |

**b)弱发生**

|  |  |
| --- | --- |
| **分支2：If条件分支1** | **If条件分支1一共执行了1000次，其真实的分支序列为：TFFF TFFF(250个TFFF)。所以使用last-time计数器时，预测的分支序列为：TTTF FFFF(249个FFFF)，故预测准确率为：749/1000 = 74.9%** |
| **分支3：If条件分支2** | **If条件分支2一共执行了1000次，其真实的分支序列为：TFTF TFTF(250个TFTF)。所以使用last-time计数器时，预测的分支序列为：TTTT TTTT(250个TTTT)，故预测准确率为：500/1000 = 50%** |

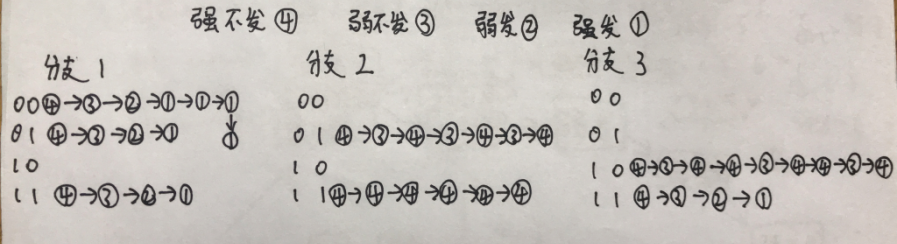
(d) 当使用两层全局历史分支预测器（2bit全局历史寄存器+每分支一张独立的模式历史表，模式历史表的每个表项是一个2-bit饱和计数器）时，三个分支的预测准确率分别是多少？假设全局历史寄存器每一位的初始状态都是‘不发生’，模式历史表中的2-bit饱和计数器初始状态都是‘强不发生’，计算预测准确率时，忽略前500次循环迭代。

**每4次循环过程如下：**

**i=0：111 三个分支都发生  
i=1：100 分支1发生  
i=2：101 分支13发生  
i=3：100 分支1发生**

**GHR序列应为：00 111 100 101 100 (以111 100 101 100循环)**

**画出前三轮（111 100 101 100循环）后三个分支的模式历史表变化情况。**

****

|  |  |
| --- | --- |
| **分支1：循环分支** | **循环判断语句一共执行了1001次（0-999正确执行，1000停止执行），其真实的分支序列为：TTTT…(1000个T)F。所以可以看到上面分支1的模式历史表中三个状态都为强发生，因此准确率为：500/501 = 99.8%** |
| **分支2：If条件分支1** | **If条件分支1一共执行了1000次，其真实的分支序列为：TFFF TFFF(250个TFFF)。所以可以看到分支2的模式历史表中都为预测不发生，在每4次分支预测中只有01状态预测中出现预测错误“强不发生->弱不发生”，11状态预测一直为强不发生，故预测准确率为：375/500 = 75%** |
| **分支3：If条件分支2** | **If条件分支2一共执行了1000次，其真实的分支序列为：TFTF TFTF(250个TFTF)。查看上面分支3的模式历史表可以看到11状态预测为强发生，10状态以“强不发生->弱不发生->强发生”循环，在每4次循环中只有“强不发生->弱不发生”一次预测错误，因此预测准确率为：375/500 = 75%** |

# 两层分支预测器 25分

假设一个两层全局预测器由全局历史寄存器和所有分支共享的一张模式历史表组成(称为预测器 A)

1）我们把分支预测器中不同的分支映射到相同位置的情况称为"分支干扰"。预测器A的结构中，不同分支会在哪里发生这种干扰?

