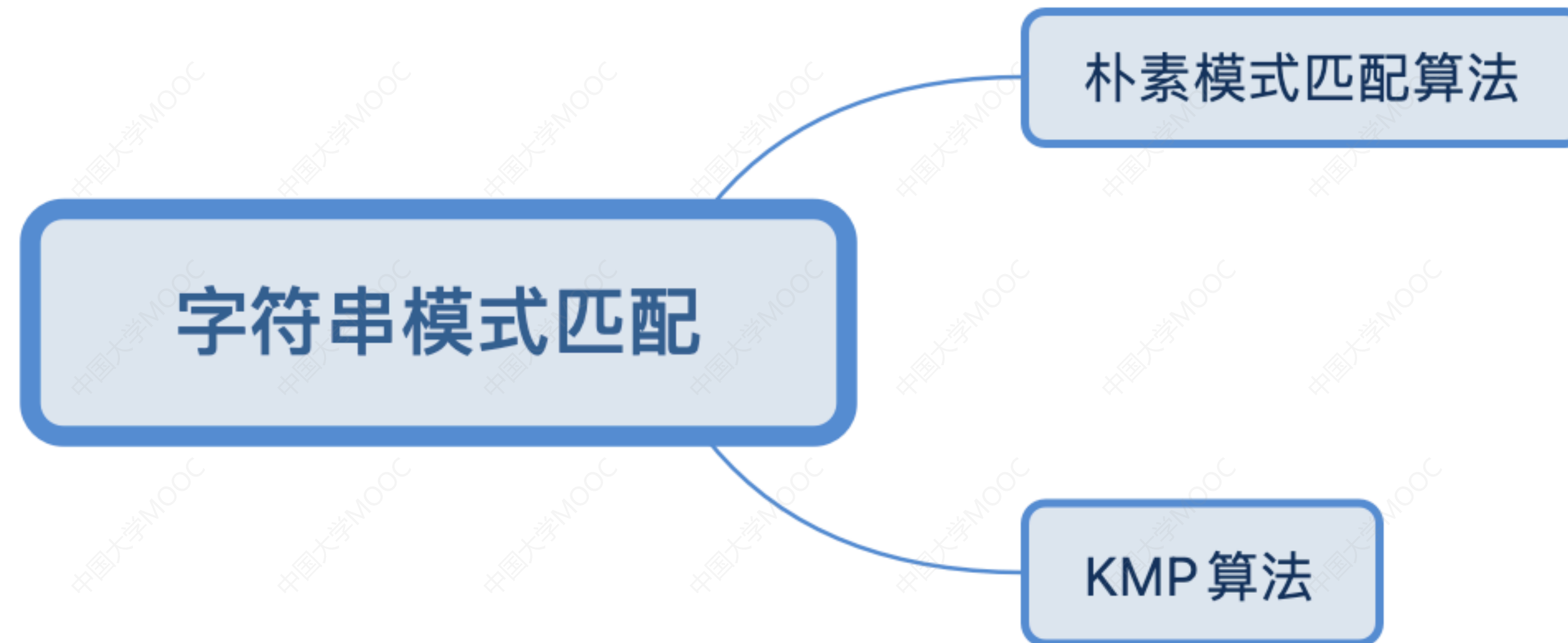


本节内容

字符串

KMP算法

两种模式匹配算法



KMP算法



由D.E.Knuth, J.H.Morris和V.R.Pratt提出，因此称为 **KMP**算法



你不要凶我 我害怕。

朴素模式匹配算法



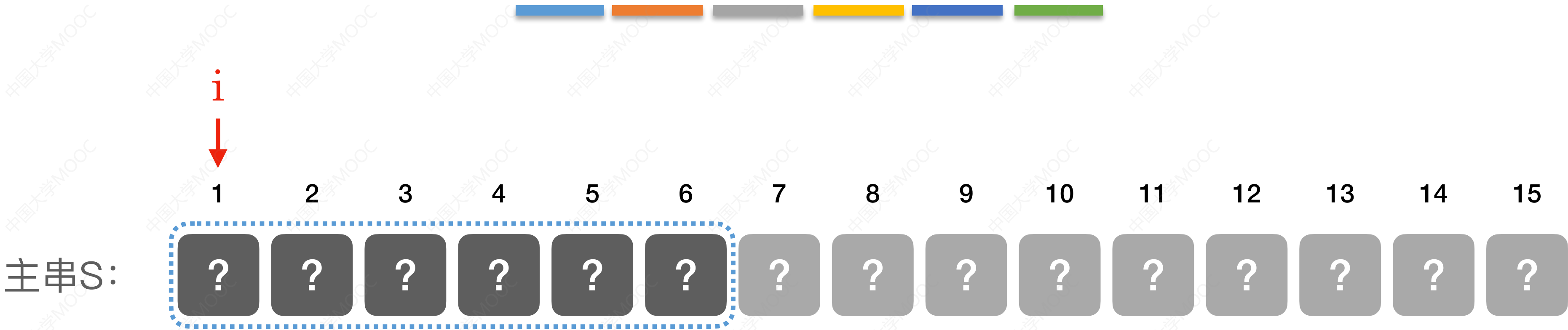
主串S:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

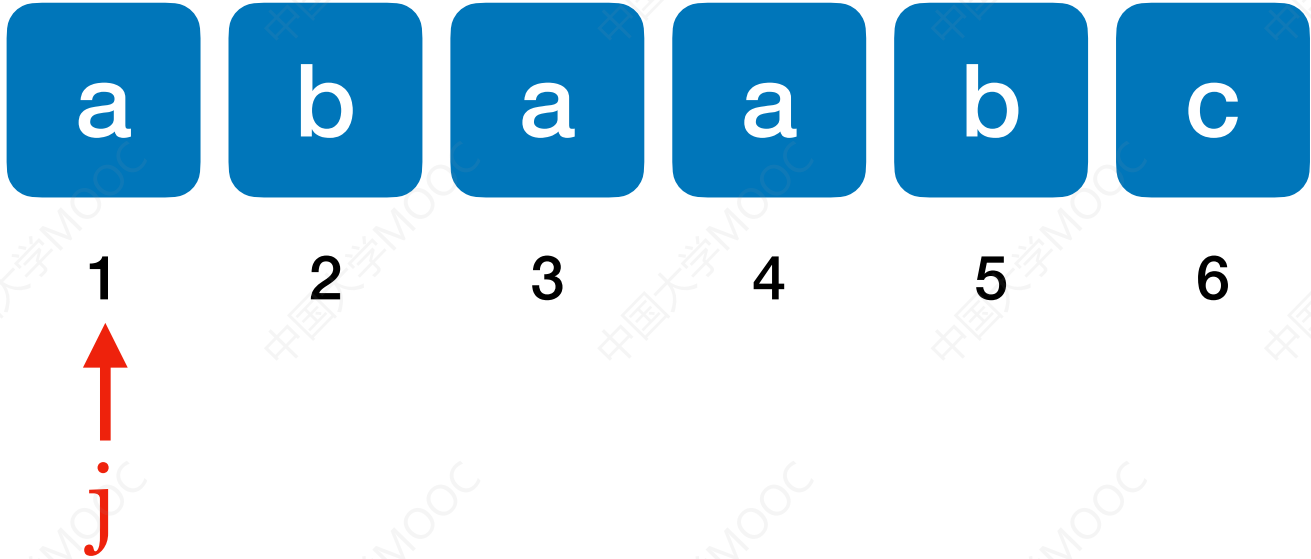
模式串T:

a	b	a	a	b	c
1	2	3	4	5	6

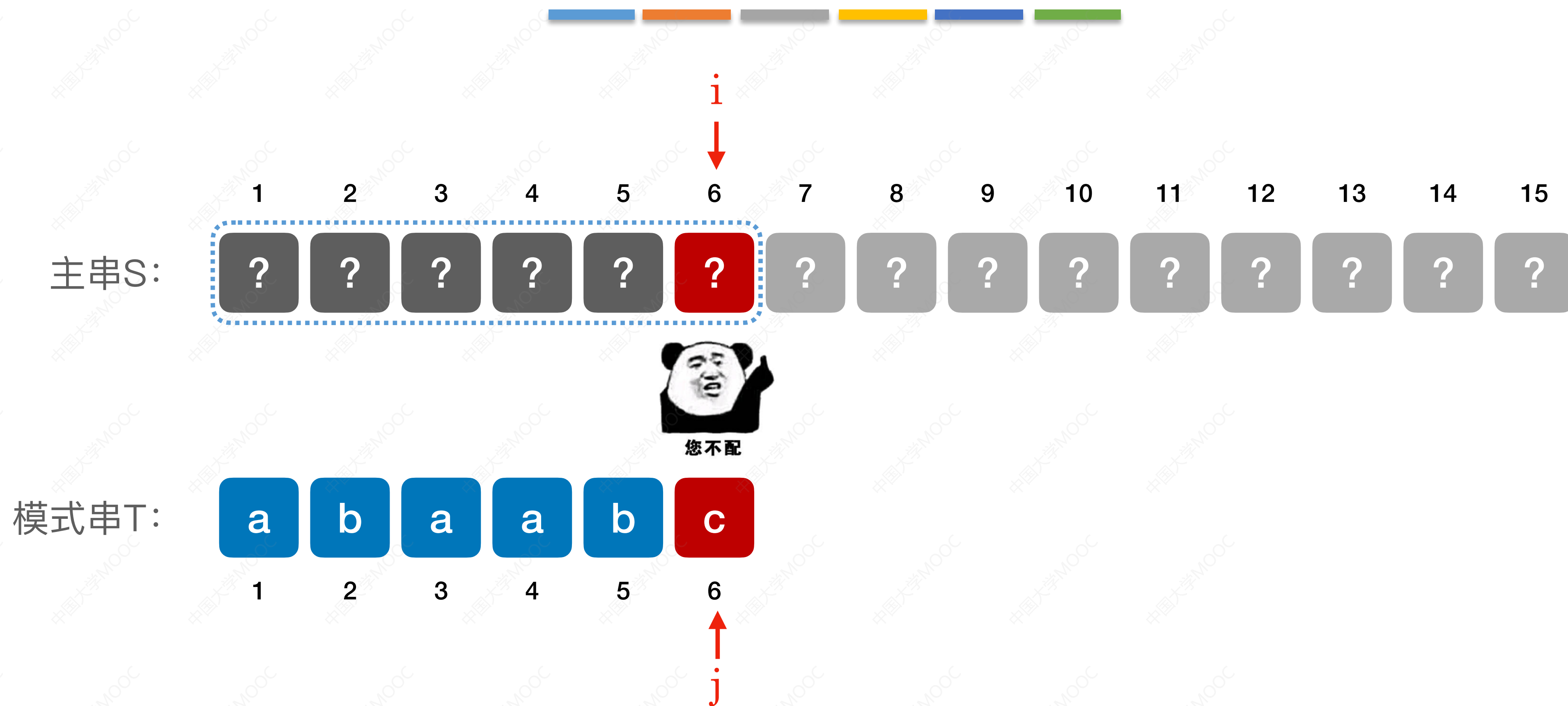
朴素模式匹配算法



模式串T:



朴素模式匹配算法



一旦发现当前这个子串中某个字符不匹配，就只能转而匹配下一个子串（从头开始）

朴素模式匹配算法



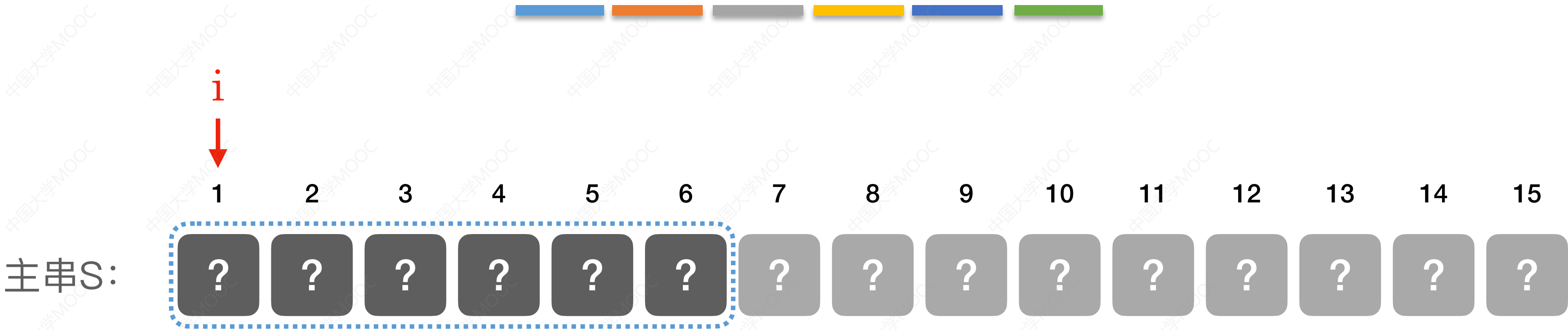
一旦发现当前这个子串中某个字符不匹配，就只能转而匹配下一个子串（从头开始）

朴素模式匹配算法

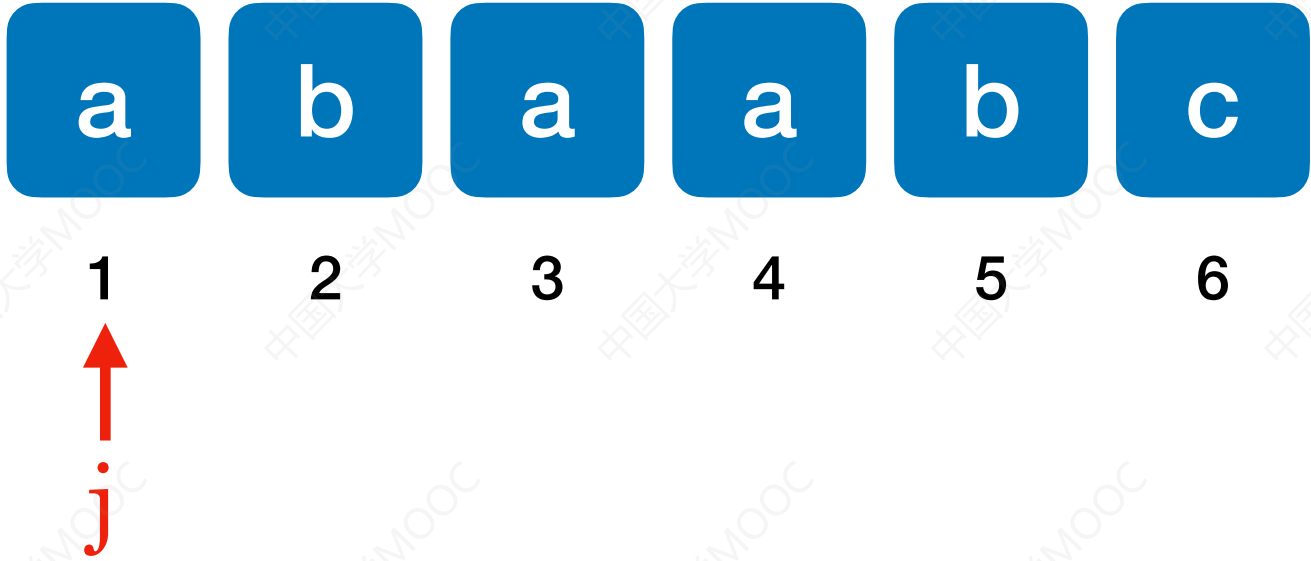


一旦发现当前这个子串中某个字符不匹配，就只能转而匹配下一个子串（从头开始）

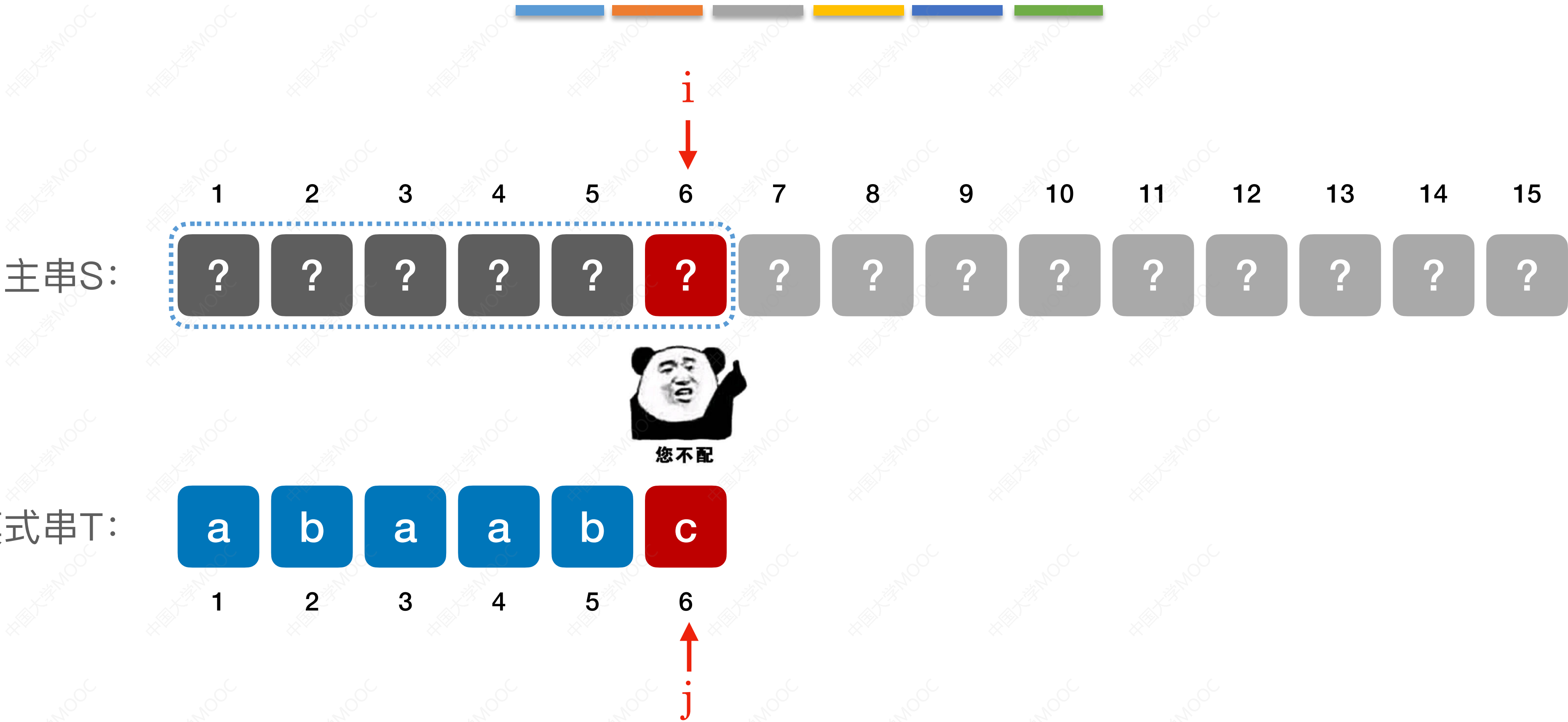
朴素模式匹配算法



模式串T:



朴素模式匹配算法



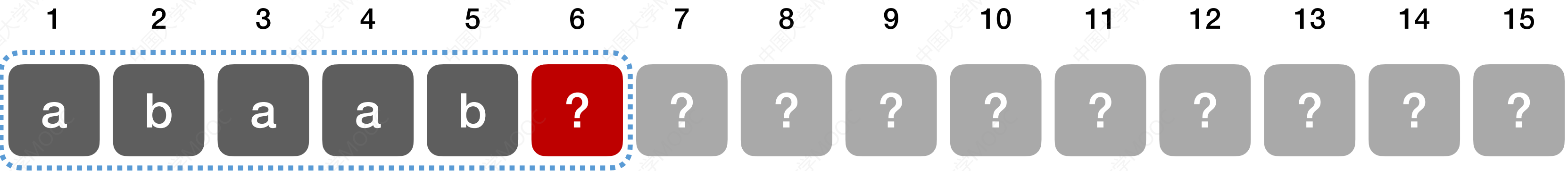
朴素模式匹配算法



i



主串S:



您不配

模式串T:



1

2

3

4

5

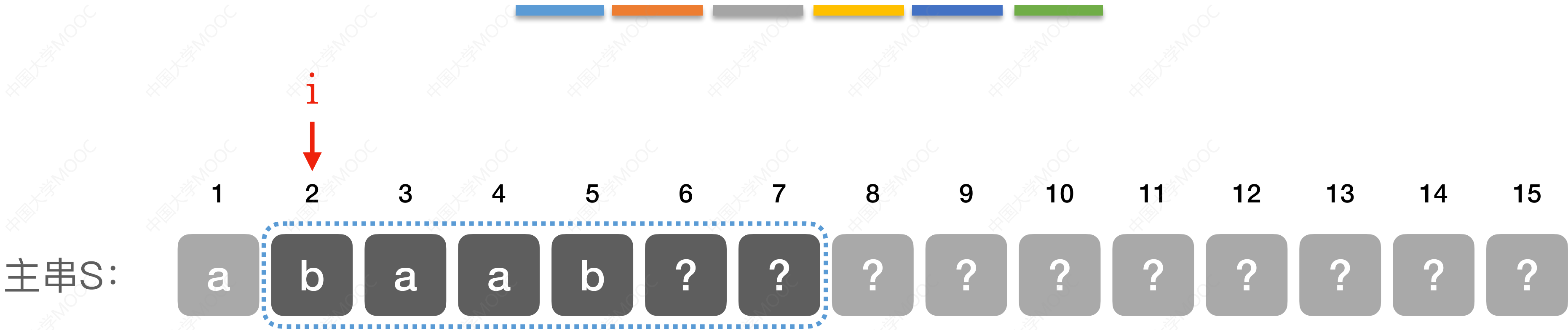
6



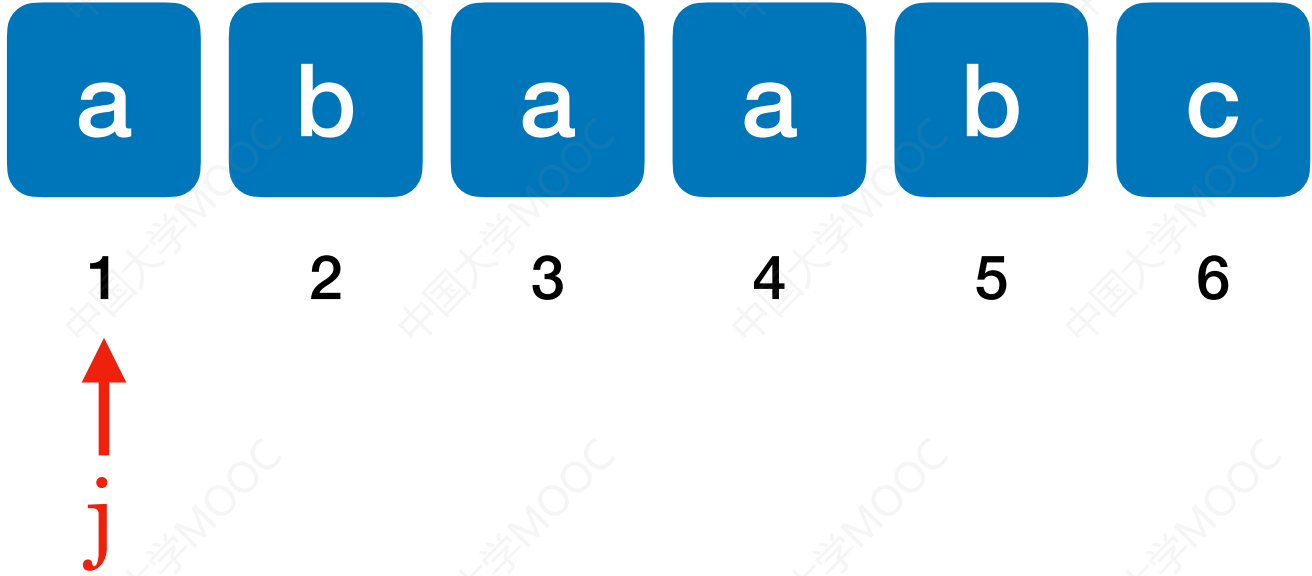
j

不匹配的字符之前，一定是和模式串一致的

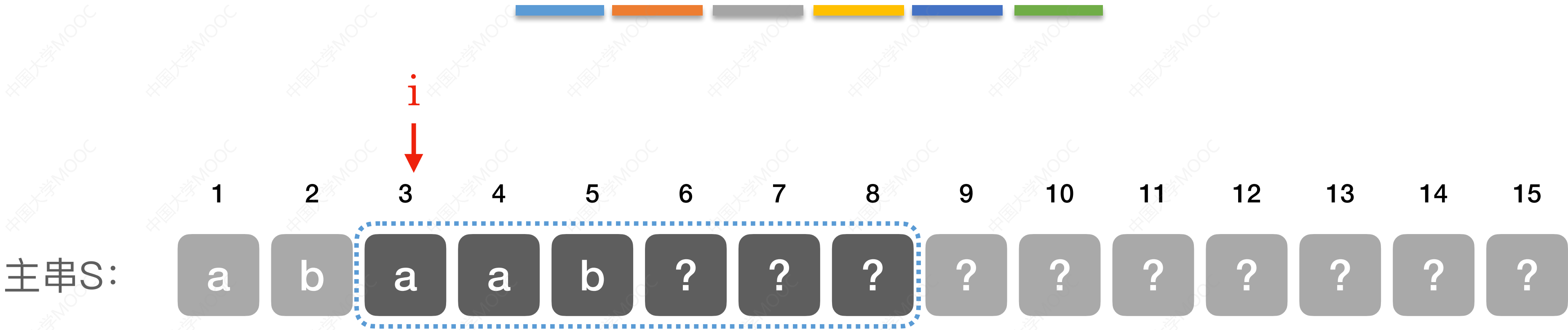
朴素模式匹配算法



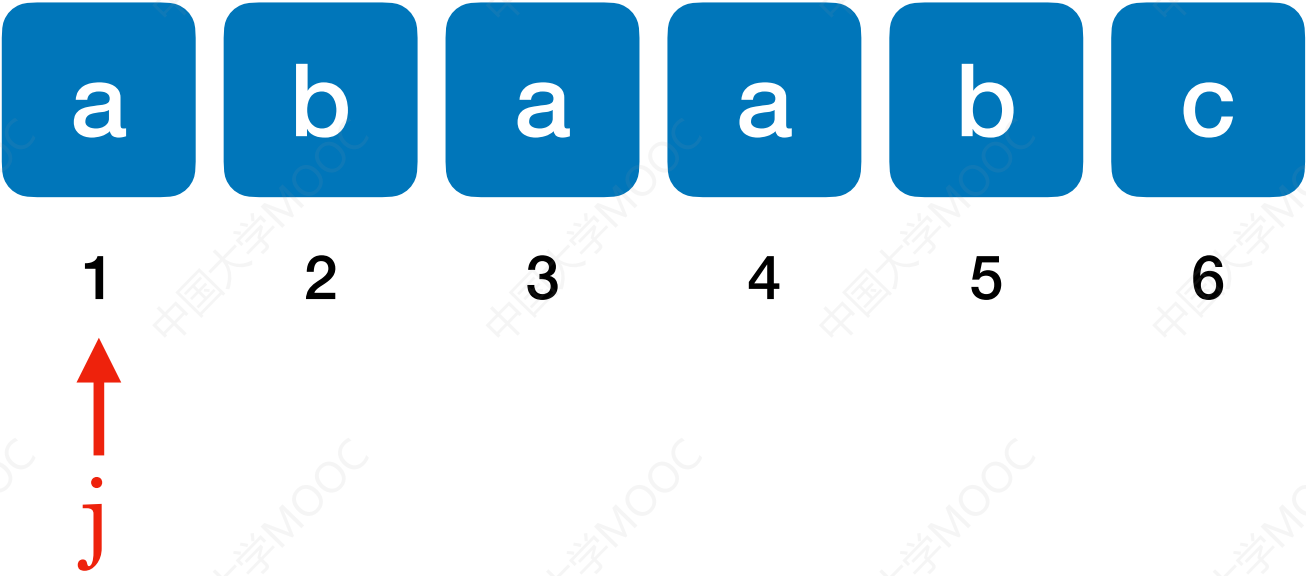
模式串T:



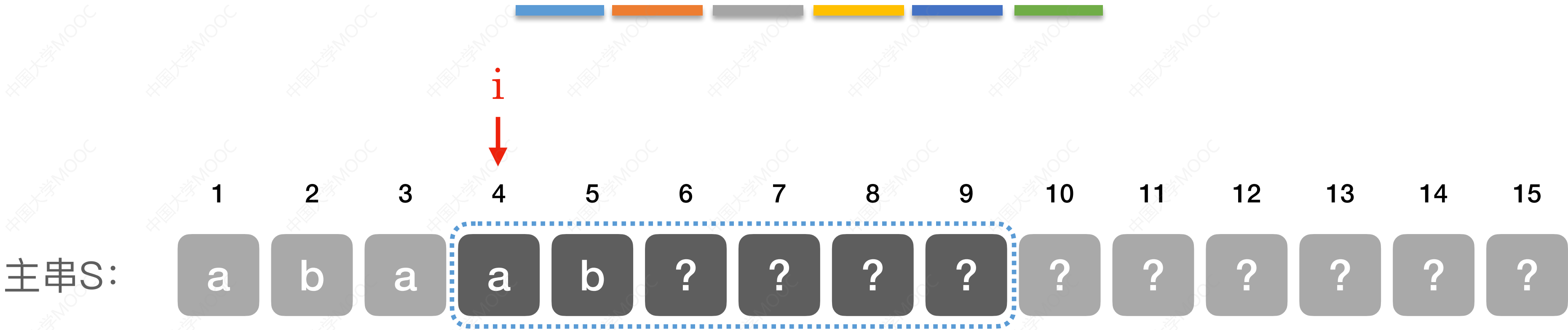
朴素模式匹配算法



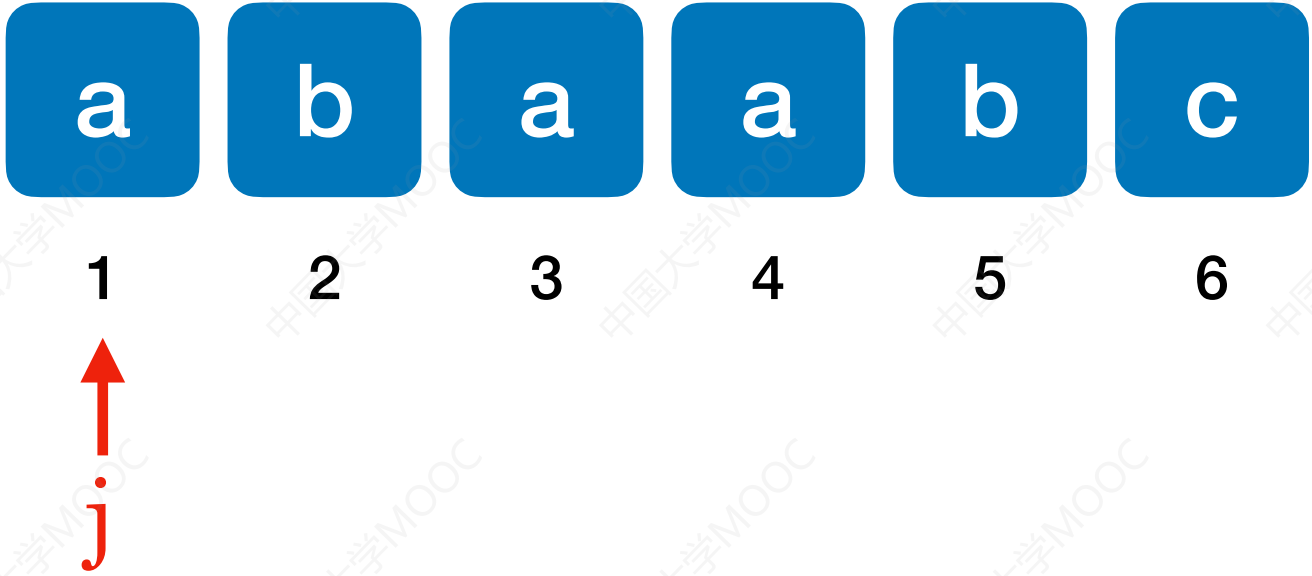
模式串T:



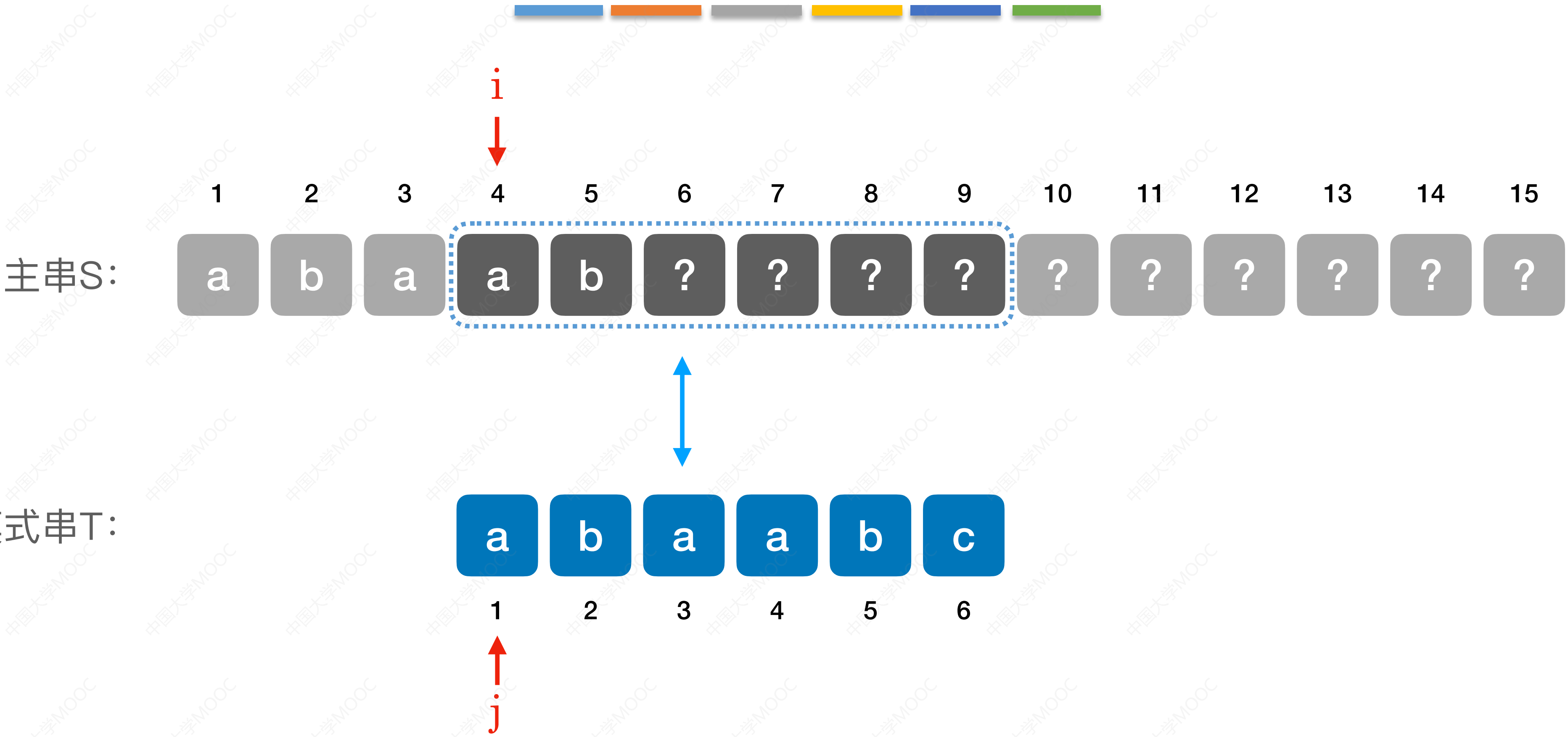
朴素模式匹配算法



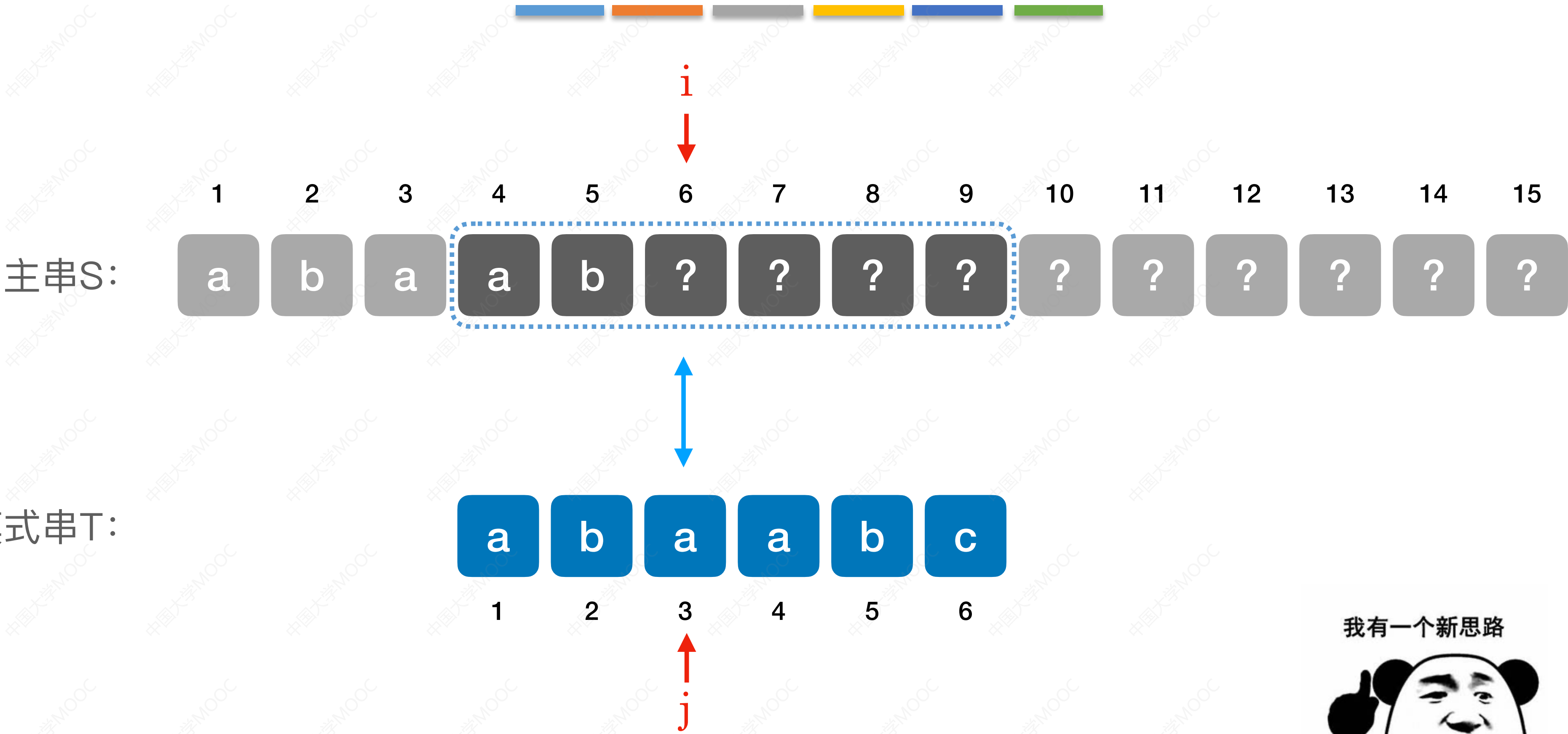
模式串T:



