

第五讲：最少操作次数-HZOJ-511

胡船长

初航我带你，远航靠自己

一、贪心策略

局部：

ans 代表最少操作次数，则 ans 更新策略如下：

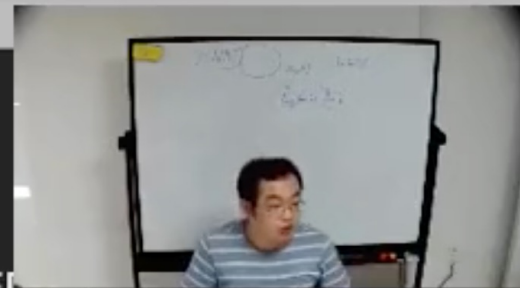
情况1：若 $a*k \leq b$ ， $ans += 1 + b\%k$ ， $b /= k$

情况2：若 $a*k > b$ ， $ans += (b - a)$ ，终止

整体：

按照如上策略执行，最终得到的 ans，就是最少操作次数

```
vim %1 bash %2 bash %3
39 }
40
41 Node *insert_maintain(Node *root) {
42     if (!hasRedChild(root)) return root;
43     if (root->lchild->color == RED && root->rchild->color == RED, {
44         if (!hasRedChild(root->lchild) && !hasRedChild(root->rchild)) return root;
45         root->color = RED;
46         root->lchild->color = root->rchild->color = BLACK;
47         return root;
48     }
49     if (root->lchild->color == RED) {
50         if (!hasRedChild(root->lchild)) return root;
51
52
53     } else {
54         if (!hasRedChild(root->rchild)) return root;
55
56     }
57
58 }
```



最少操作次数-HZOJ-511: 代码演示

```
61 Node *__insert(Node *root, int key) {
62     if (root == NIL) return getNewNode(key);
```