**答：假如全局状态寄存器有n位，当分支i预测时候前n次全局分支执行的真实序列和分支j预测时候的前n次全局分支执行的真实序列相同时，会发生分支干扰。**

**例如：以上一题为例n=2,** **全局分支执行的真实序列为111 100 101 100循环，假如分支共享同一张模式历史表，那么111 100、111 100、111 100三次分支预测都会进到模式历史表同一个项中发生干扰。**

2）另一个两层全局预测器由全局历史寄存器和每个分支一张模式历史表组成 (称为预测器 B"),

(a) 什么情况下预测器A的预测准确率低于预测器B? 请解释理由，并举例说明。可以通过代码来说明。

**答：当发生分支干扰的时候，会使得预测器A的预测准确率低于预测器B。**

**以题目1中的代码为例，预测器A的预测准确率就低于预测器B。（假设GHR为2位）**

**GHR序列应为：00 111 100 101 100 (以111 100 101 100循环)**

**00：强不发生🡪弱不发生🡪弱发生**

**01：强不发生🡪弱不发生🡪强不发生**

**10：强不发生🡪 强不发生🡪弱不发生🡪强不发生**

**11：强不发生🡪弱不发生🡪弱发生🡪弱不发生🡪强不发生**

**上面为第一轮模式历史表结果，可以发现00最后状态为强发生，而01、10、11则是上面状态循环， 以第三个分支为例，预测器A的4次预测中准确率为50%低于预测器B的结果75％。**

(b) 预测器A能获得比预测器B更高的预测准确率吗? 请解释理由，并举例说明。可以通过代码来说明。

**答：不能。因为预测器A会经常发生分支干扰，而全局历史寄存器的优点在于记住历史过程中相似的预测序列从而进到历史的预测结果中，对于预测分支而言，假如记住历史上相似属性特点被破坏，分支预测结果肯定是不正确的，准确率势必会下降。预测器B不会受到分支干扰的影响，预测结果更加准确。**

(c) 分支干扰是否总是影响预测器的预测准确率? 请解释理由，并举例说明。可以通过代码来说明。

**答：不一定。假如在某程序中的所有分支预测都为不发生，则无论是否存在分支干扰，都不影响准确率。**

**例如：**

**i=0;**

**if (i>101)**

**i++;**

**else**

**i++;**

**if (i>102)**

**i++;**

**else**

**i++;**

**…**

**if (i>200)**

**i++;**

**else**

**i++;**

# 分支预测和推断 30分

考察两台具有15段流水线的机器A和B，流水段分布如下：

取指 (1个阶段)

译码 (8个阶段)

执行 (5个阶段)

写回 (1个阶段)

两台机器都会在发生流相关时采用数据转发。在译码的最后一个阶段检测是否有流相关，指令会在停顿在这个阶段等待检测结果。

机器A有一个预测准确率为P%的分支预测器，分支方向和目标在执行的最后一个阶段产生。

机器B采用推断执行，类似咱们在课堂上讲的方式。

1）考察以下在机器A上执行的代码段:

Add r3 r1, r2

sub r5 r6, r7

beq r3, r5, X

addi r10 r1, 5

add r12 r7, r2

add r1 r11, r9

X: addi r15 r2, 10

.....

我们把这段代码转化为在机器B上执行的推断代码，大概是下面这个样子:

add r3 r1, r2

sub r5 r6, r7

cmp r3, r5

addi.ne r10 r1, 5

add.ne r12 r7, r2

add.ne r14 r11, r9

addi r15 r2, 10

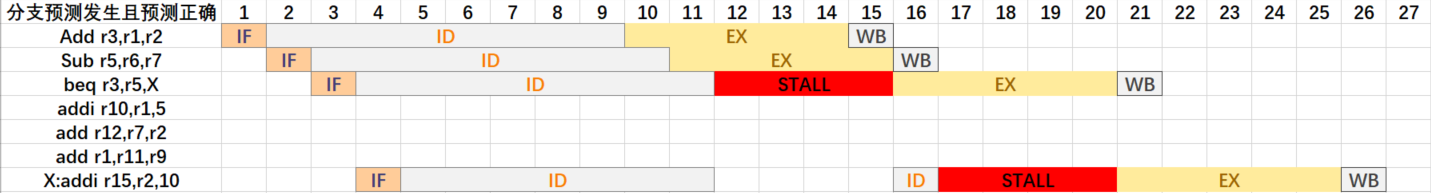
.....