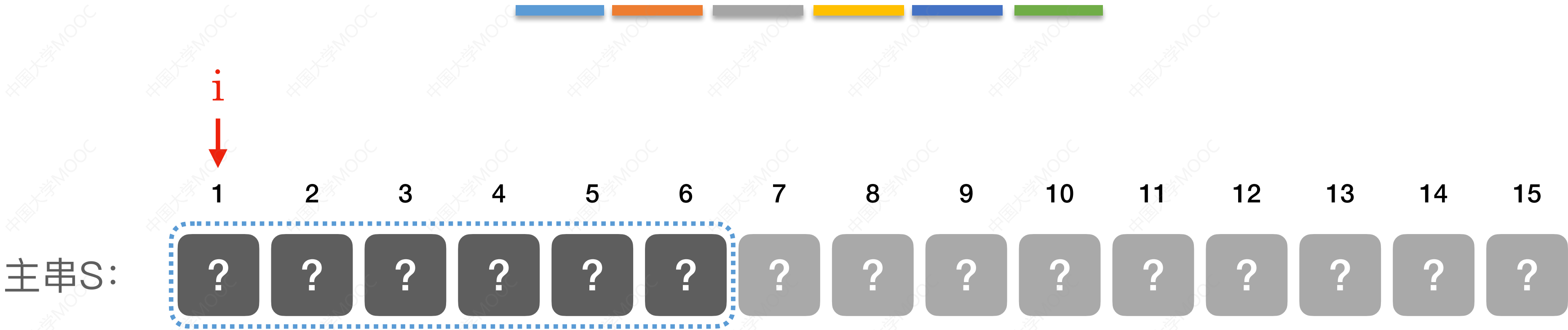
朴素模式匹配算法



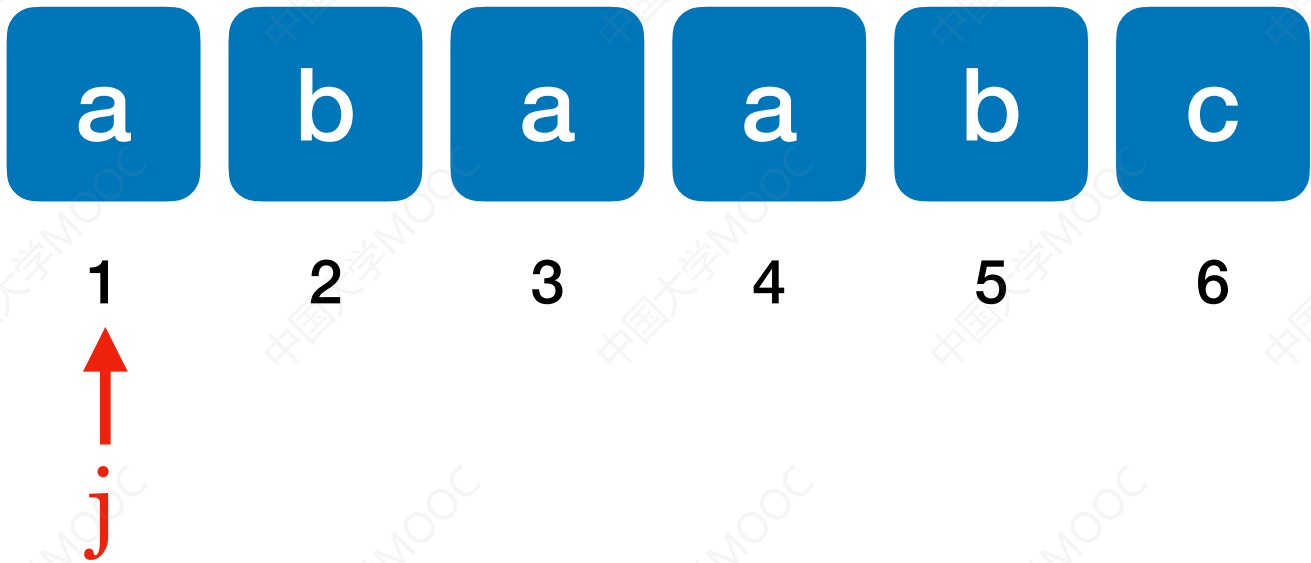
朴素模式匹配算法



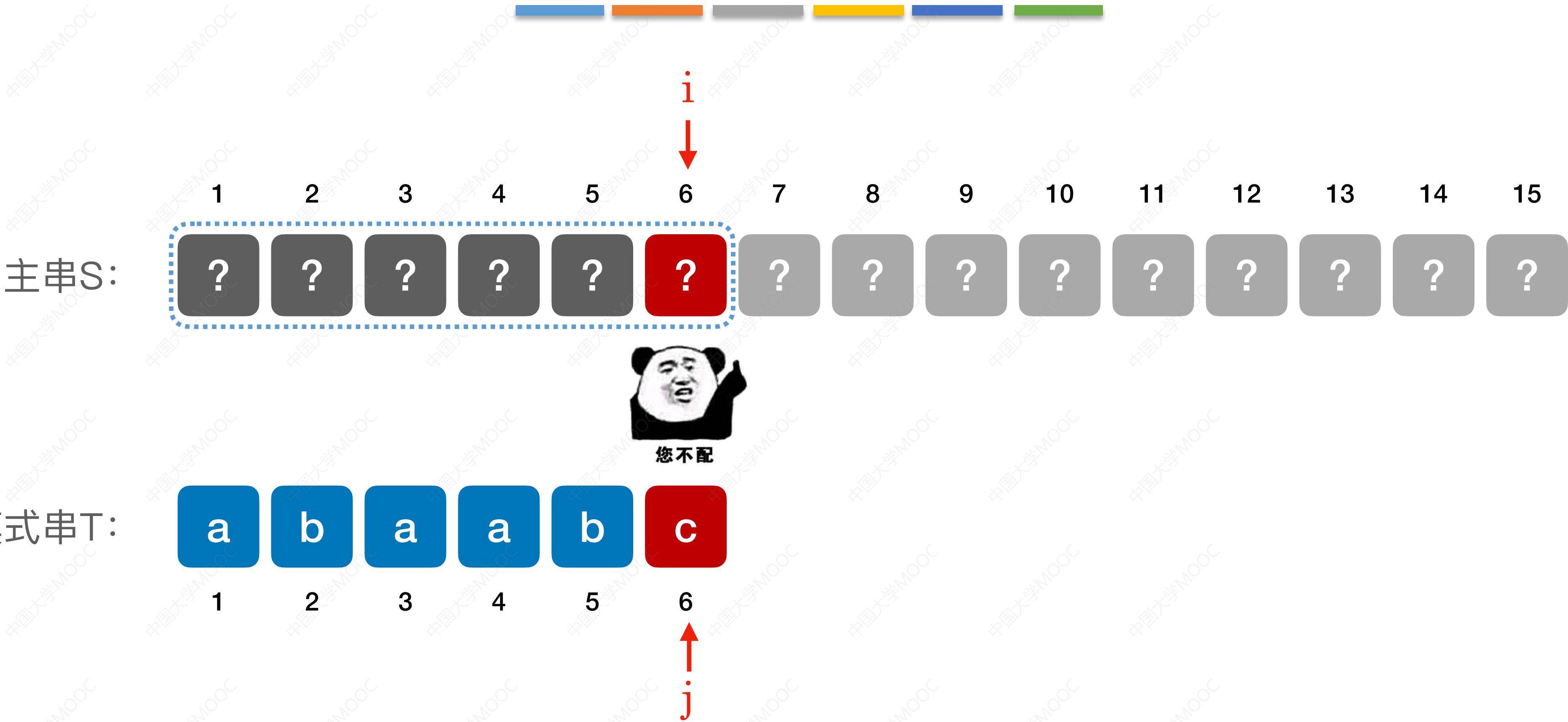
朴素模式匹配算法优化思路



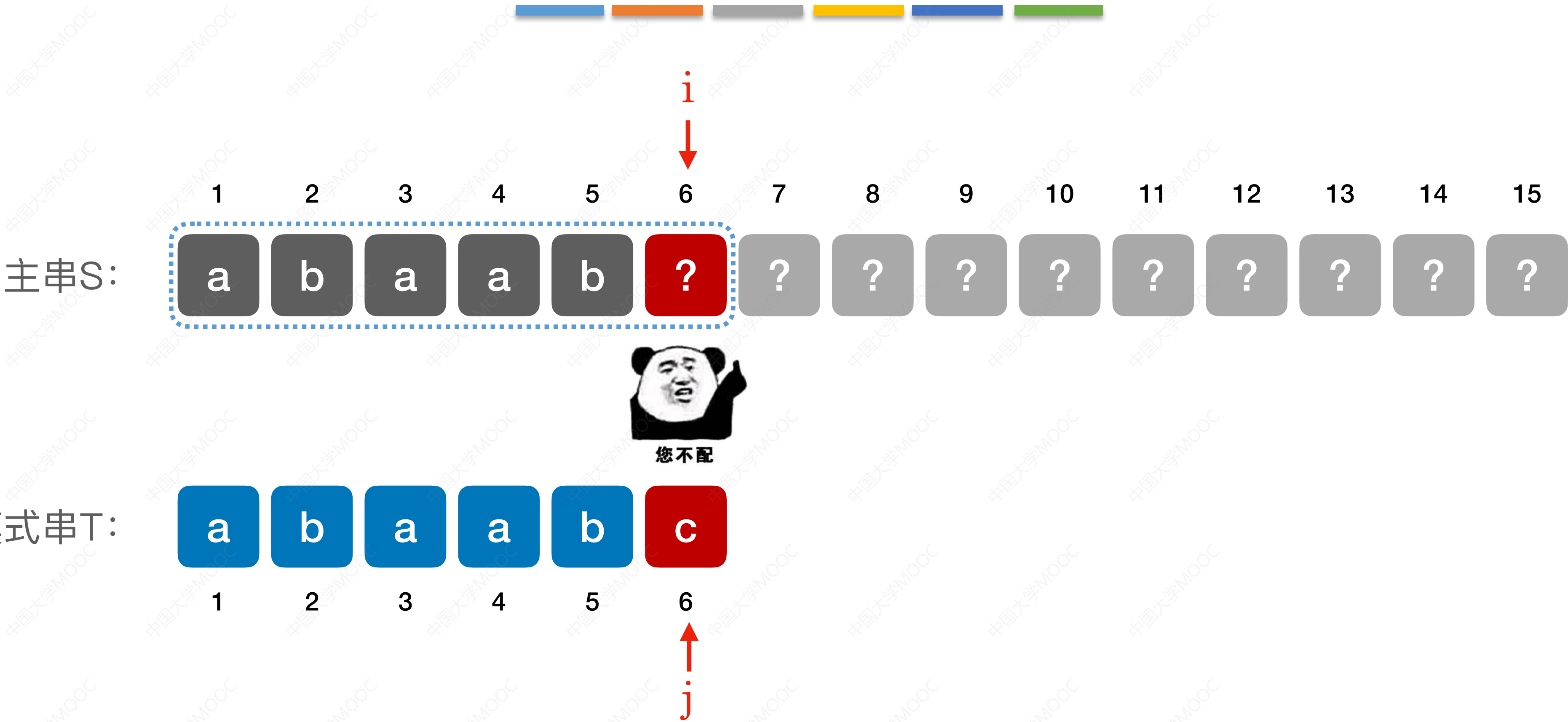
模式串T:



朴素模式匹配算法优化思路

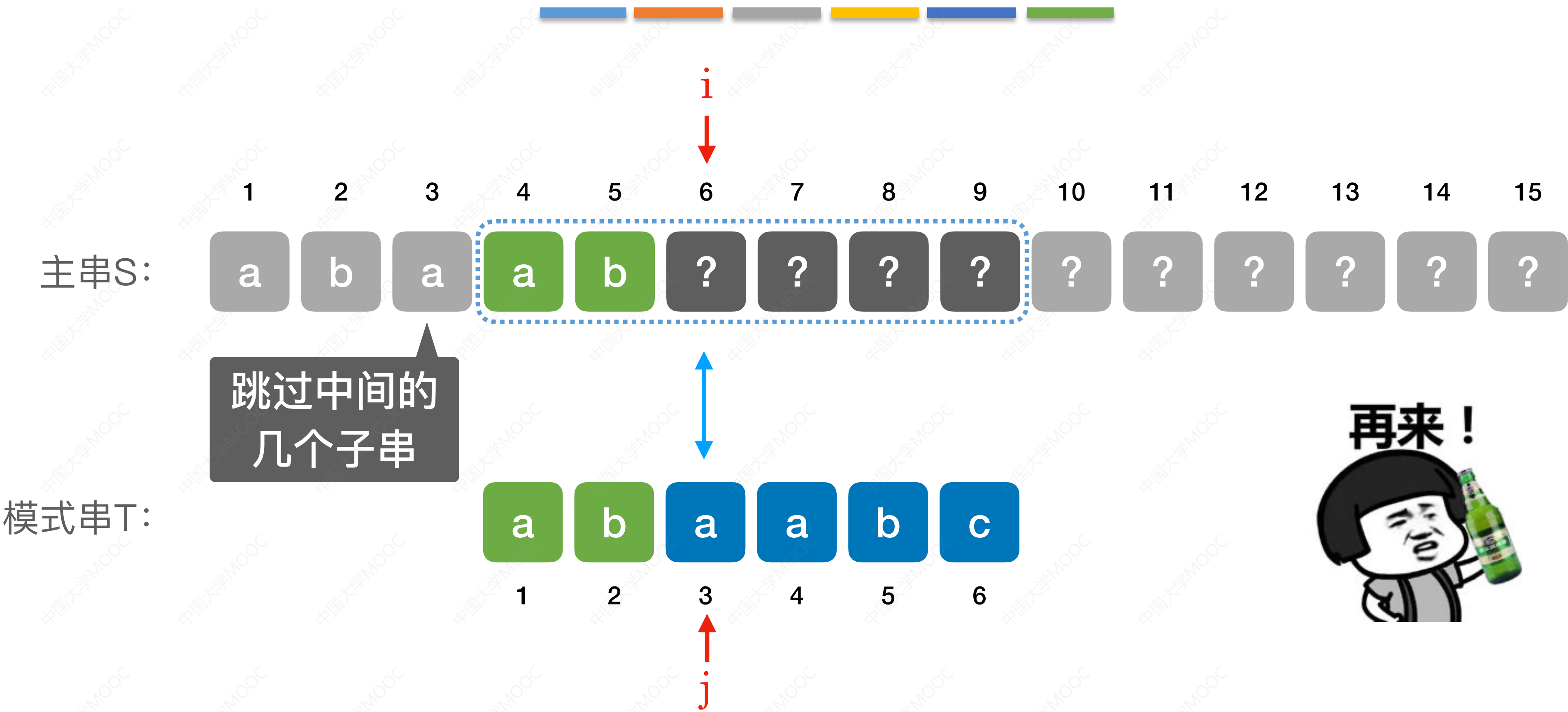


朴素模式匹配算法优化思路



不匹配的字符之前，一定是和模式串一致的

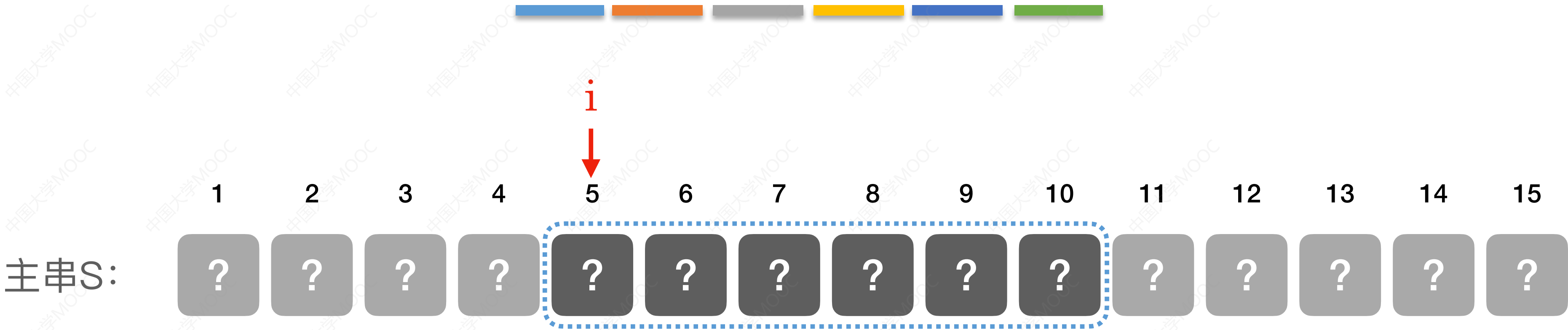
朴素模式匹配算法优化思路



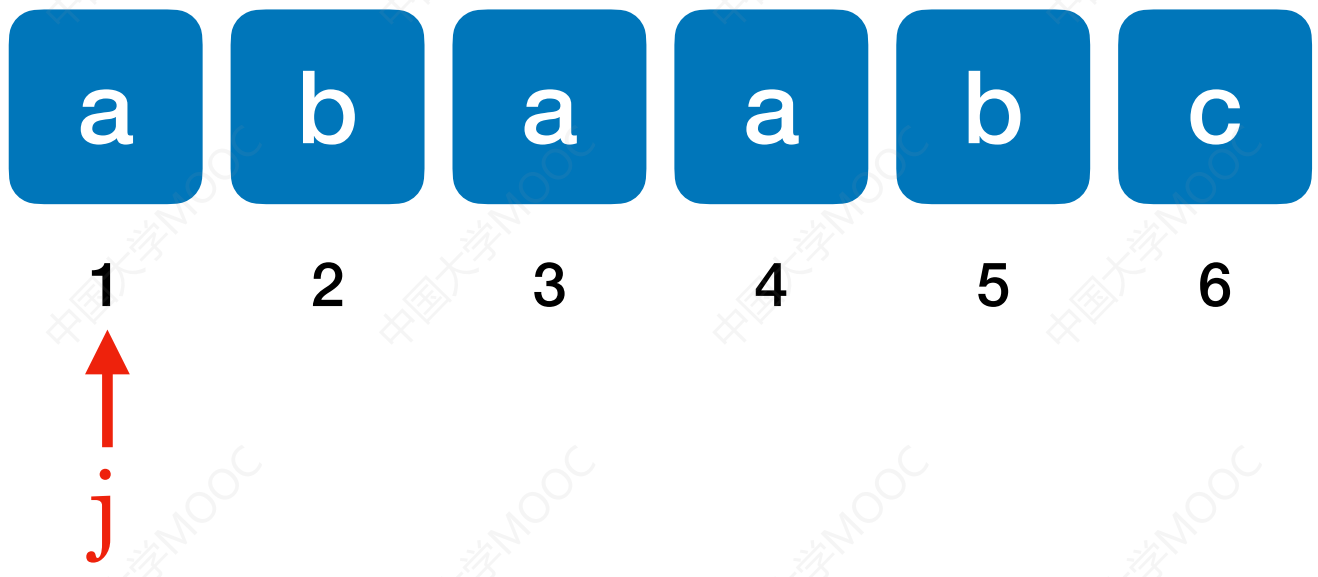
可以直接从这里继续匹配

对于模式串 $T = \text{'abaabc'}$ ，当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