二、基础知识：进制的理解

数字：5

二、基础知识：进制的理解

数字：5

二进制

	1	0	1
--	---	---	---

二、基础知识：进制的理解

数字：5

二进制

	1	0	1
--	---	---	---

十进制

			5
--	--	--	---

二、基础知识：进制的理解

数字：5

二进制

	1	0	1
--	---	---	---

十进制

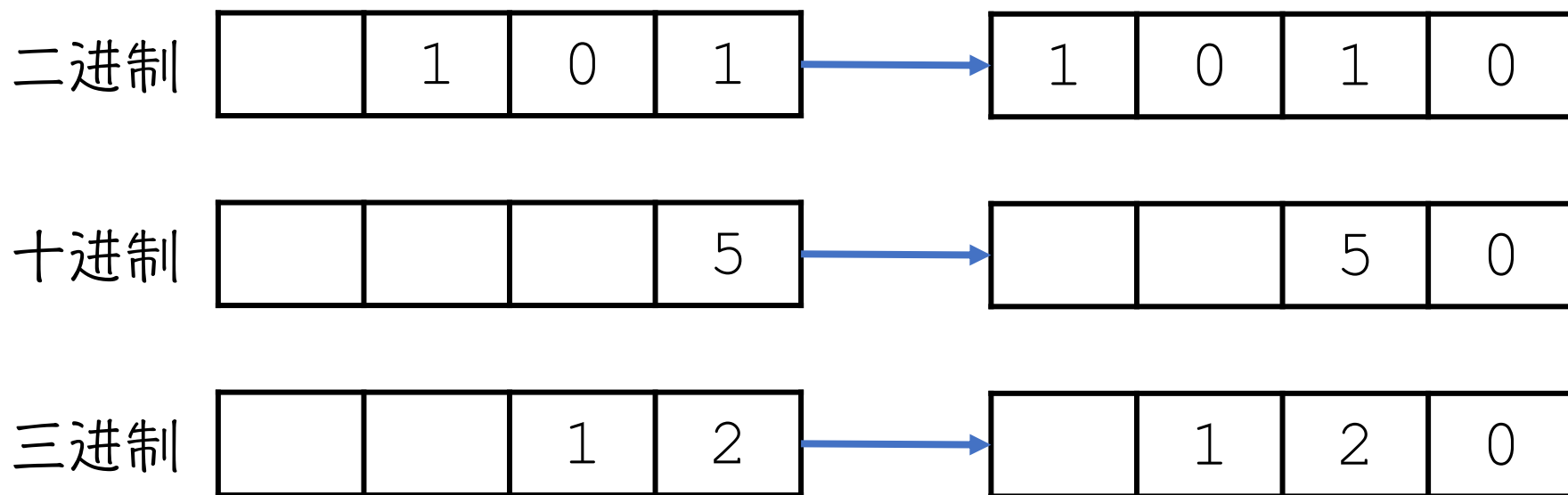
			5
--	--	--	---

三进制

		1	2