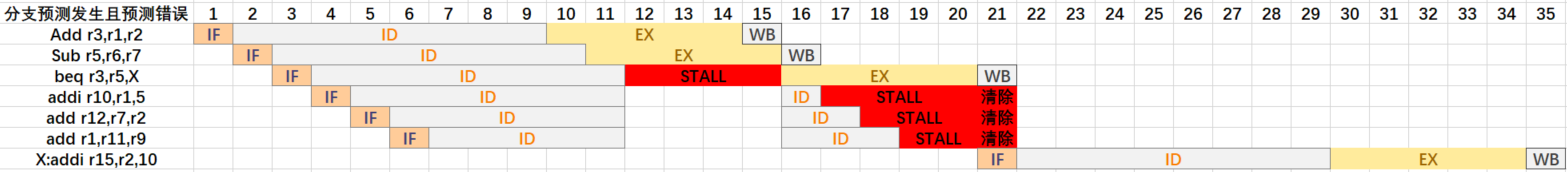
(假设条件结果由‘cmp’指令计算，推断由‘.ne’指令根据条件结果执行。条件结果在执行的最后一个阶段计算并且可以像其他值一样被转发)

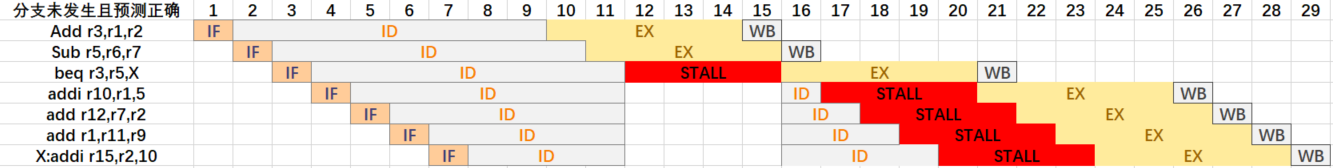
这一段代码会重复执行上百次，分支40%可能性发生，60%可能性不发生，平均而言，P取什么样的值会使机器A比机器B具有更高的指令吞吐量?

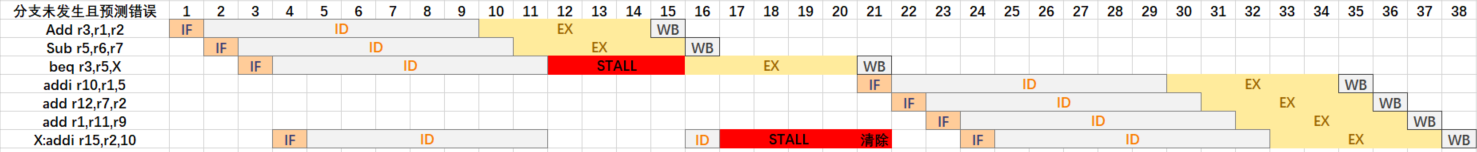
**答：**

**机器A为：**



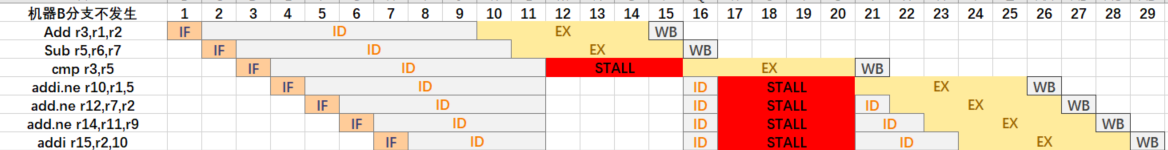


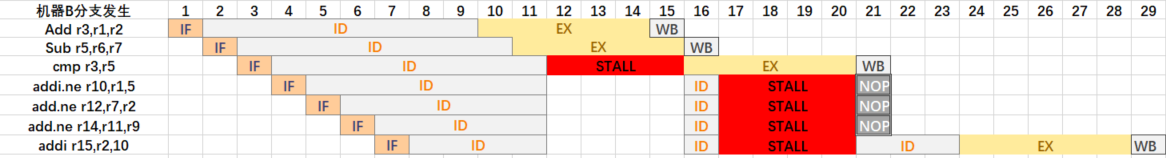




**平均周期为：26\*0.4\*P+35\*0.4\*(1-P)+29\*0.6\*P+38\*0.6\*(1-P) = 36.8-9P**

**机器B为：**





**因此当36.8-9P<29，即P>86.6%时，机器A的吞吐量大于机器B。**

2）考察在机器A上执行的另一段代码:

add r3 r1, r2

sub r5 r6, r7

beq r3, r5, X

addi r10 r1, 5

add r12 r10, r2

add r14 r12, r9

X: addi r15 r14, 10

.....

