

12.

1) 以下是与该汇编代码功能一致的 C 语言代码:

```
for(int a1=0; a1<10000; a1++){
    for(int a3=2; a3<a1+2; a3++){
        int a2 = a1 % a3;
        if(a2 == 0){
            //...CodeA
            goto Rem2;
        }
    }
    //...CodeB
Rem2:
}
```

2) 三条 bne 指令发生跳转的比例分别为:

B1: 每个 a1 的循环开头都会执行一次, 跳转概率为 1;

B2: 每个 a1 与 a3 的循环中最多会执行一次, 跳转概率为  $1/9999 \approx 0$ ;

B3: 总共会执行 9999 次, 跳转概率为 1。

3) 引入静态分支预测器后, 对于 B1 指令, 预测准确率为 100%; 对于 B2 指令, 预测准确率为 0%; 对于 B3 指令, 预测准确率为 0%。因为静态分支预测器总是预测向前跳转, 而 B2 和 B3 指令都是向后跳转。

13. (1) 5

2)  $N \geq 2$

3) B1:50% B2:80% B3:100%

14. H 的最小值为 4

15. M 的最小值为 5

16. 对于方案 A, 每次进入外循环时, 预测器的计数器状态被重置, 因此内循环中的分支指令对外层循环的预测不会产生影响, 可以看做内循环体中不存在分支指令。因此, 除了外层循环体之外, 预测器只需要预测跳转到外循环体的概率, 即预测  $P$  次跳转。因此, 方案 A 的预测准确率为:

方案 A 的预测准确率 =  $1 - P$

对于方案 B, 预测器表的索引是内循环体的迭代次数  $j$ , 预测器的计数器状态会一直保持直到下一次进入该迭代次数时被重置。因此, 在每次出现新的  $j$  值时, 预测器必须重新预测跳转与否的结果。如果内循环体中存在分支指令, 那么预测准确率将进一步降低。因此, 方案 B 的预测准确率为:

方案 B 的预测准确率 =  $1 - (Q - 1)/(2^H)$

将方案 A 的预测准确率大于方案 B 的预测准确率的条件代入, 可得:

$1 - P > 1 - (Q - 1)/(2^H)$

化简得:

$P < (Q - 1)/(2^H)$

因此, 方案 A 的预测准确率优于方案 B 的预测准确率的条件是:

$P < (Q - 1)/(2^H)$

即：

$$P < \{Q-1\}/\{2^{\{Q-1\}}\}$$

当 P 的取值大于该阈值时，方案 A 的预测准确率优于方案 B 的预测准确率。

17.(1)共错误 7 次

(2) 错误 9 次

(3) 错误 7 次

(4) 可以看出，在这种情况下，全局历史表的位数越多，预测准确率越高。这是因为随着历史表位数的增加，预测器能够记录更多的历史信息，因此在判断跳转方向时能够更准确地预测。当 n 非常大时，循环迭代次数很多，分支指令出现的相对频率较低，因此全局历史表的位数对预测准确率的影响会更加显著。

(5) 当数组 p[] 的数据模式变为在 0 和 1 之间以均等概率随机取值时，内部分支指令的跳转率为 0.5，所有分支指令的跳转率也为 0.5。对于 2 位的局部预测器，有 4 个预测器表项，分别对应如下四种状态：00，01，10，11。初始状态下，这些计数器的值均为 0。在这种情况下，内部分支指令的跳转方向与上次迭代的跳转方向无关，因此全局历史表对于预测的影响就比较小了。

18.在顺序的 5 级 RISC 流水线中，指令引发的异常也可能会乱序产生的原因是在流水线中，每个指令都在经过不同的流水线阶段后才被执行完成。如果在某个流水线阶段中出现了异常，那么这个指令就会被标记为异常，并在后续的流水线阶段中被丢弃。由于不同指令在不同的流水线阶段中，因此即使是在顺序的流水线中，指令引发的异常也有可能会乱序产生。为了支持精确的异常处理，处理器需要在流水线中实现 Exception Handling Unit (EHU) 等硬件机制来处理异常。

20.

	周期				操作码	目标	源 1	源 2
	Decode (ROB enqueue)	Issue	WB	Committed				
I1	0	1	2	3	fld	T0	a0	—
I2	1	3	13	14	fmul.d	T1	T0	f0
I3	2	14	16	17	fadd.d	T2	T1	F0
I4	3	4	5	18	addi	T3	A0	—
I5	4	5	6	19	fld	T4	T3	—
I6	5	14	24	25	fmul.d	T5	T4	T4
I7	6	25	27	28	fadd.d	T6	T5	T2
	周期				操作码	目标	源 1	源 2
	Decode (ROB enqueue)	Issue	WB	Committed				
I1	0	1	2	3	fld	T0	a0	—
I2	1	3	13	14	fmul.d	T1	T0	f0
I3	4	14	16	17	fadd.d	T2	T1	F0
I4	15	16	17	18	addi	T3	A0	—
I5	18	19	20	21	fld	T4	T3	—

I6	19	21	31	32	fmul.d	T5	T4	T4
I7	22	32	34	35	fadd.d	T6	T5	T2