--	--	---	---

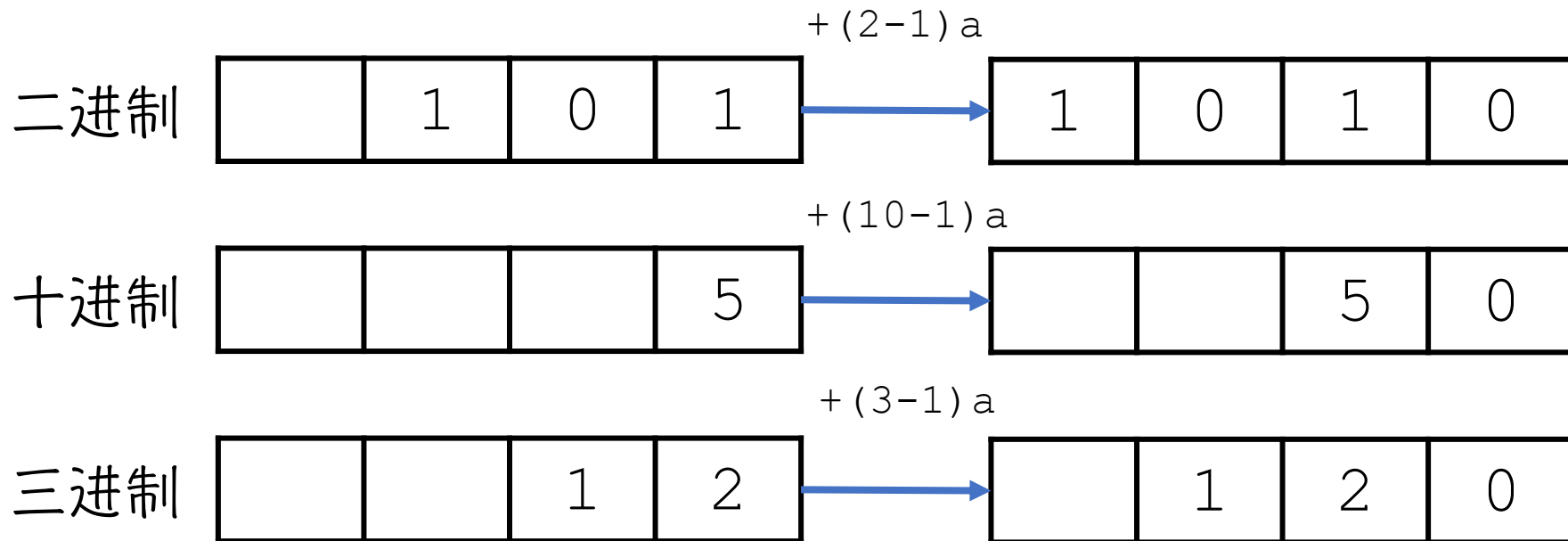
二、基础知识：进制的理解

乘以 k ，就是在 k 进制表示后加个0



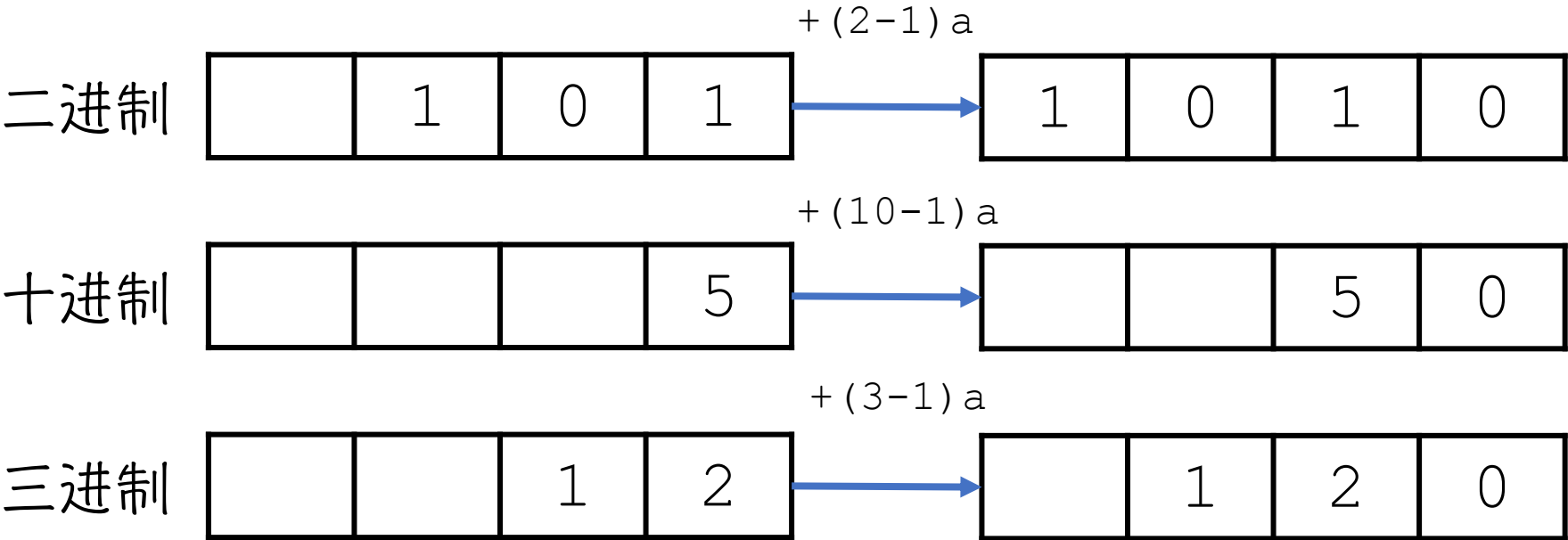
二、基础知识：进制的理解

+1 等价于乘以 k 的效果，需要加 $(k-1)a$



二、基础知识：进制的理解

结论1：使用 *k 操作，优先于 +1 操作



三、最优操作顺序

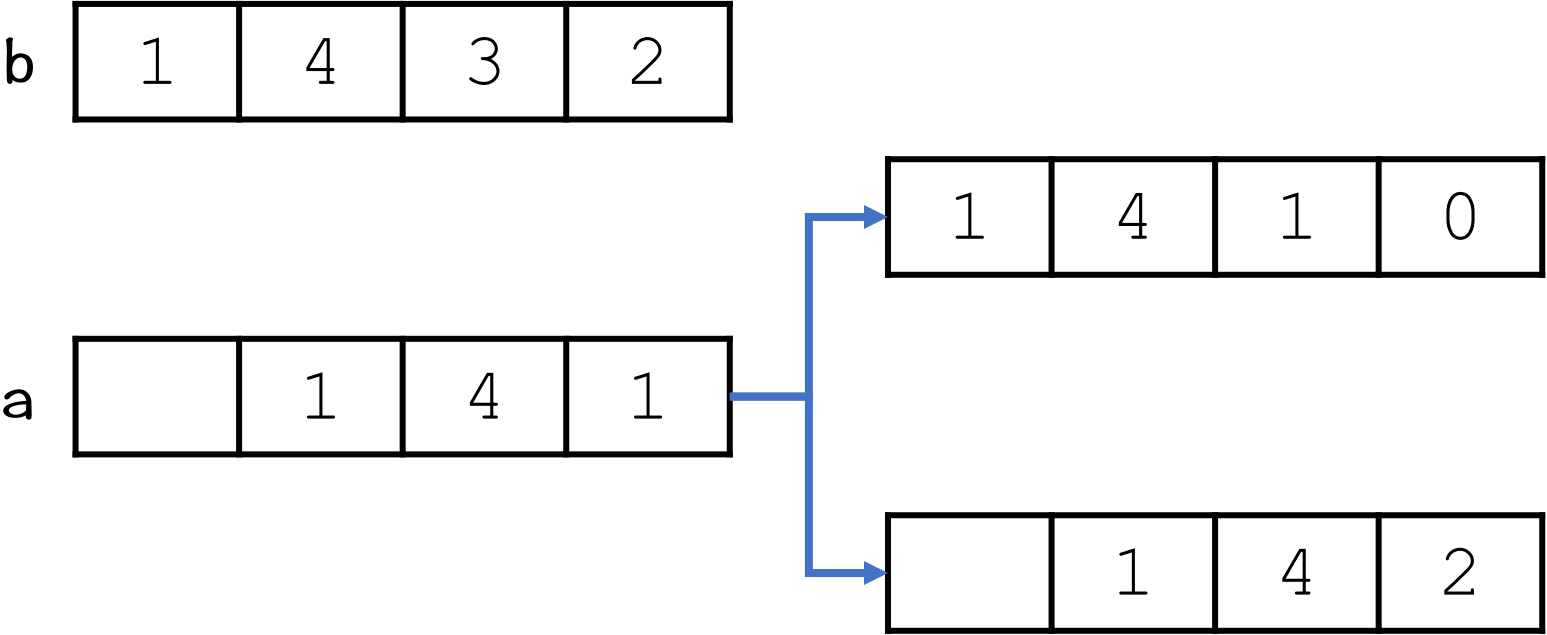
