

$$1. \rightarrow 5. CPI_A = 15\% \times (10\% \times 3 + 90\% \times 10\% \times 4) + 85\% = 0.949$$

$$CPI_B = 15\% \times 2 + 85\% = 1.15 \quad S = \frac{CPI_B}{CPI_A} = 1.212 \quad \text{降低了} 1.212 \text{倍。}$$

12. 1) int main()

int a₀ = 0;

int a₄ = 10000;

int a₁ = a₀

for (int i=0; i<a₄; i++)

int a₃ = a₀ + 2;

int a₂ = a₁ % a₃;

if (a₂ != 0)

11..- CodeA

} else

a₃ = a₀ + 5

a₂ = a₁ % a₃

if (rem != a₂) { 11..- CodeB }

11..- CodeB

}

a₁++

}

return 0;

}

→2) 从图中可知，
每处跳转都需要一处分支预测器。

13. 1) K 的最小值为 3。因为每处跳转都需要对应一处分支预测器。
2) 由于 12 中最低的预测准确率为 $\frac{1}{3}$ ，故从图中可知 K 的最小值为 2。

3) 在程序稳定时，

B_1 的预测准确率为 50%。
 B_2 为 80%。

B_3 为 99.99%。（ $(1 + 0.2 \times 0.001)^{10^6} \approx 1.0002$ ）

14. 假设局部被打破成有 2^K 个表项，每个表项的历史长度为 H 位。

则 $2^{K+H} \geq 3$ ，

故 $K+H$ 的最小值为 2

15. M 的最小值为 $9999 - 1 + 2000 = 8000 \approx 3998$

16. 方案 A 的准确率为 $\frac{PQ}{PQ} = \frac{Q(1+P-1)}{PQ} = \frac{Q}{PQ} = \frac{1}{PQ}$

对于方案 A，内存坏

16. 对于方案A的预测正确率:
outer loop: $\frac{P-2}{P}$
inner loop: $\frac{Q-2}{Q}$

对于方案B的预测正确率:
outer loop:

16. 对于方案A，每次进入outer loop都会更新计数器；而对于方案B，每次进入外循环都会更新局部分支历史。但进入内循环时不会。
因此，当P较小而Q较大时，方案A优于方案B。
 $P < 2^k$ 而 $Q \geq 2^k$ 时， $A \succ B$

17. 1) 对于B1：分支指令1次不跳转，1次不跳转，共有4次错误。

对于B2：7次跳转，1次不跳转，初始共4次错误，最后一次错误
共有3次错误

2) B1：第一次预测正确， $GHR=0, PHT=00$ 。

第二次， $GHR=1, PHT=00$ ，预测错误， PHT 变为1

第三次~第八次， $GHR=1$

故共有3次预测错误

B2：因为 GHR 对应2个 PHT ，故需要2次初始化，共错误4次。

最后一条分支指令也会输出共错误5次。

则一共会发5次错误

3) 对于B1：

1 00 00 ✓

2 01 00 ✗

3 11 00 ✓

4 01 01 ✗

5	11	00	✓
6	01	10	✓
7	11	00	✓
8	01	11	✓

错误2次

对于B2:

1	00	00	✗
2	11	00	✗
3	10	00	✗
4	11	00	✗
5	10	01	✗
6	11	00	✓

错误6次

故一共8次错误

所以GHR位数增加会减少和B1一样结果变化频繁的指令的错误率，增加和B2一样结果稳定的指令初始化阶段的优势。

当n非常大时，2位全局分支表的表现最好
因此时GHR丧失优势。故少用GHR的方式表现最好

18. ①因为指令可能在不同流水之间出现异常。

②流水线会在写回阶段后再加一个提交阶段，使用ROB记录指令的顺序，并按顺序处理异常和提交。

20. 1)

周期

Decade	Issue	WB	Committed	操作	目标	源 源2
I1	0	1	2	fld	T0	a0
I2	1	3	13	fmul.d	T1	T0 f0
I3	2	14	16	fadd.d	T2	T1 f0
I4	3	4	6	addi		a0
I5	4	5	7	fld	T3	a0
I6	5	8	18	fmul.d	T4	T3 T3
I7	6	19	21	fadd.d	T5	T4 T2
<hr/>						

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	18	19	addi	a0	a0
I5	18	19	20	21	fld	T3	a0
I6	20	21	31	32	fmul.d	T4	T3 T3
I7	22	32	34	35	fadd.d	T5	T4 T2