我们把这段代码转化为在机器B上执行的推断代码，大概是下面这个样子:

add r3 r1, r2

sub r5 r6, r7

cmp r3, r5

addi.ne r10 r1, 5

add.ne r12 r10, r2

add.ne r14 r12, r9

addi r15 r14, 10

.....

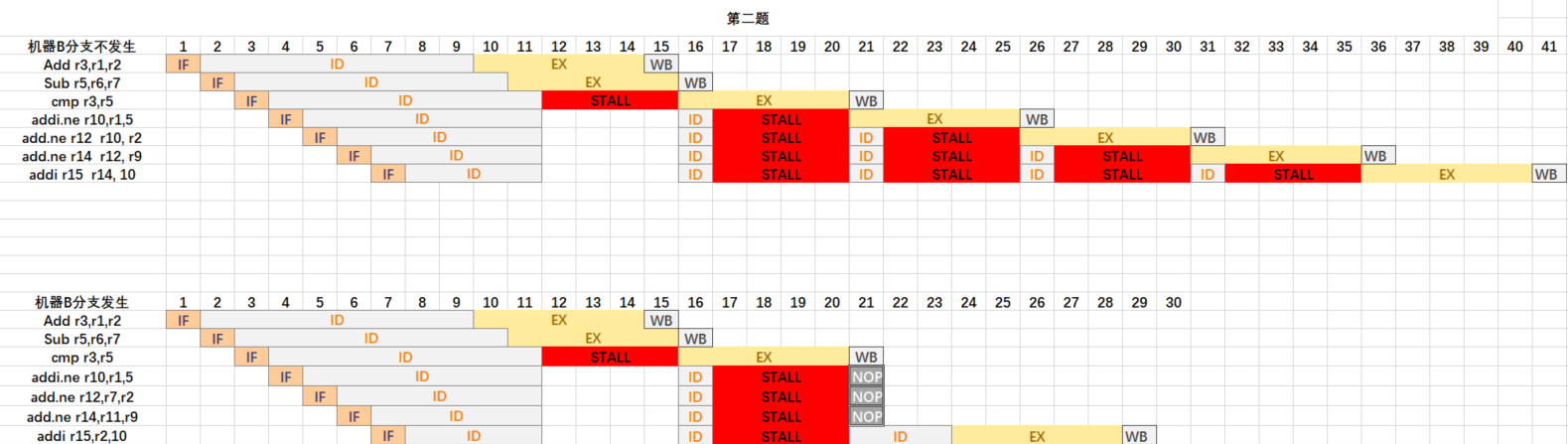
(假设条件结果由‘cmp’指令计算，推断由‘.ne’指令根据条件结果执行。条件结果在执行的最后一个阶段计算并且可以像其他值一样被转发)

这一段代码会重复执行上百次，分支40%可能性发生，60%可能性不发生，平均而言，P取什么样的值会使机器A比机器B具有更高的指令吞吐量?

**答：相比第一题后四条指令增加了流相关，后三条指令执行都得等前一条指令执行完毕才能执行，因此在分支不发生的情况下，需要额外等待15-3(3条指令)=12个周期。**

**所以机器A分支发生且预测正确的时钟周期为：26；分支发生且预测错误时钟周期为：36；分支未发生且预测正确时钟周期为：29+12=41；分支未发生且预测错误时钟周期为：38+12=50**

**机器B分支不发生29+12=41，分支发生29**

****

**机器A平均周期为：26\*0.4\*P+35\*0.4\*(1-P)+41\*0.6\*P+50\*0.6\*(1-P) =44-9P**

**机器B平均周期为：29\*0.4+41\*0.6=36.2**

**当44-9P<36.2,即P>86.7%时，机器A比机器B具有更高的指令吞吐量。**