b

1	4	3	2
---	---	---	---

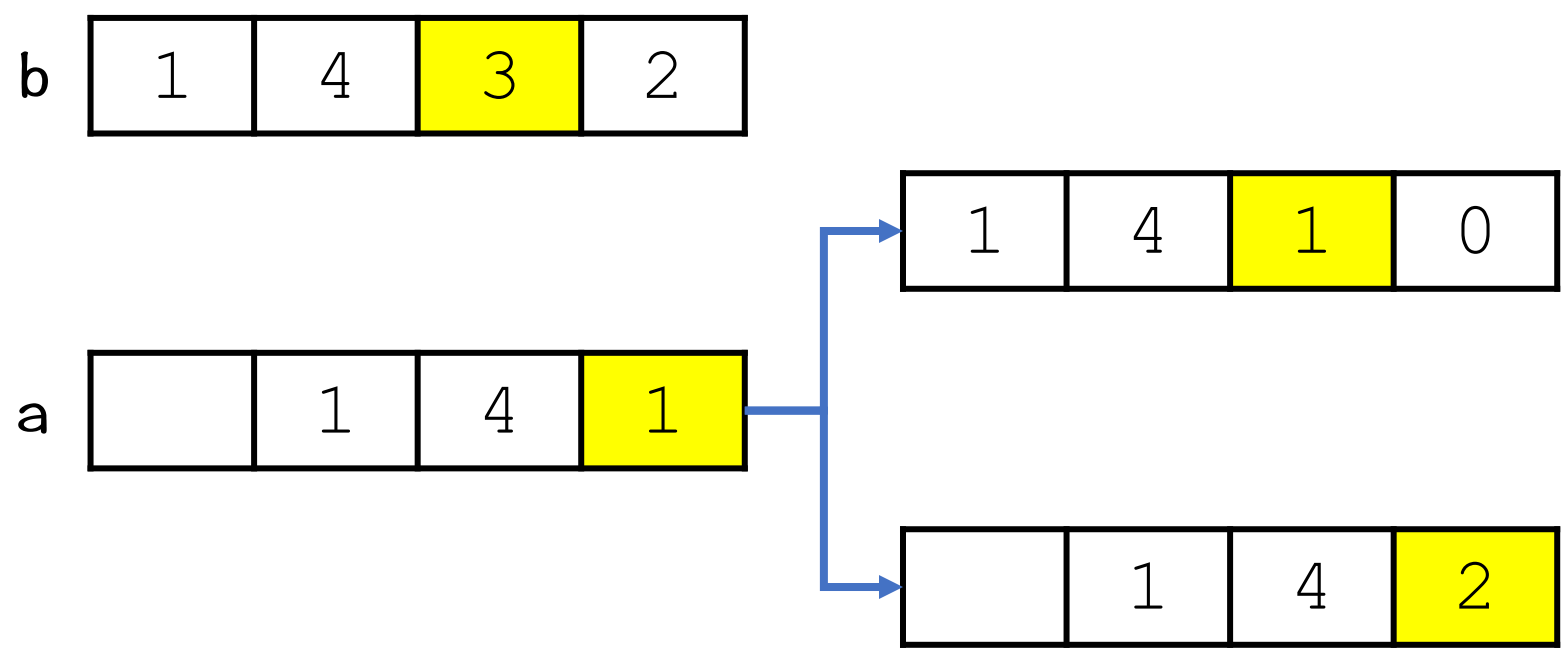
a

	1	4	1
--	---	---	---

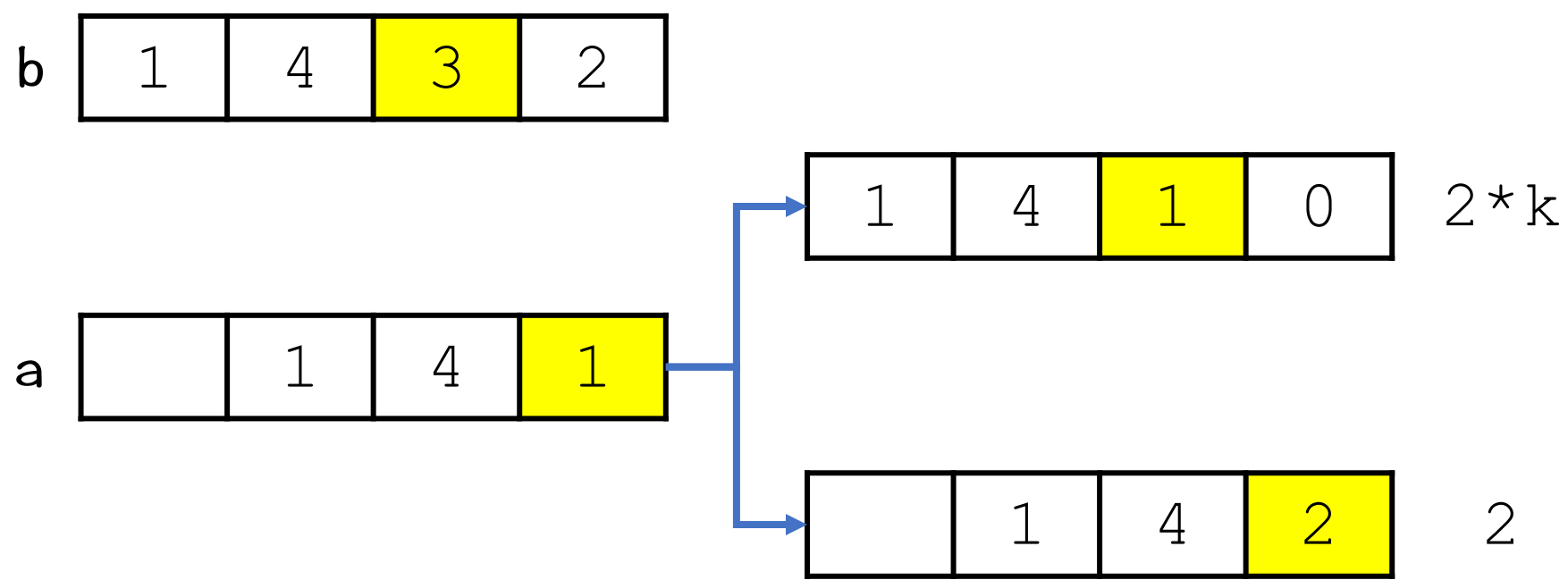
三、最优操作顺序



三、最优操作顺序

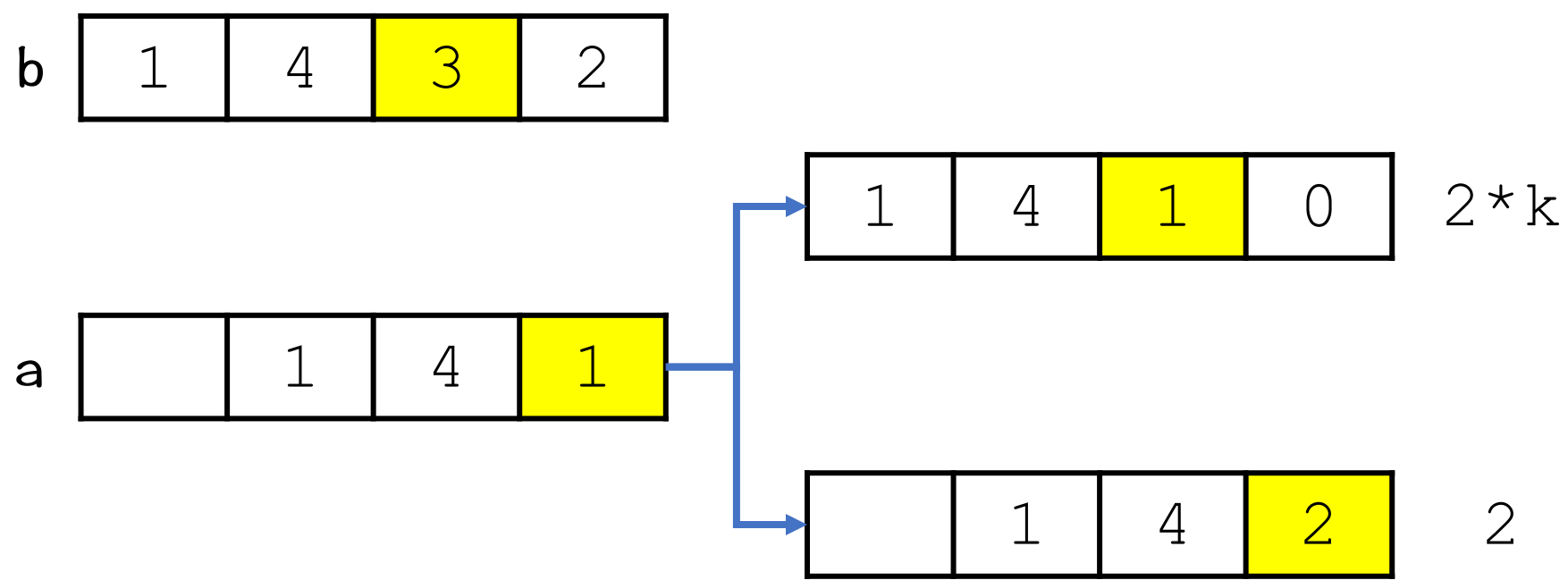


三、最优操作顺序



三、最优操作顺序

结论2：当 a 为 b 的数字前缀时，再使用 *k 操作



四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作

b

1	4	3	2
---	---	---	---

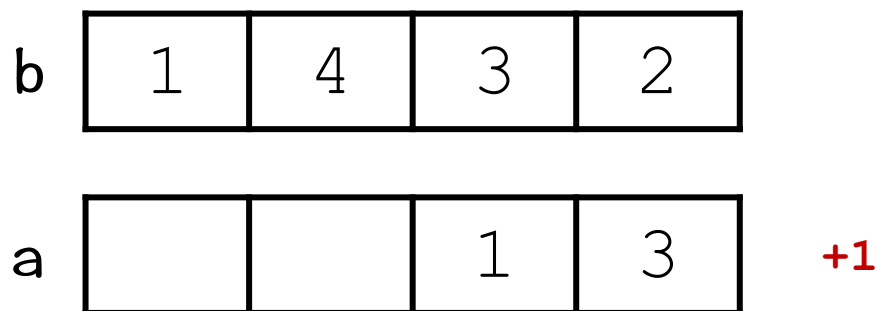
a

		1	2
--	--	---	---

四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作



四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作

b

1	4	3	2
---	---	---	---

a

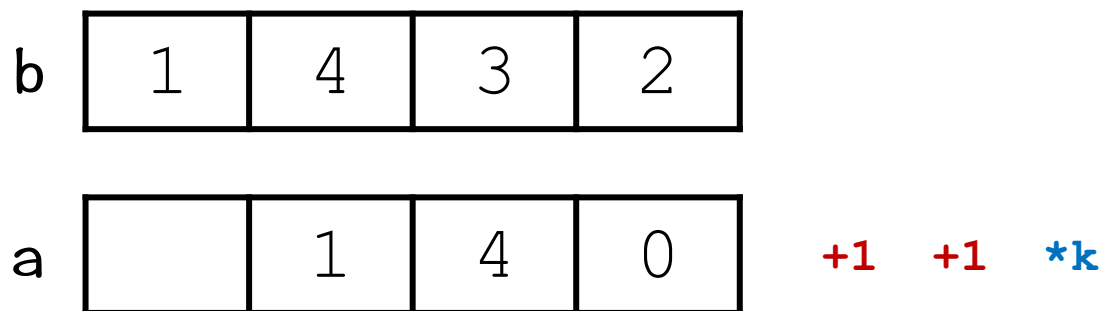
		1	4
--	--	---	---

+1 +1

四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

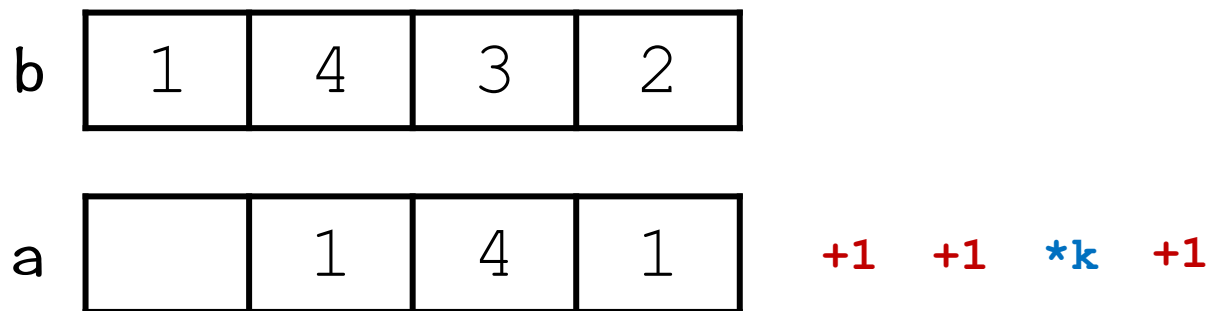
结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作



四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

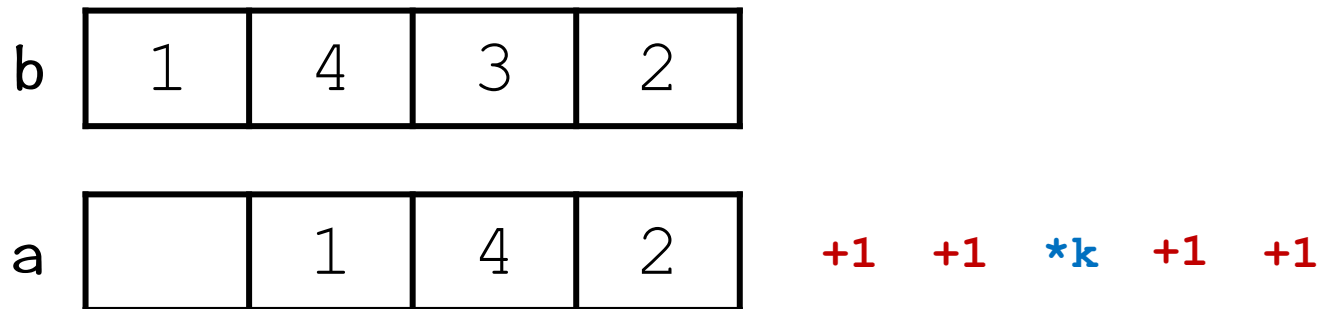
结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作



四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

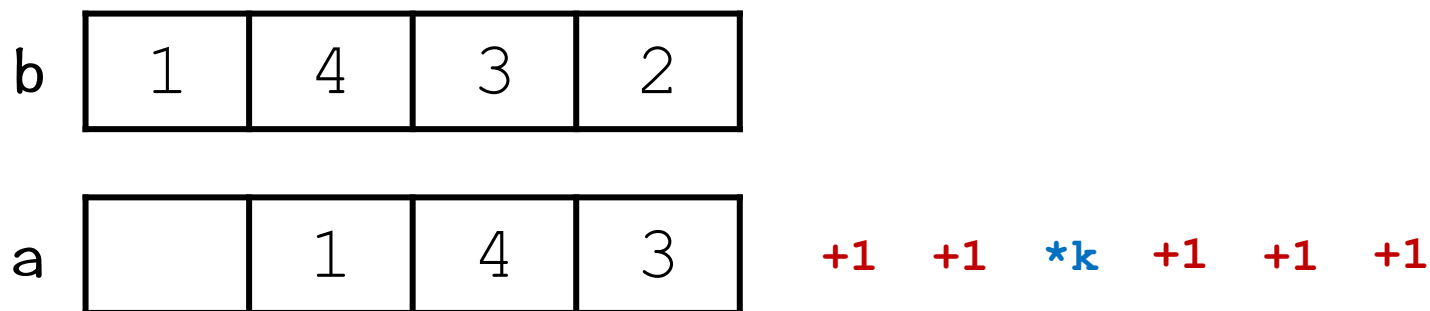
结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作



四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作



四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作

b

1	4	3	2
---	---	---	---

a

1	4	3	0
---	---	---	---

$+1$ $+1$ $\times k$ $+1$ $+1$ $+1$ $\times k$

四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作

b

1	4	3	2
---	---	---	---

a

1	4	3	1
---	---	---	---

$+1$ $+1$ $\times k$ $+1$ $+1$ $+1$ $\times k$ $+1$

四、综合理解

结论1：使用 $\times k$ 操作，优先于 $+1$ 操作

结论2：当 a 为 b 的数字前缀时，再使用 $\times k$ 操作

b

1	4	3	2
---	---	---	---

a

1	4	3	2
---	---	---	---

+1 +1 $\times k$ +1 +1 +1 $\times k$ +1 +1

四、综合理解

局部：

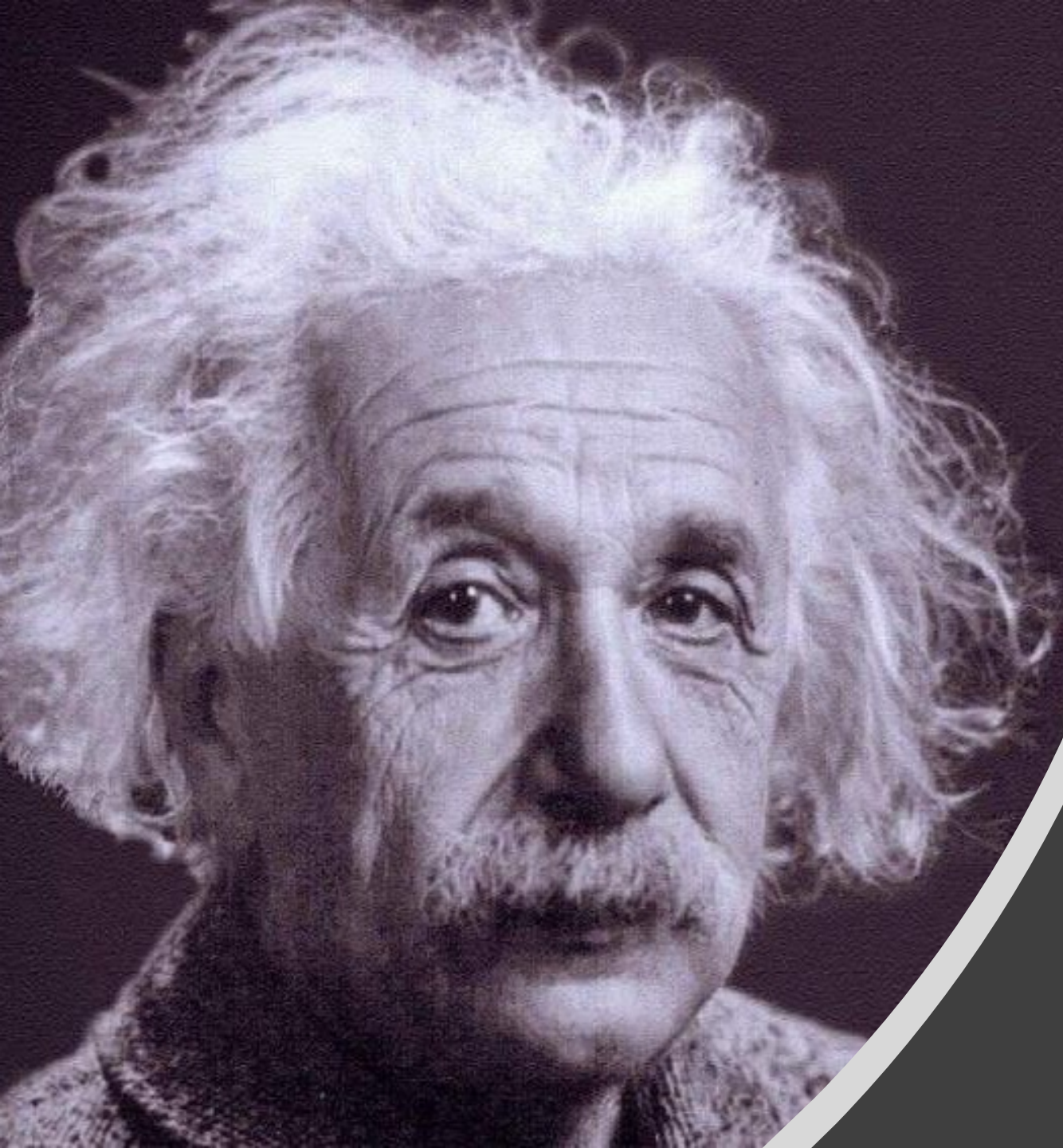
ans 代表最少操作次数，则 ans 更新策略如下：

情况1：若 $a*k \leq b$ ， $ans+=1+b\%k$ ， $b/=k$

情况2：若 $a*k > b$ ， $ans+=(b-a)$ ，终止

b	1	4	3	2
a			1	2

b-a		1+b%k				1+b%k		
+1	+1	*k	+1	+1	+1	*k	+1	+1



为什么
会出一样的题目？