朴素模式匹配算法优化思路

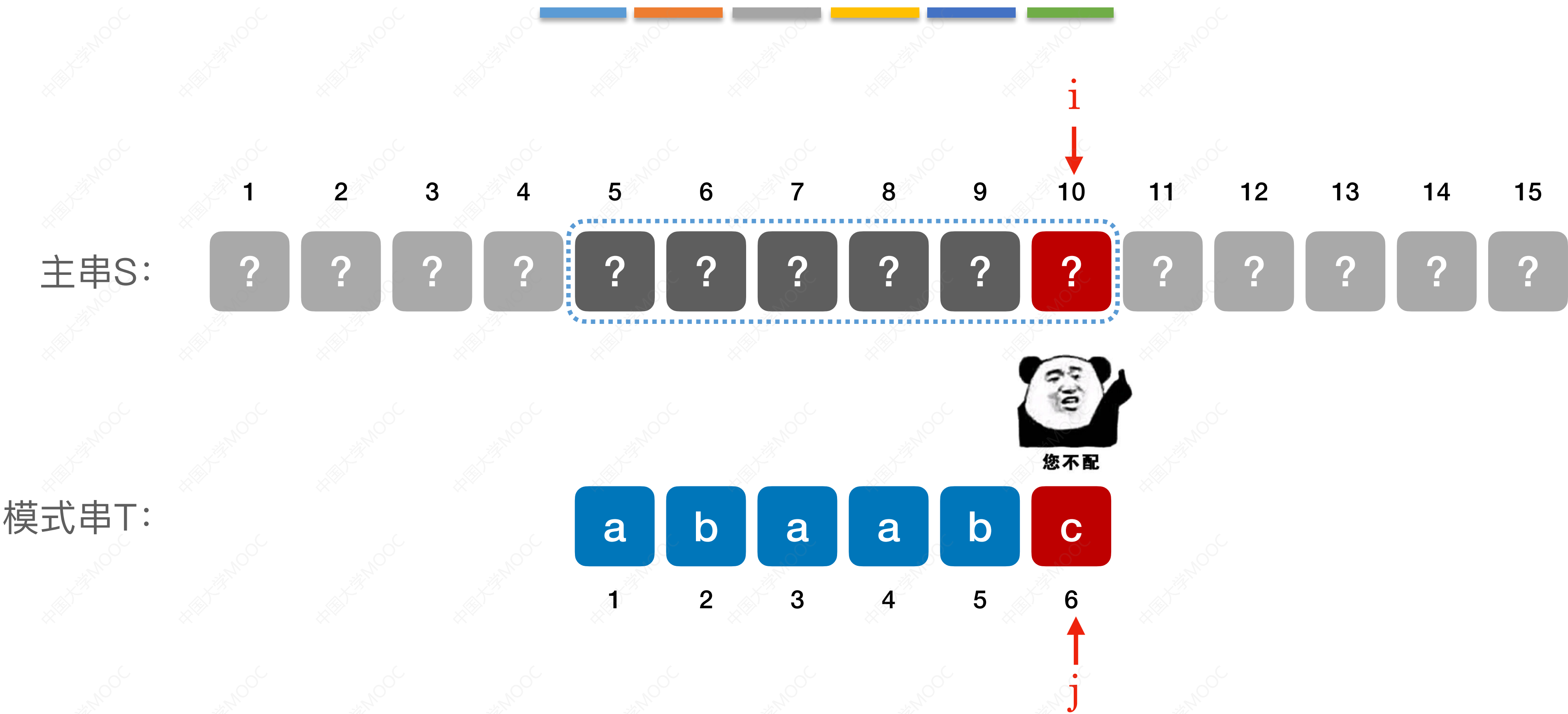


模式串T:



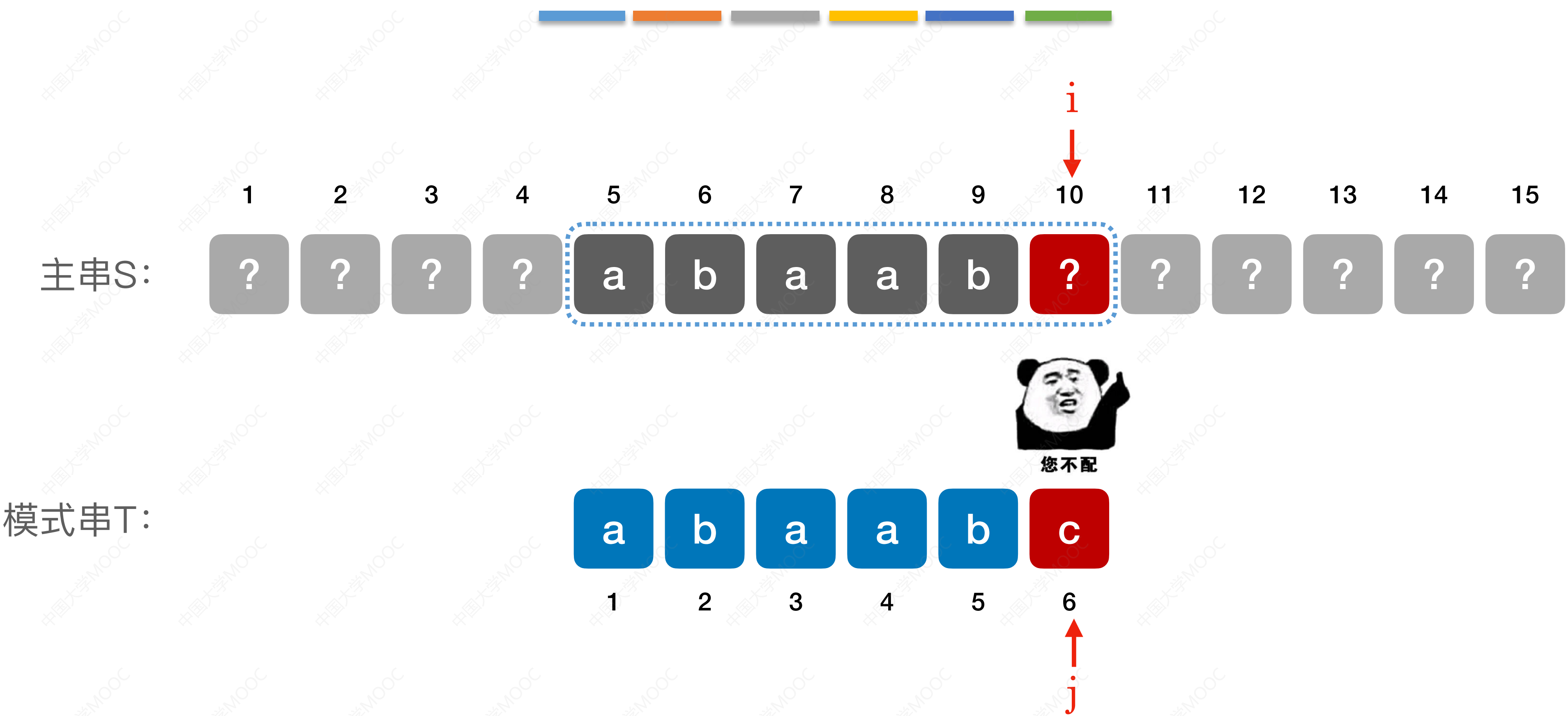
对于模式串 T = 'abaabc'，当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=3

朴素模式匹配算法优化思路



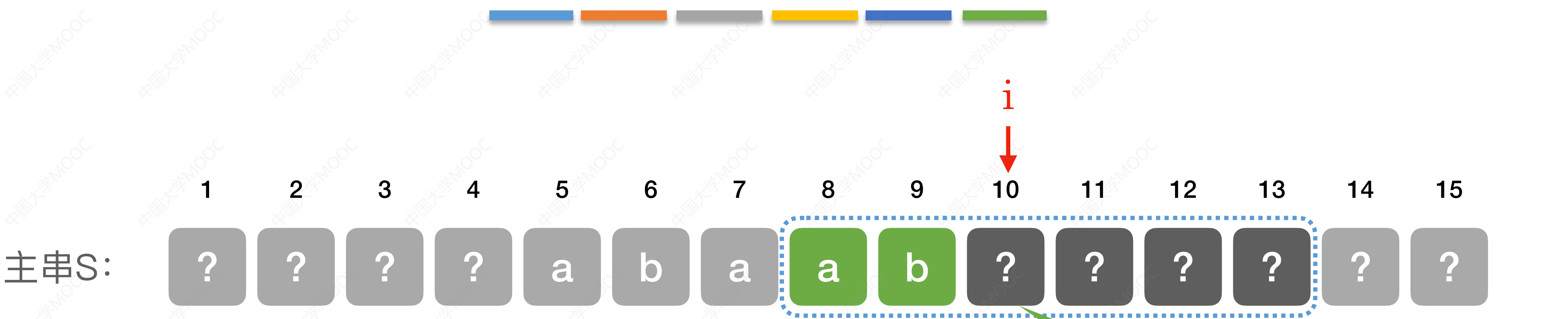
对于模式串 $T = \text{'abaabc'}$ ，当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

朴素模式匹配算法优化思路

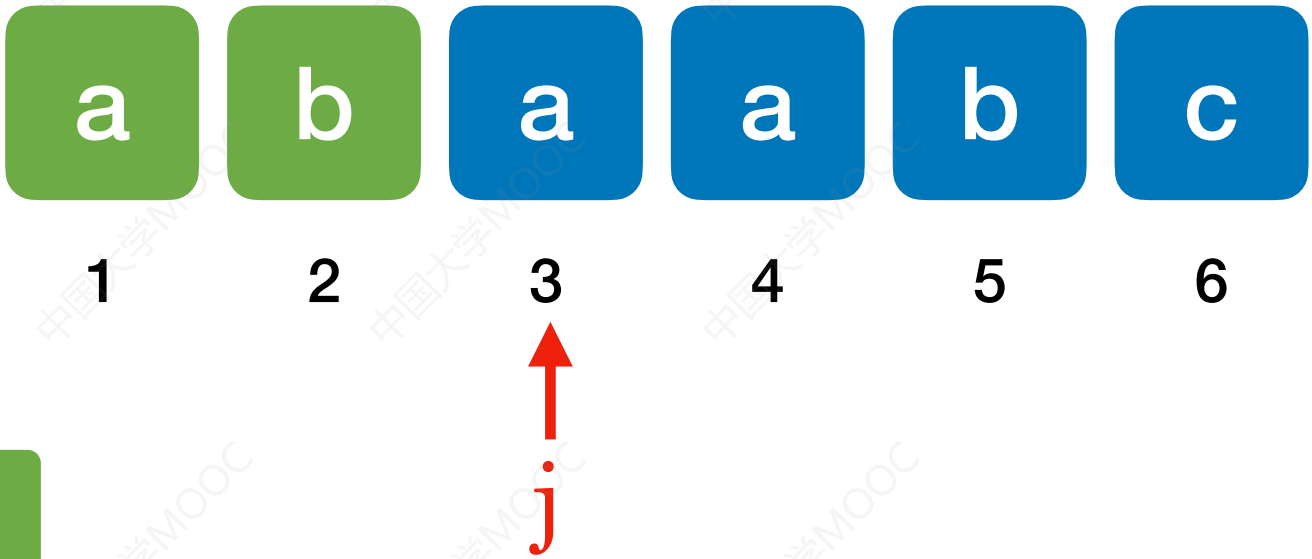


对于模式串 $T = \text{'abaabc'}$ ，当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

朴素模式匹配算法优化思路



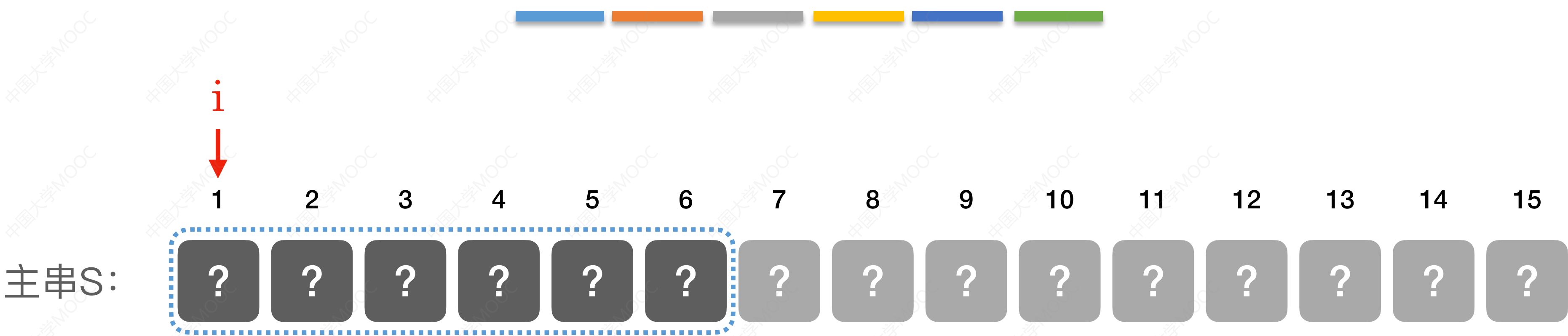
模式串T:



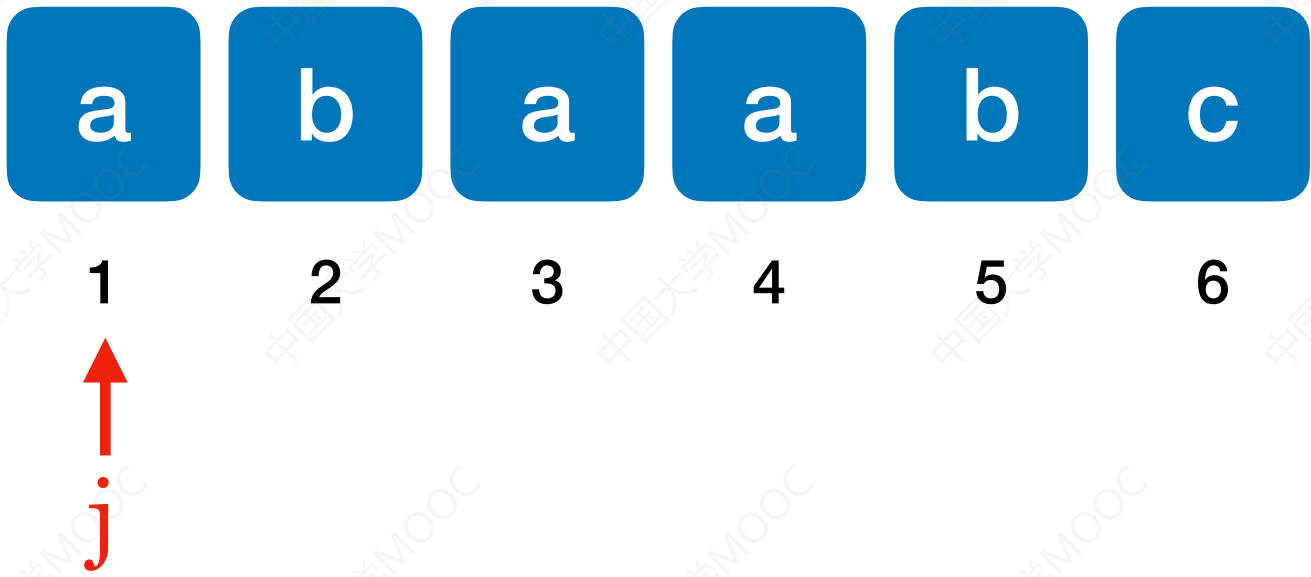
该结论对模式串'abaabc'具有通用性，和主串没有半毛钱关系

对于模式串 T = 'abaabc'，当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=3

如果其他位置不匹配呢？

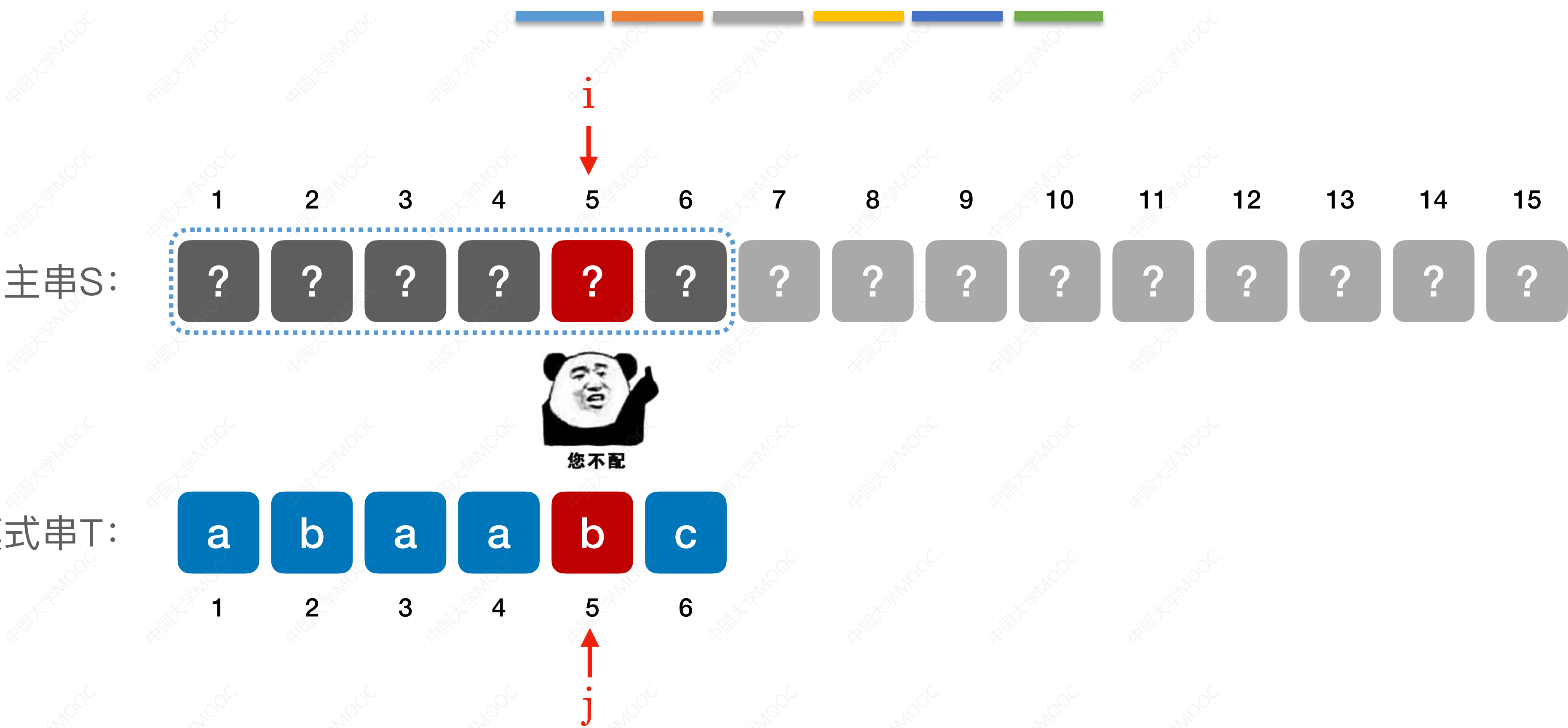


模式串T:



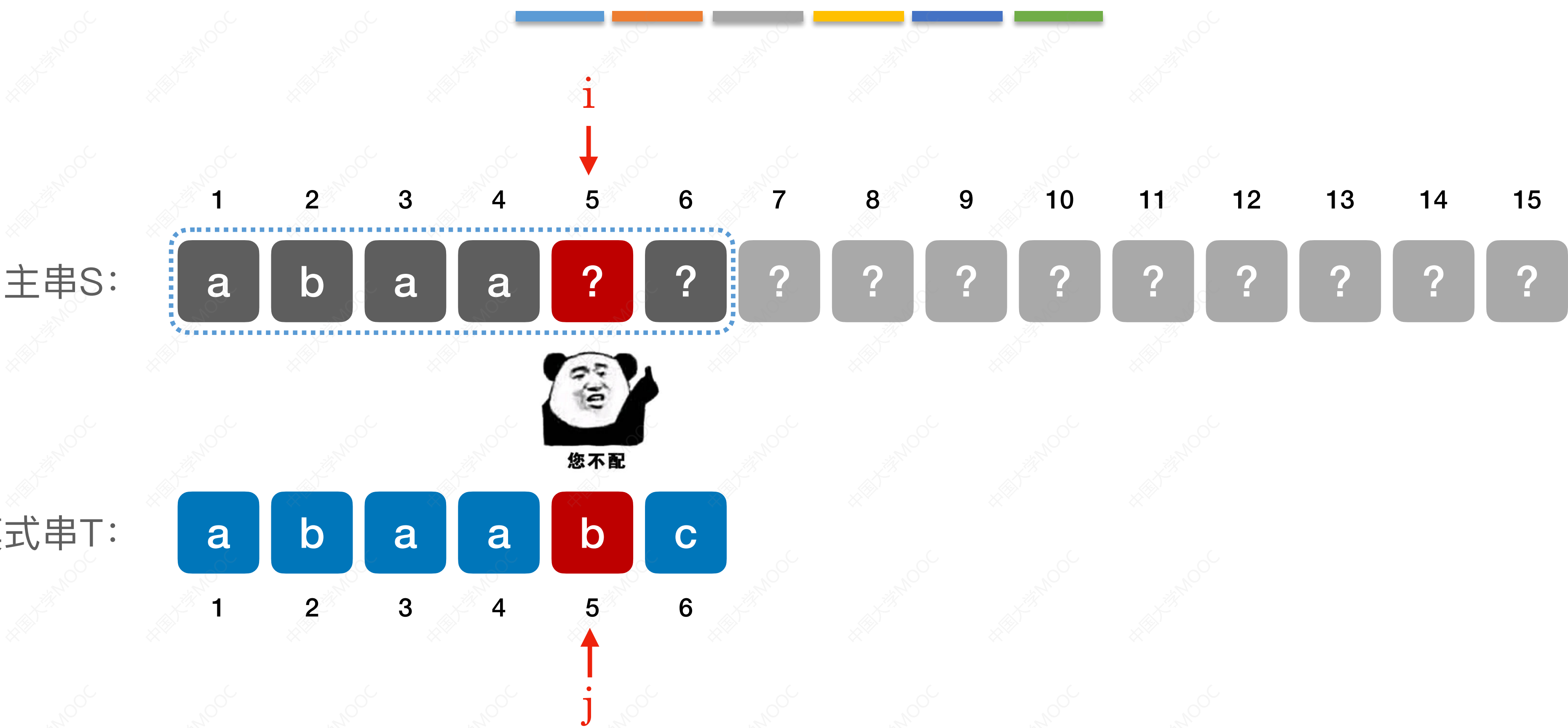
对于模式串 T = 'abaabc'，当第5个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？



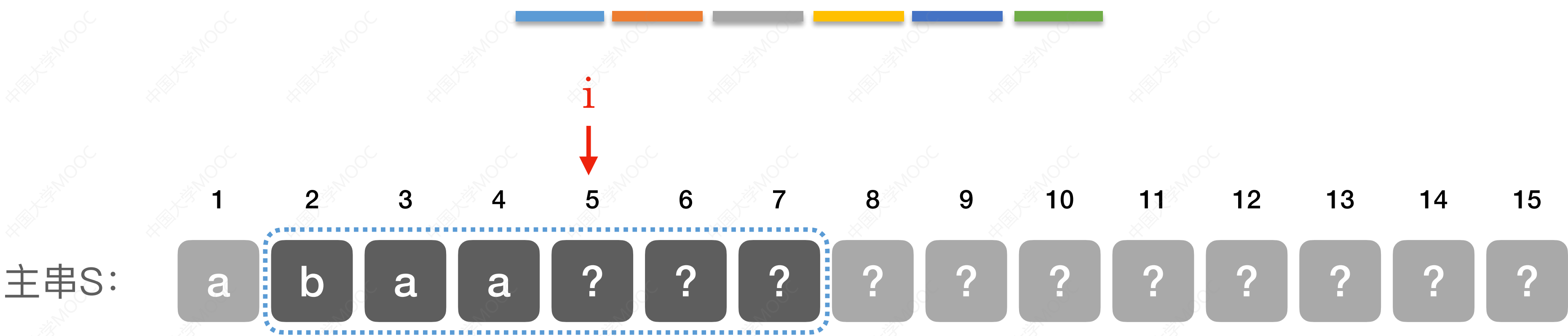
对于模式串 $T = \text{'abaabc'}$ ，当第5个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？

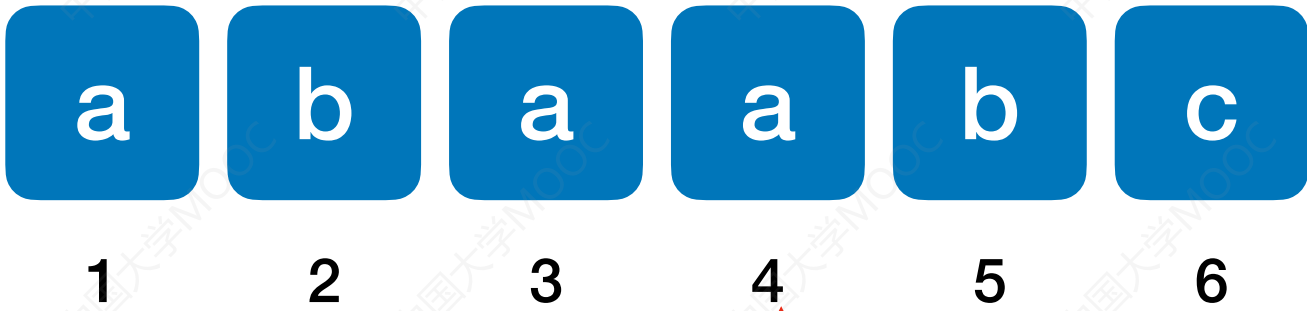


对于模式串 $T = \text{'abaabc'}$ ，当第5个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？

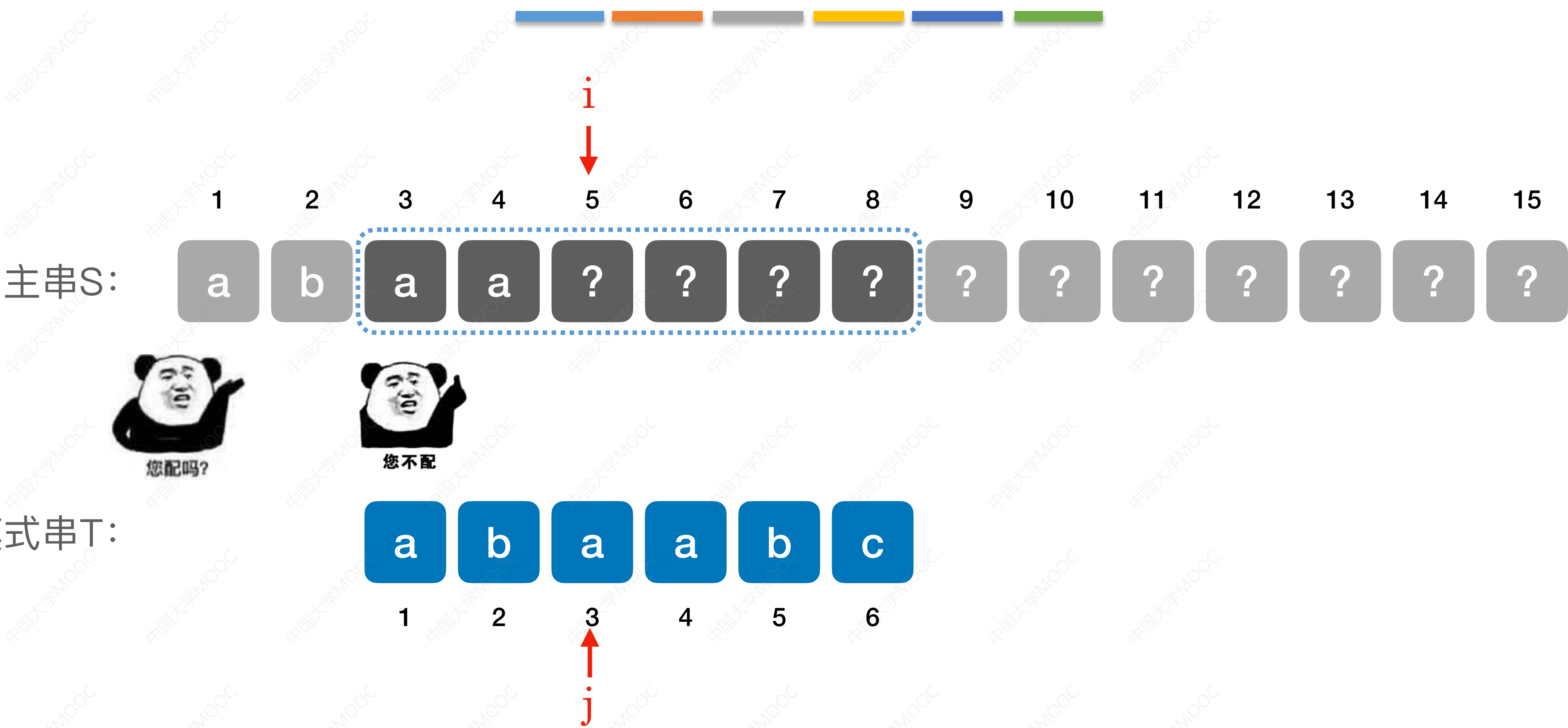


主串S:



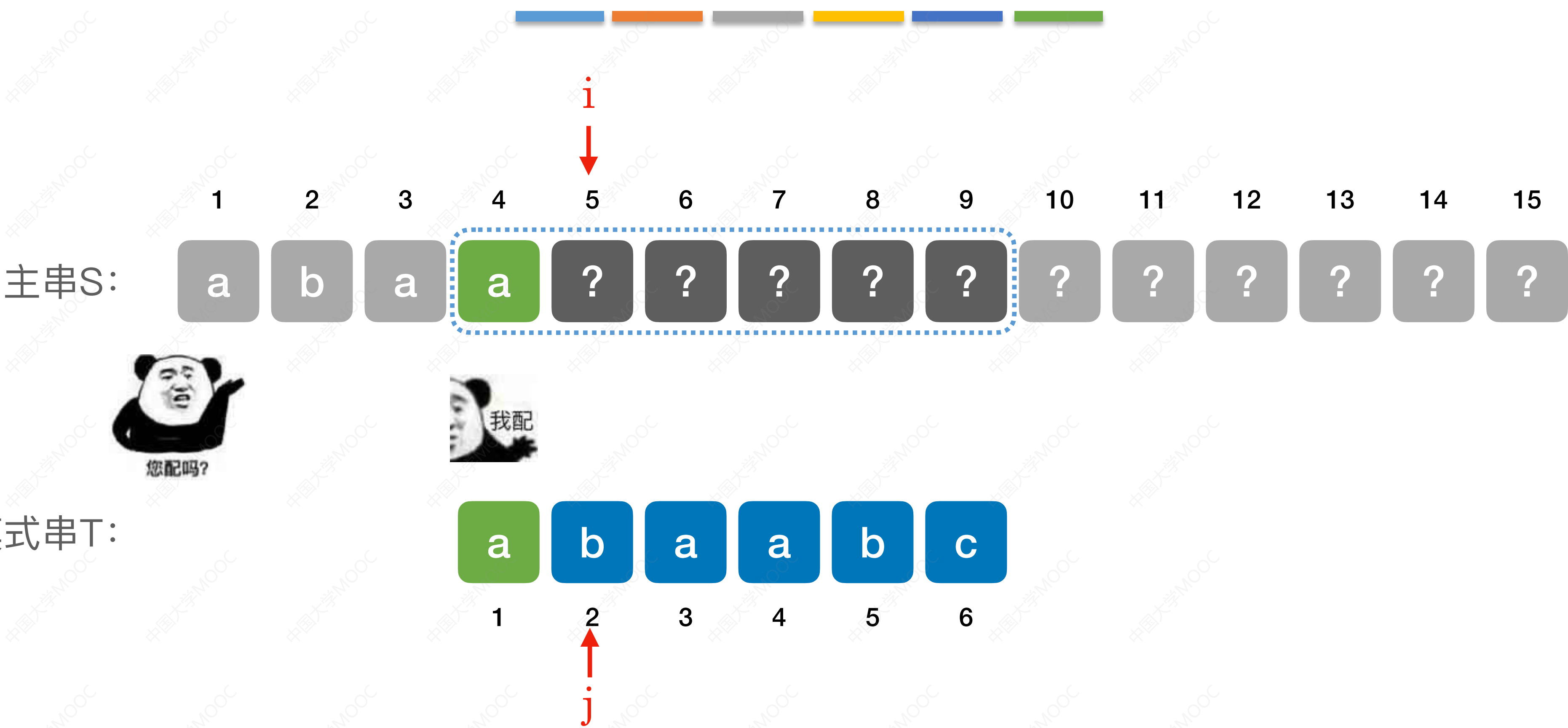
对于模式串 $T = \text{'abaabc'}$ ，当第5个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？



对于模式串 T = 'abaabc'，当第5个元素匹配失败时？怎么搞？

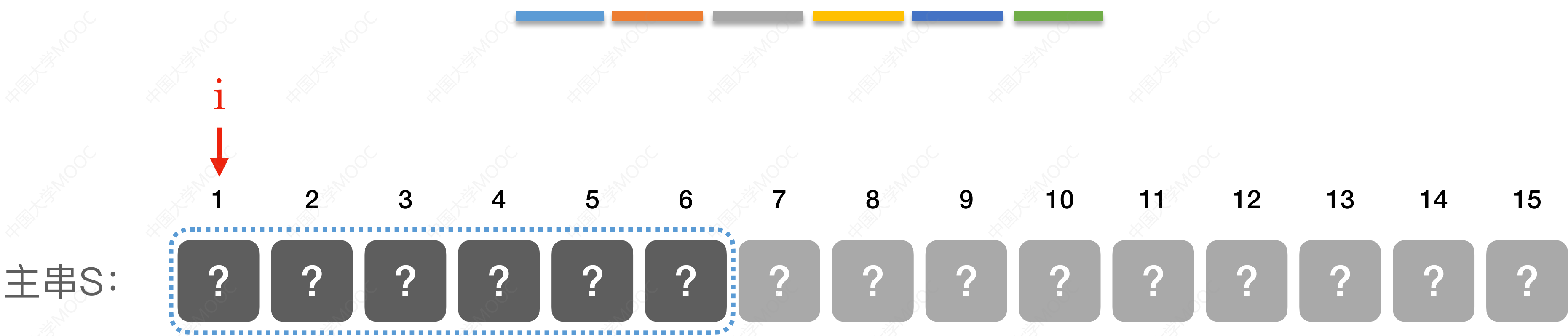
如果其他位置不匹配呢？



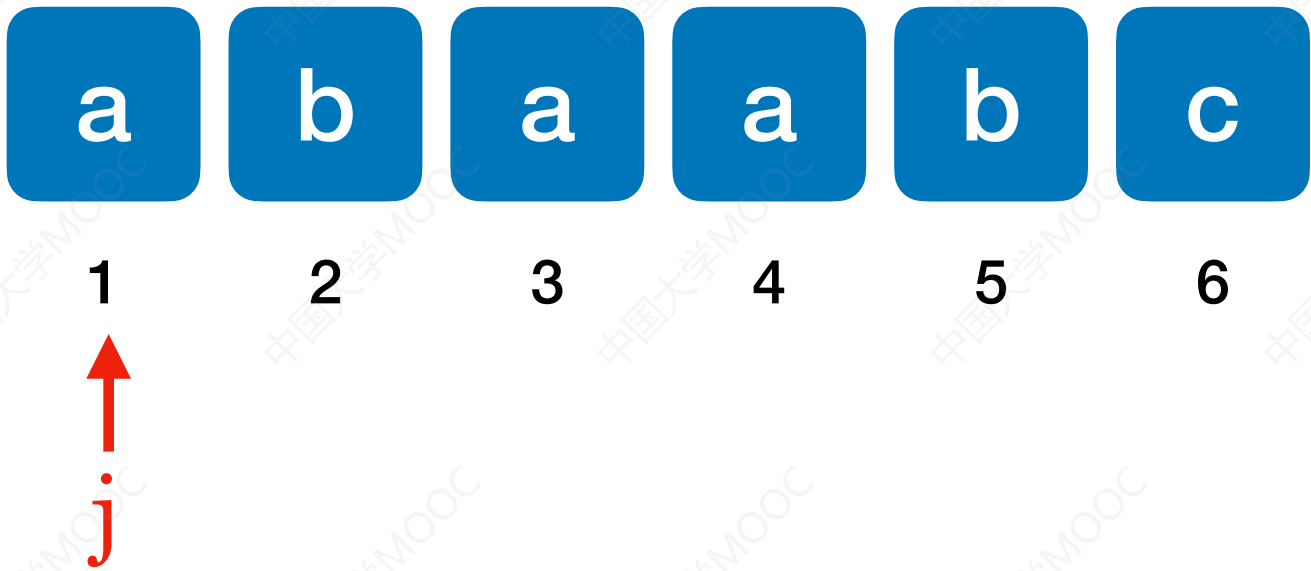
对于模式串 $T = \text{'abaabc'}$ ，当第5个元素匹配失败时？怎么搞？

可令主串指针 i 不变，模式串指针 $j = 2$

如果其他位置不匹配呢？

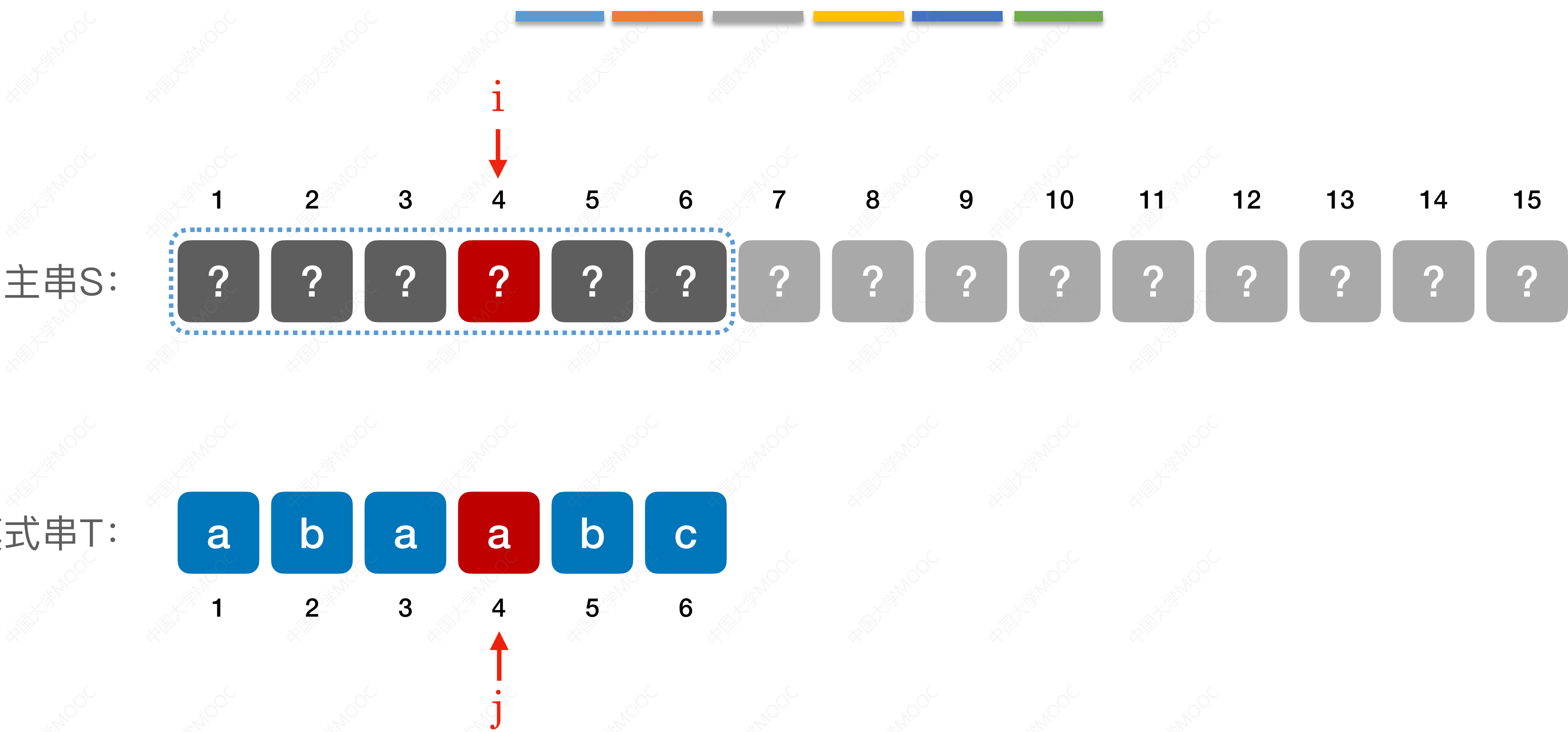


模式串T:



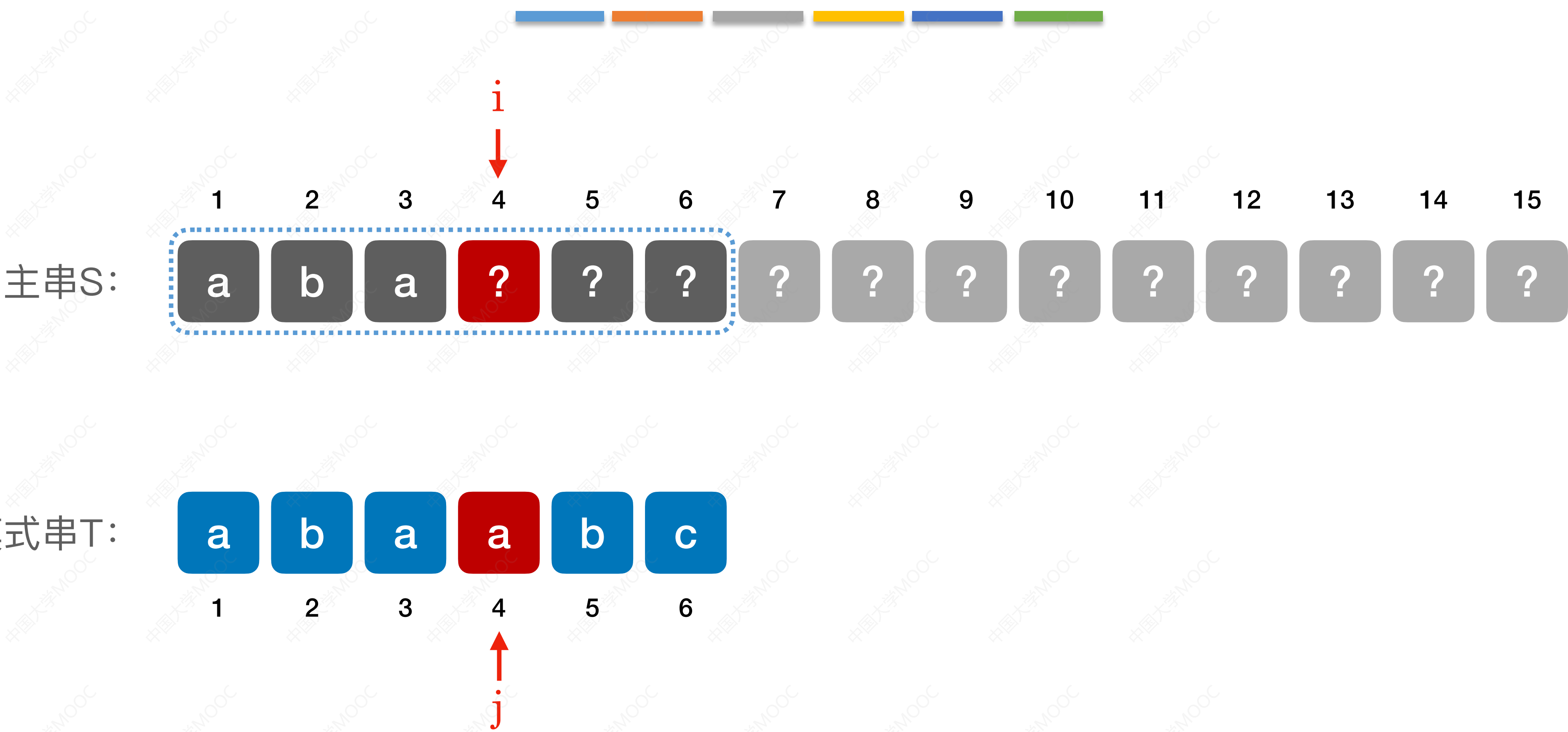
对于模式串 T = 'abaabc'，当第4个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？



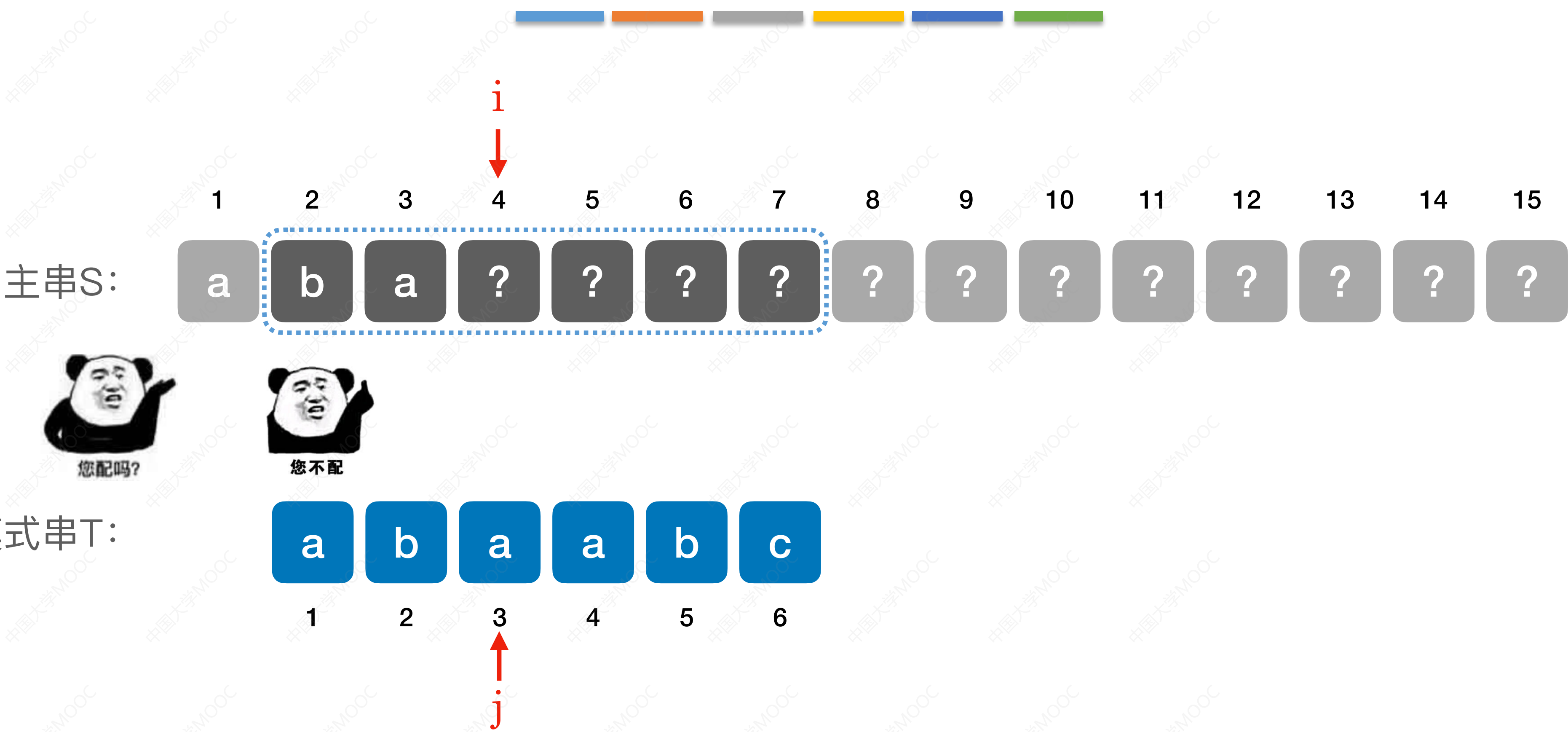
对于模式串 T = 'abaabc'，当第4个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？



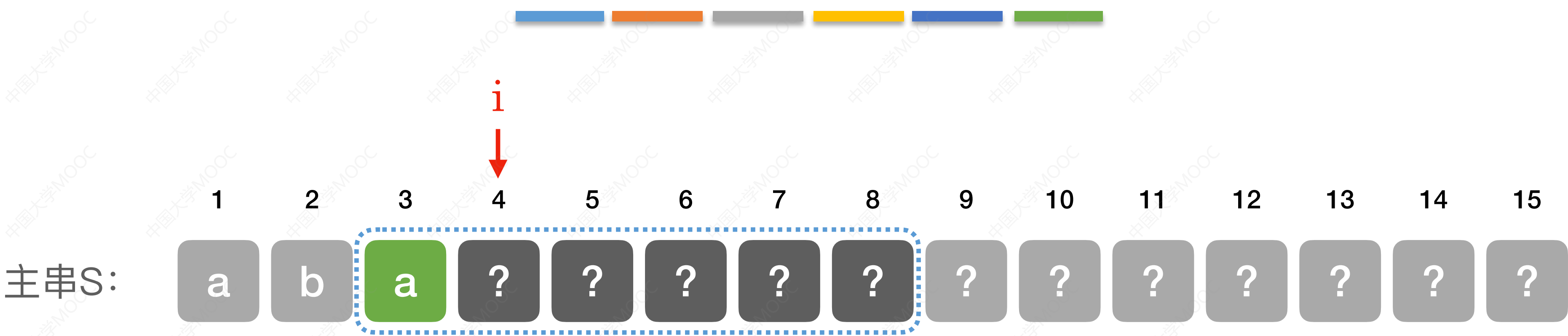
对于模式串 T = 'abaabc'，当第4个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？



对于模式串 T = 'abaabc'，当第4个元素匹配失败时？怎么搞？

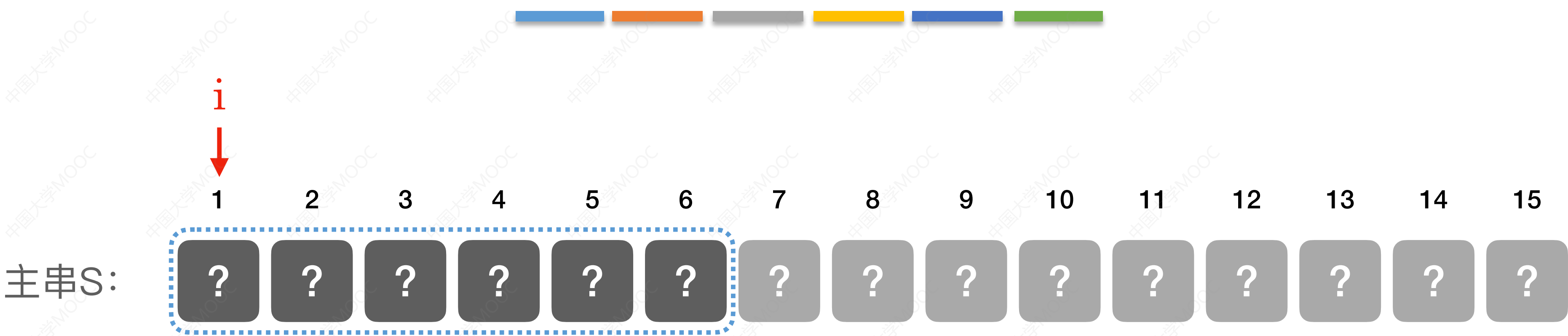
如果其他位置不匹配呢？



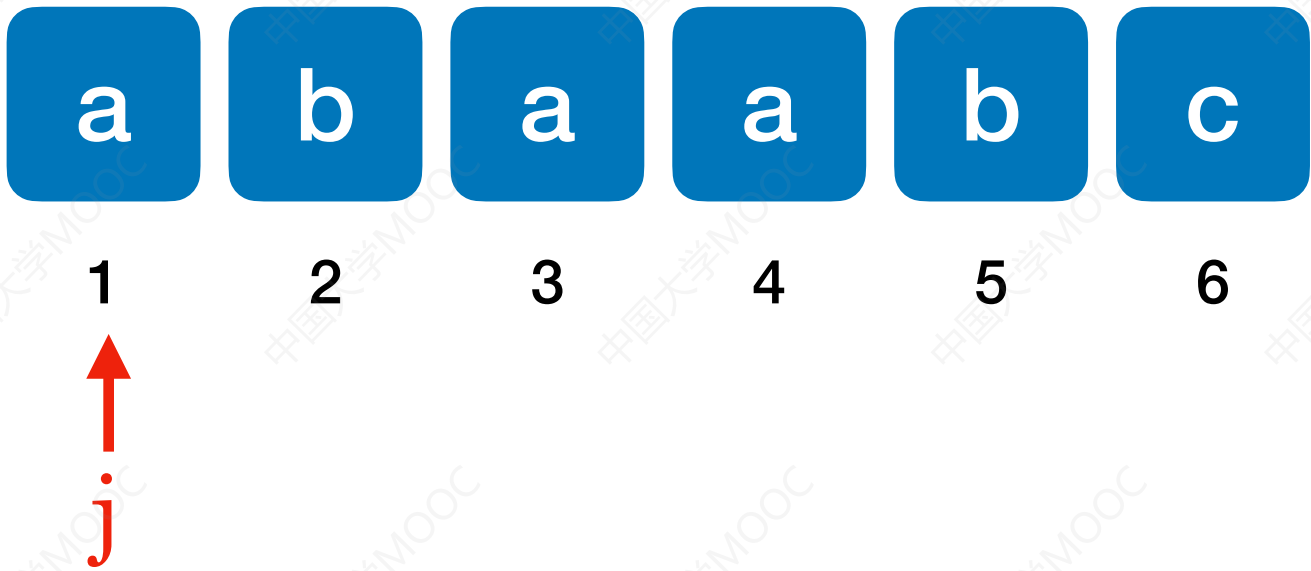
对于模式串 $T = \text{'abaabc'}$ ，当第4个元素匹配失败时？怎么搞？

可令主串指针 i 不变，模式串指针 $j = 2$

如果其他位置不匹配呢？

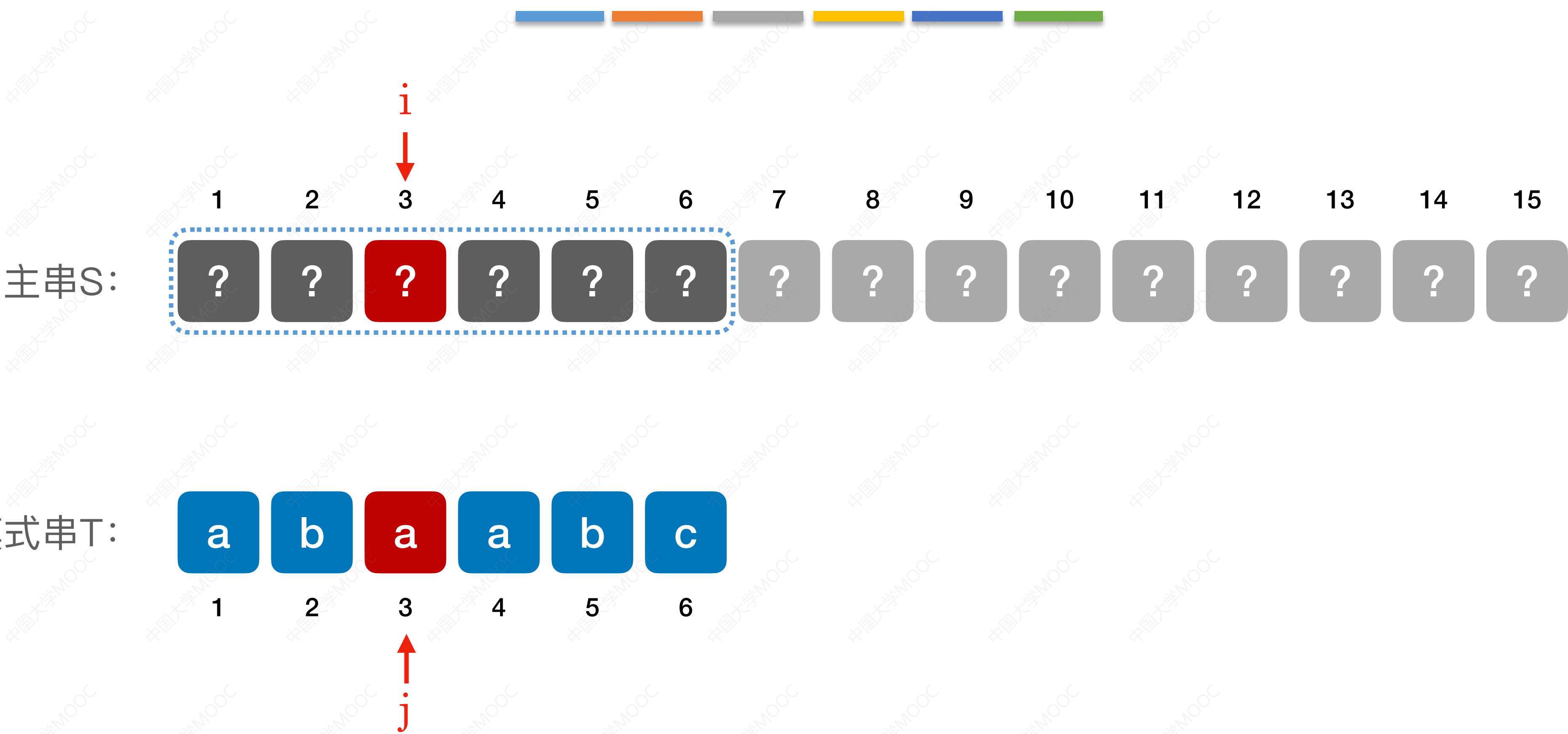


模式串T:

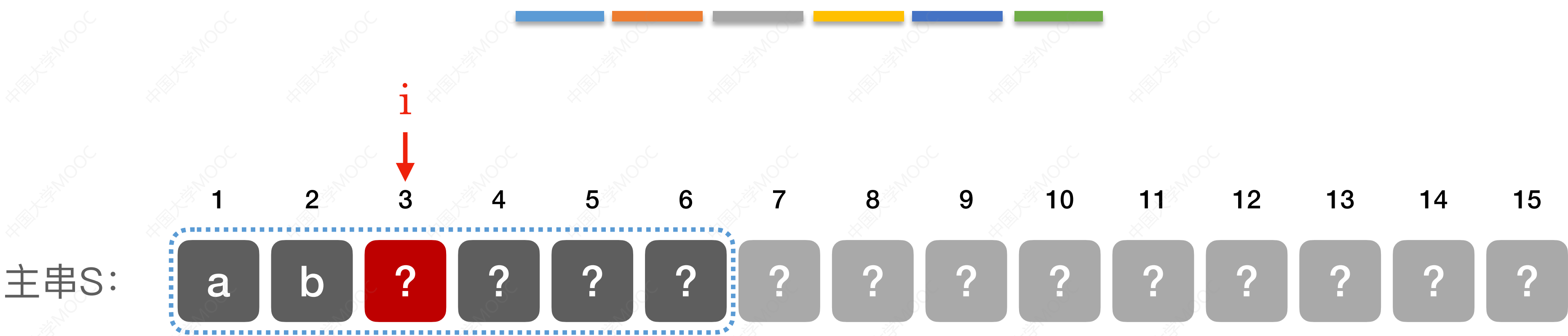


对于模式串 T = 'abaabc'，当第3个元素匹配失败时？怎么搞？

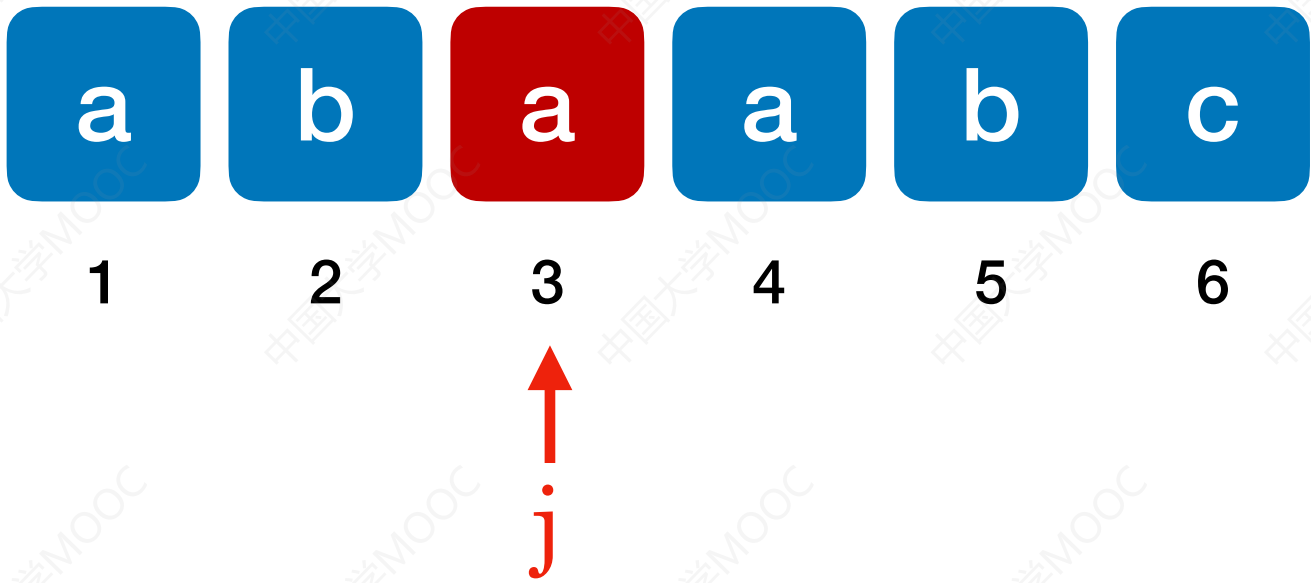
如果其他位置不匹配呢？



如果其他位置不匹配呢？

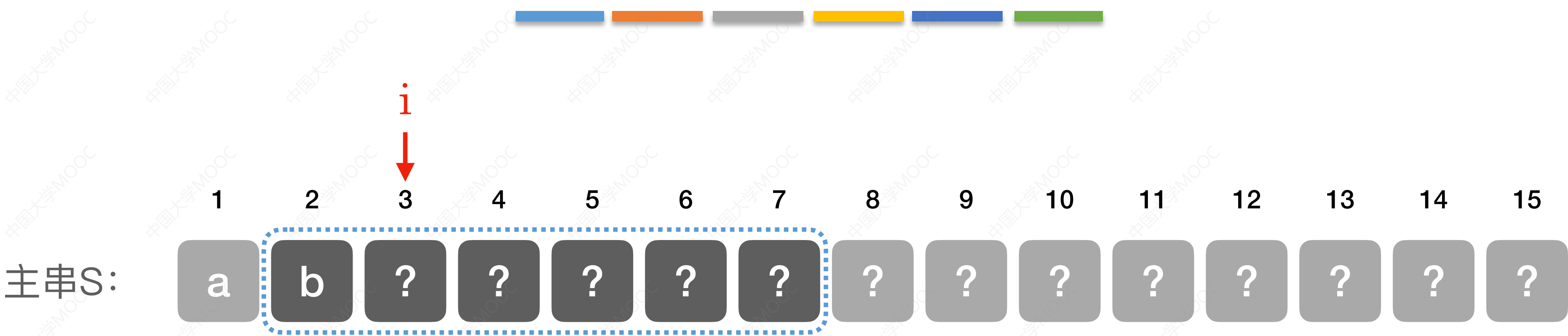


模式串T:

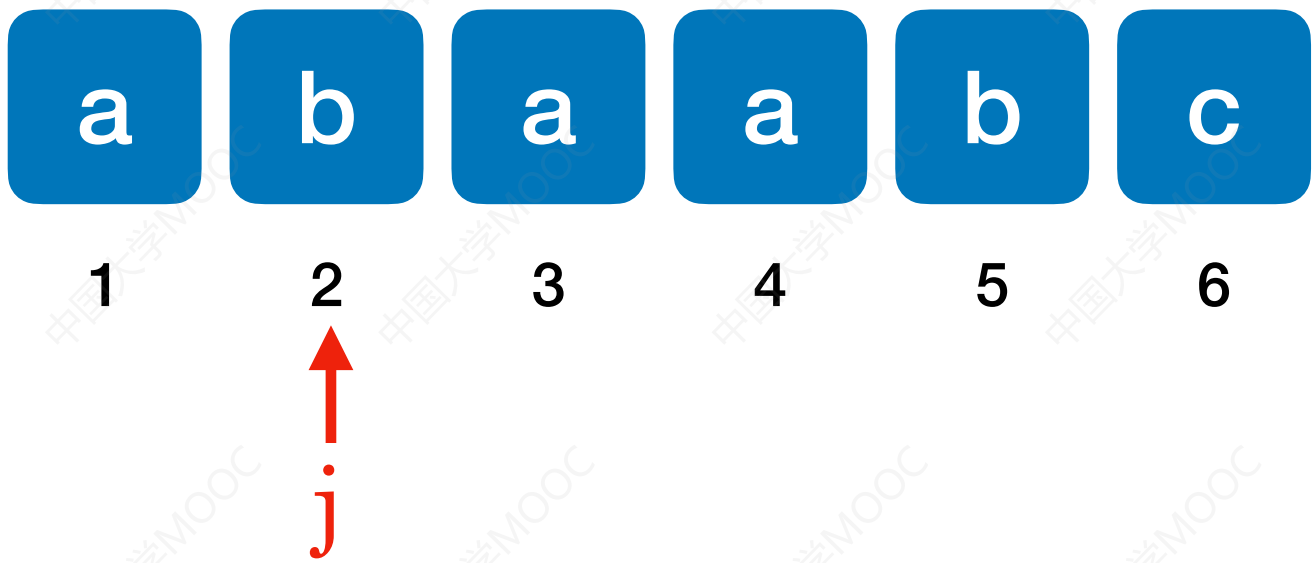


对于模式串 T = 'abaabc'，当第3个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？

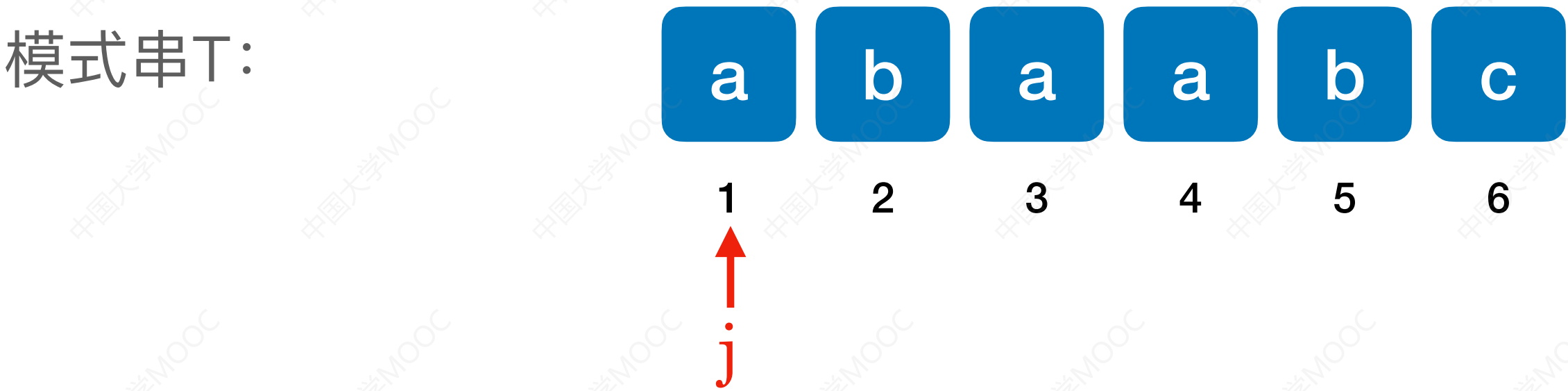
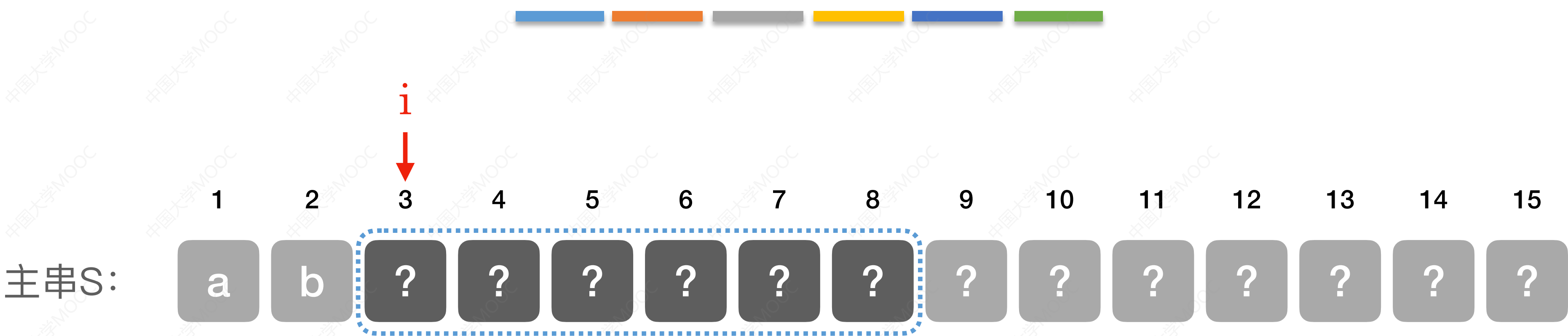


模式串T:



对于模式串 T = 'abaabc'，当第3个元素匹配失败时？怎么搞？

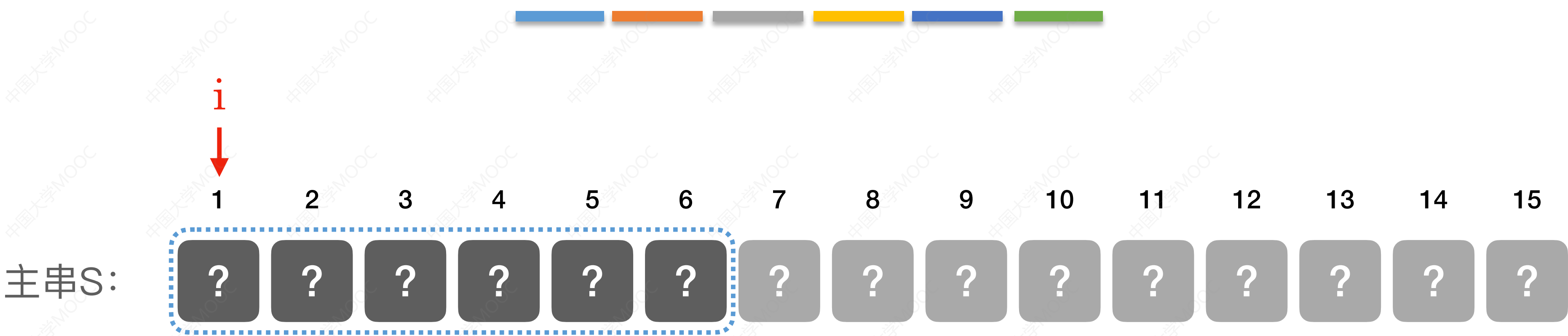
如果其他位置不匹配呢？



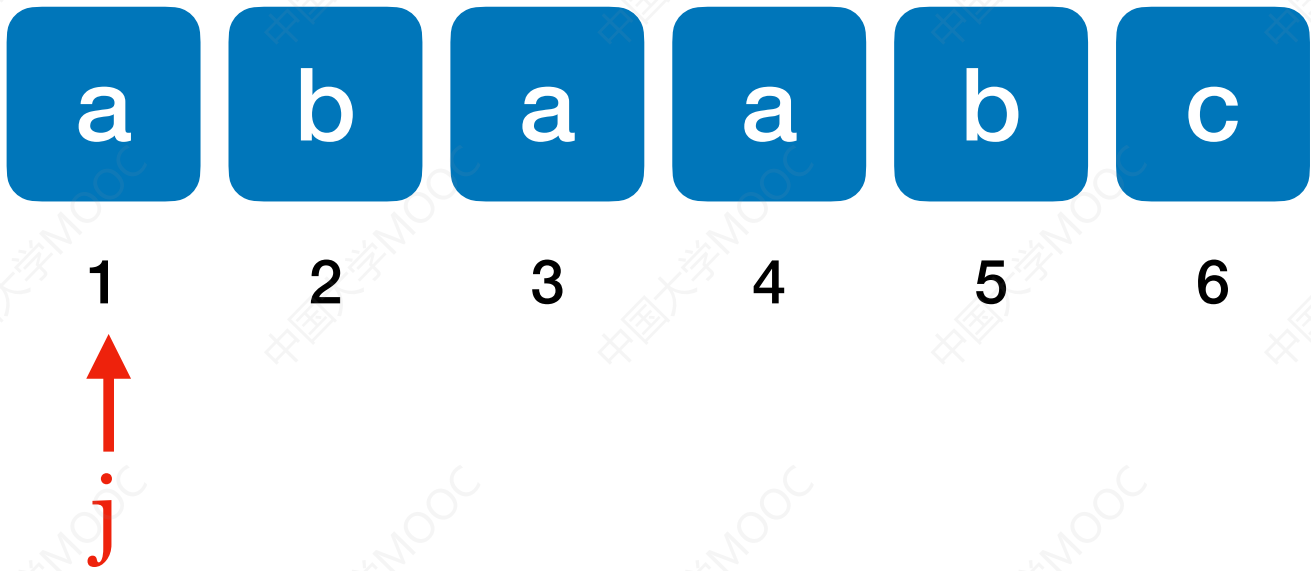
对于模式串 $T = \text{'abaabc'}$ ，当第3个元素匹配失败时？怎么搞？

可令主串指针 i 不变，模式串指针 $j = 1$

如果其他位置不匹配呢？

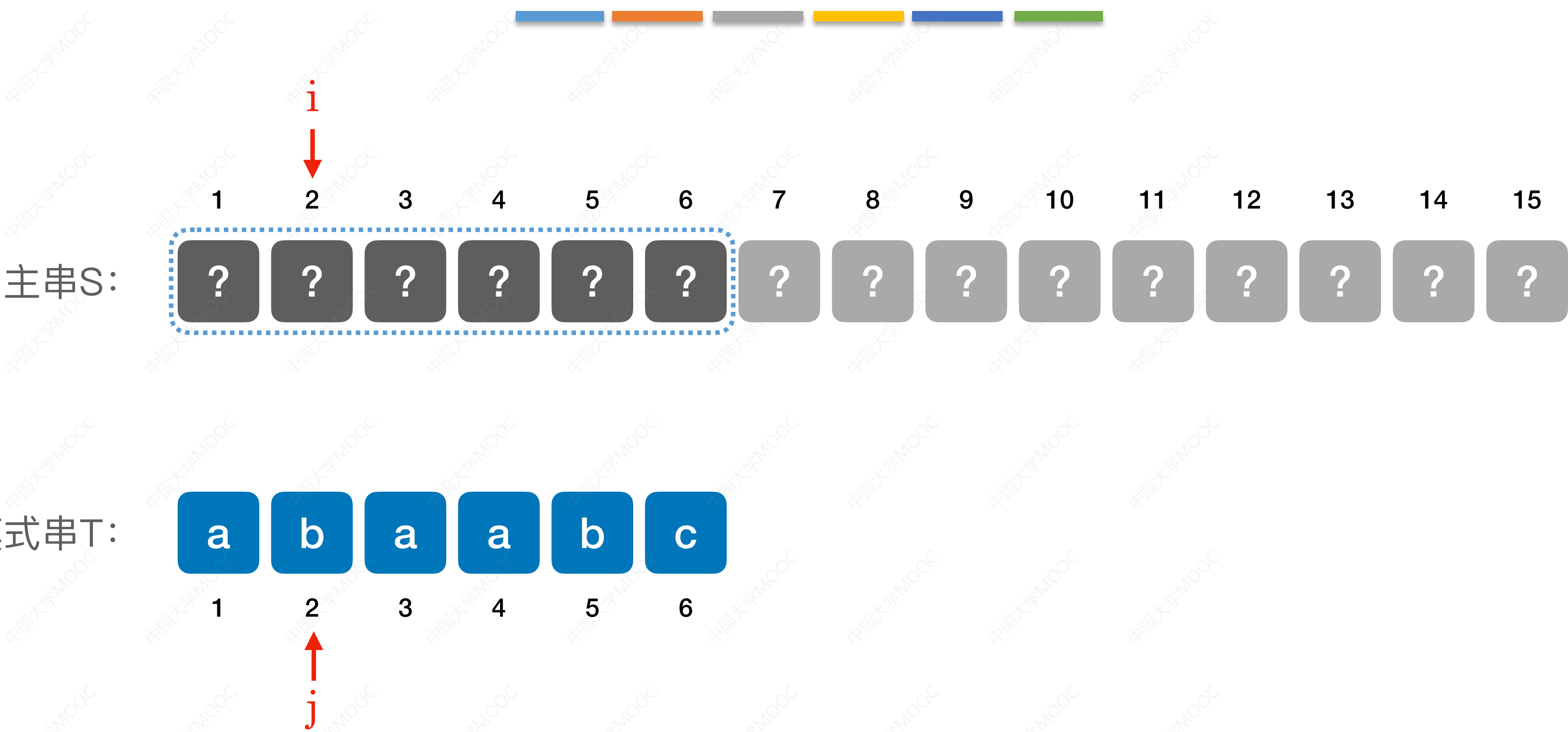


模式串T:



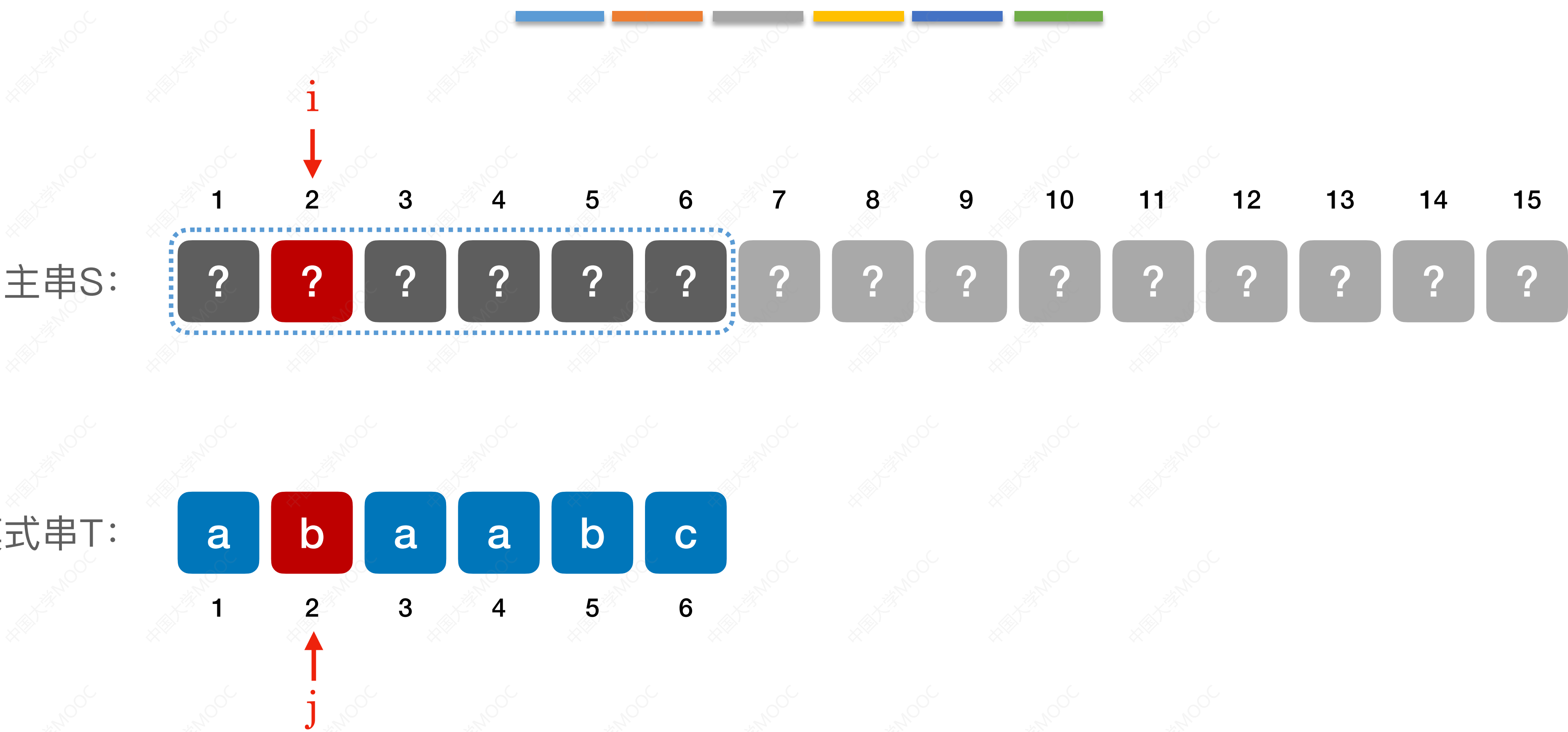
对于模式串 T = 'abaabc'，当第2个元素匹配失败时？怎么搞？

如果其他位置不匹配呢？

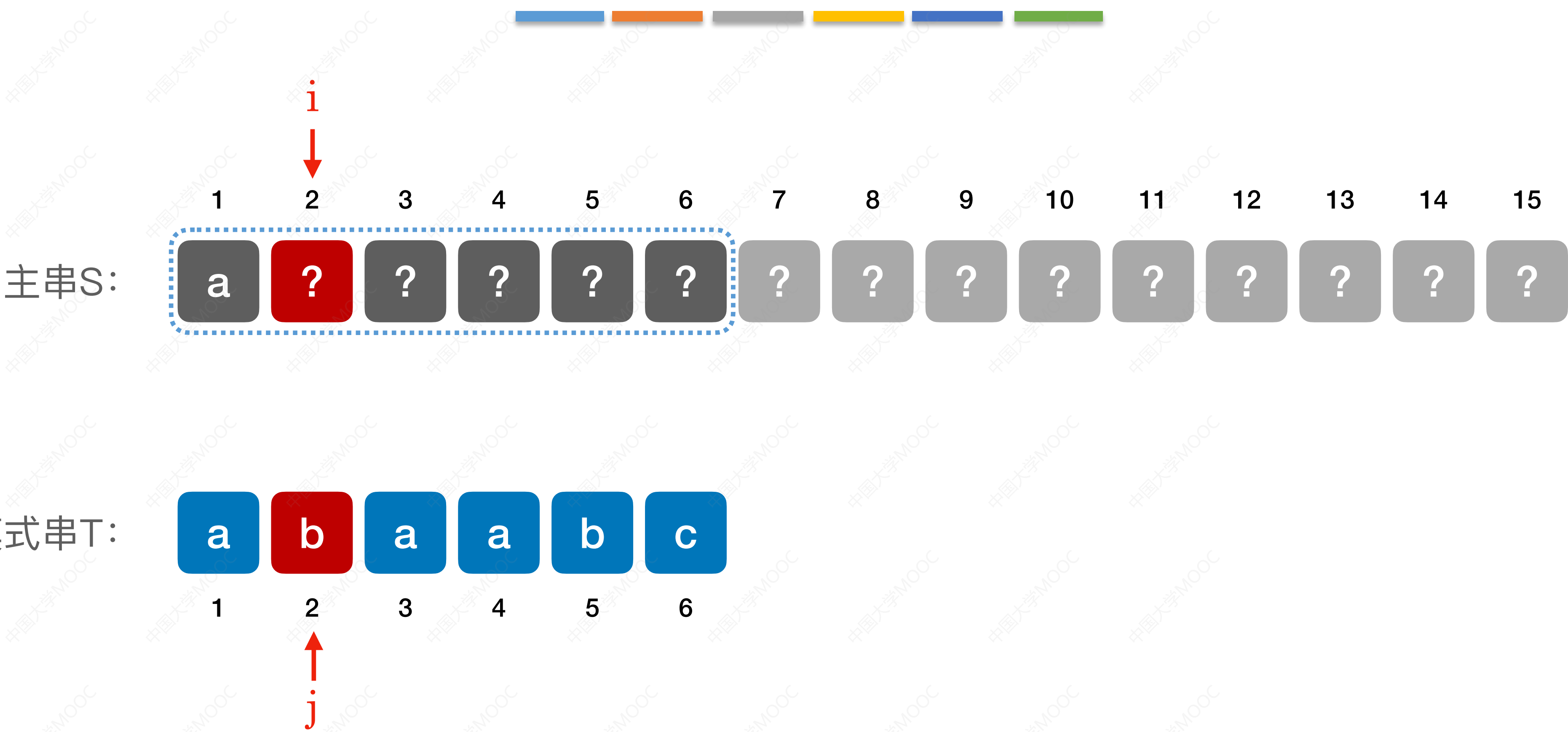


对于模式串 T = 'abaabc'，当第2个元素匹配失败时？怎么搞？

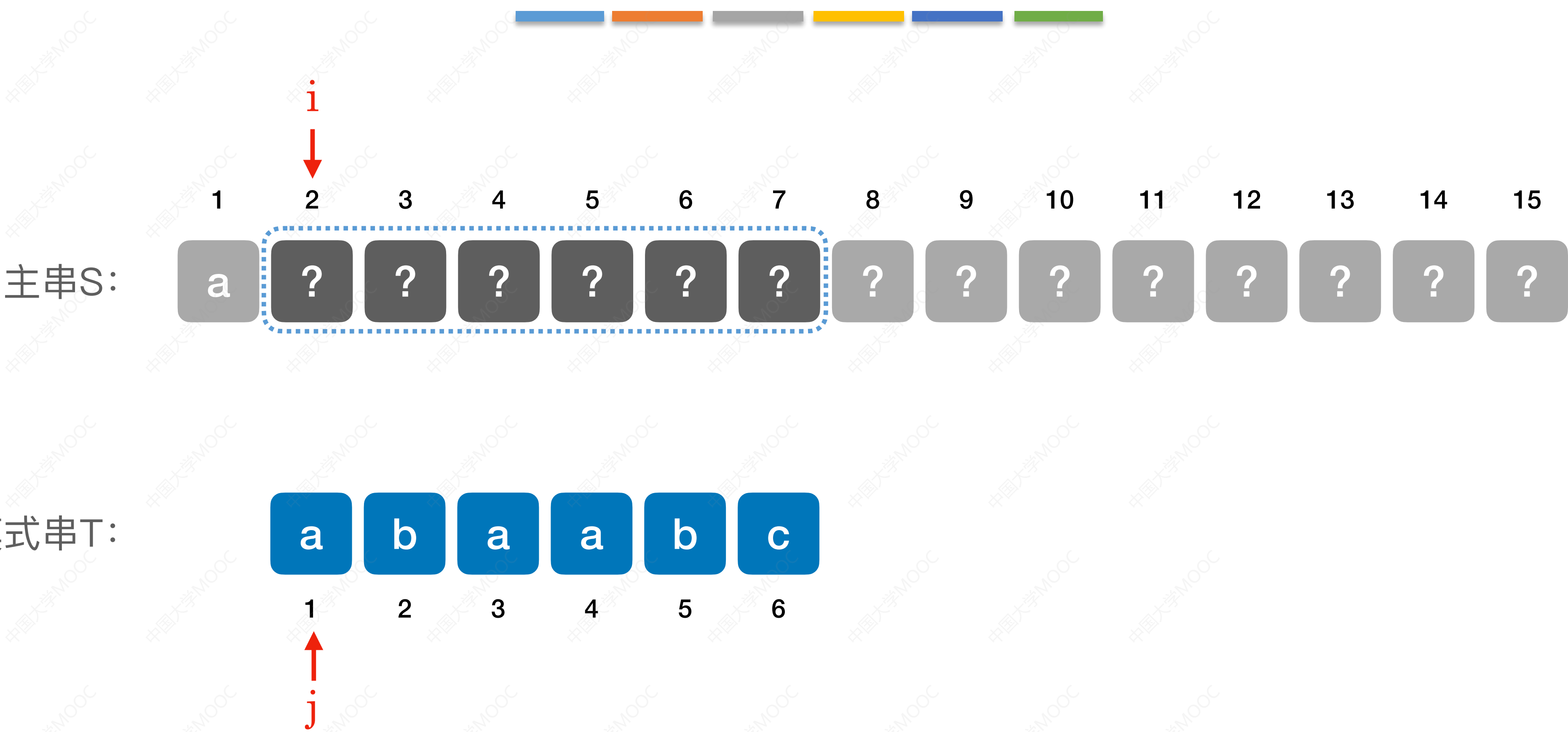
如果其他位置不匹配呢？



如果其他位置不匹配呢？



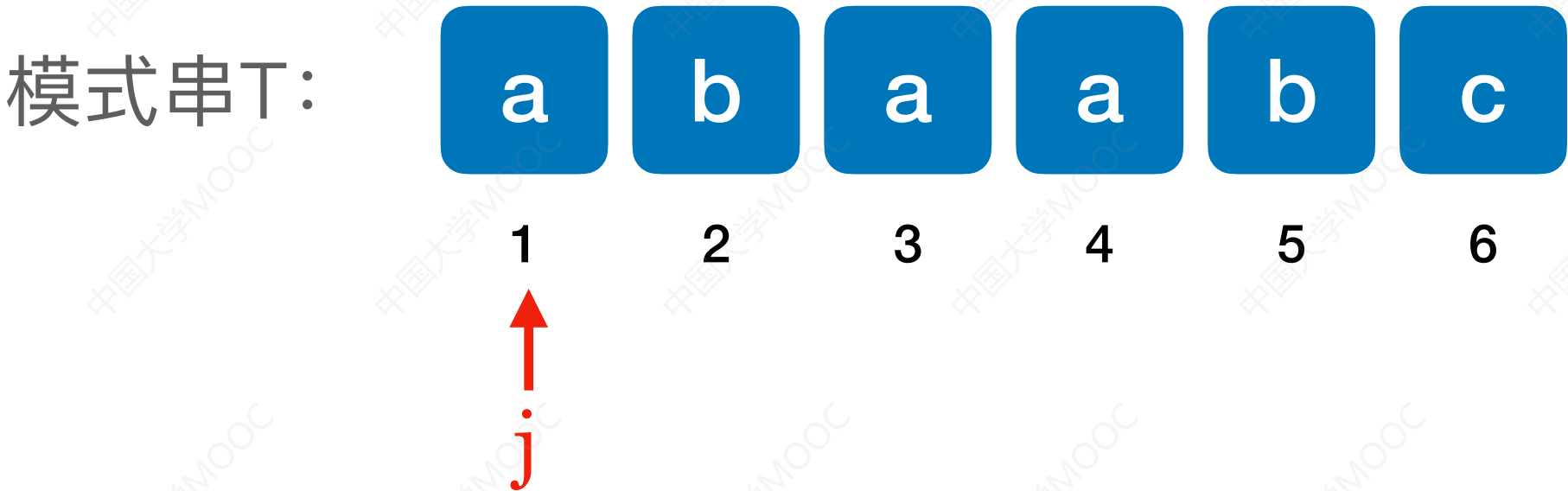
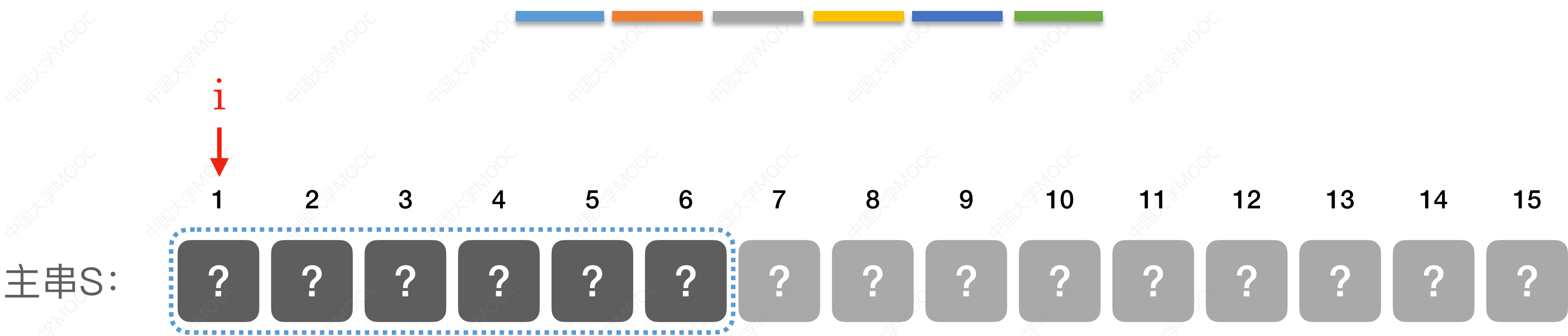
如果其他位置不匹配呢？



对于模式串 T = 'abaabc'，当第2个元素匹配失败时？怎么搞？

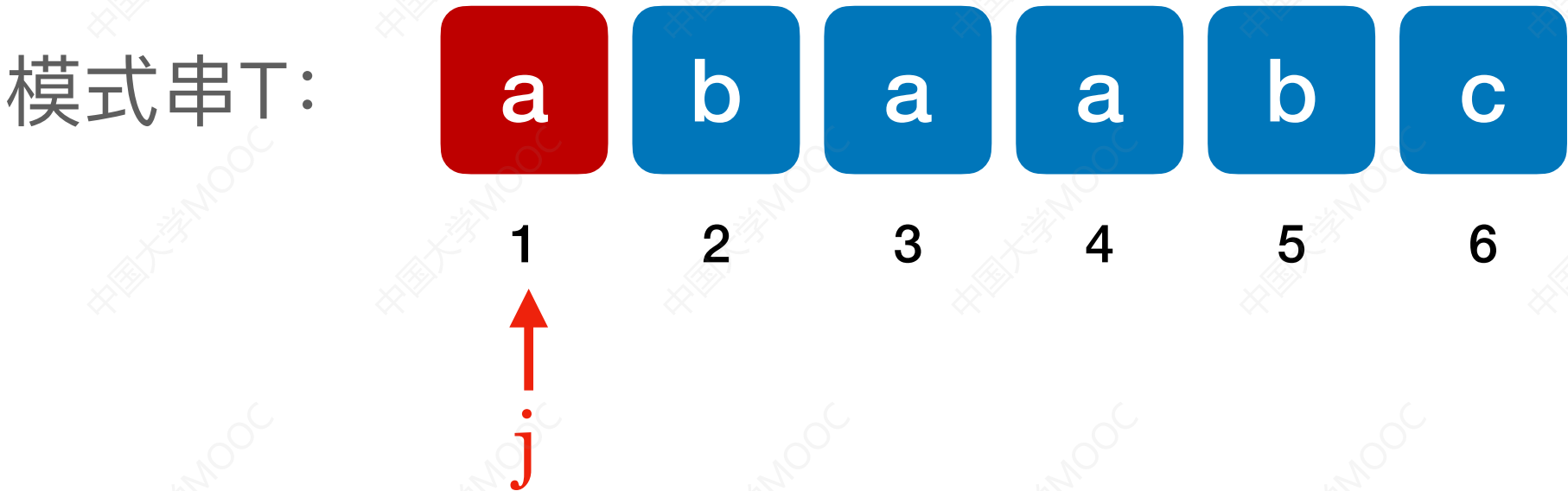
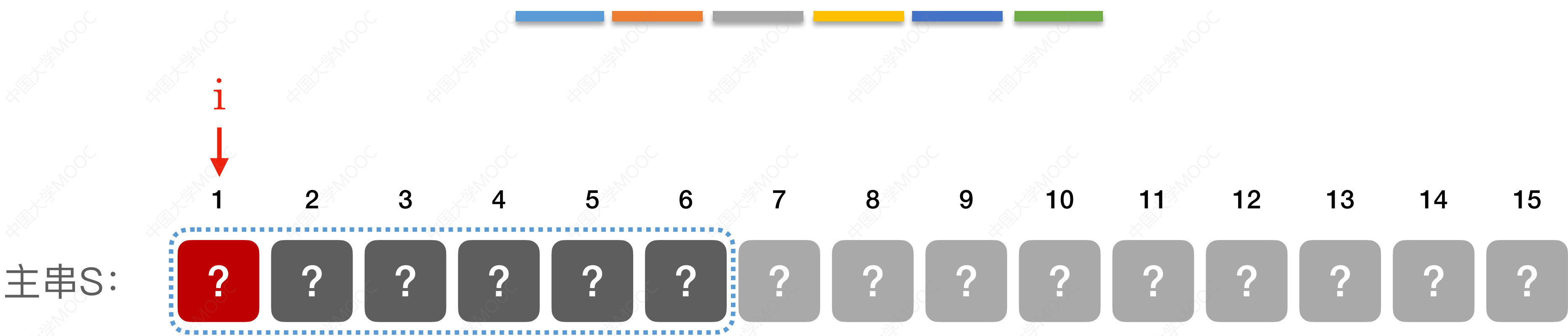
可令主串指针 i 不变，模式串指针 j = 1

如果其他位置不匹配呢？



对于模式串 T = 'abaabc'，当第1个元素匹配失败时？怎么搞？

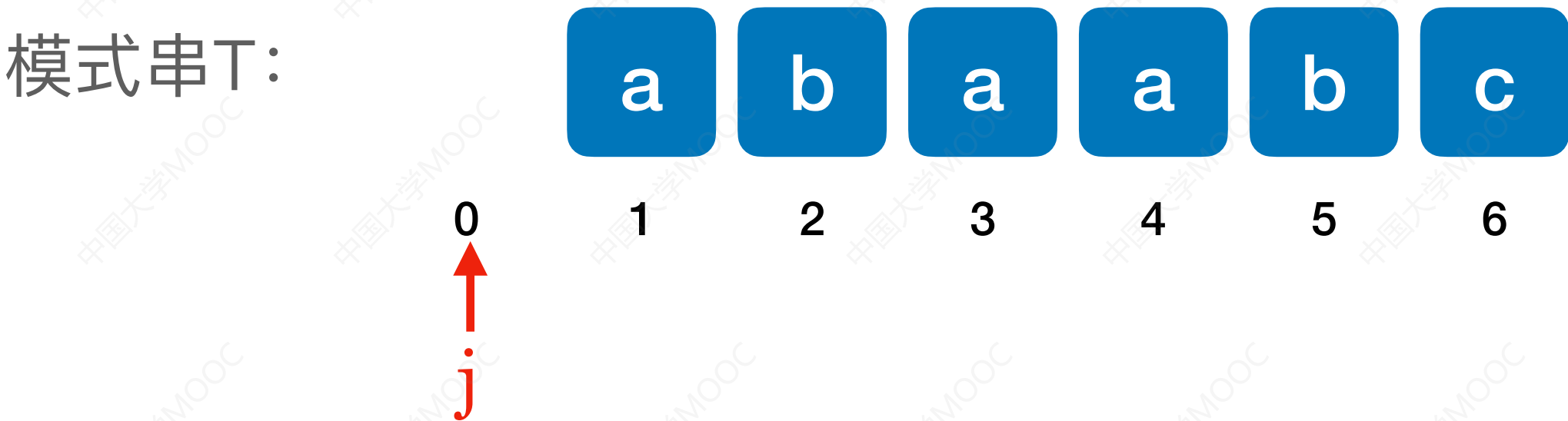
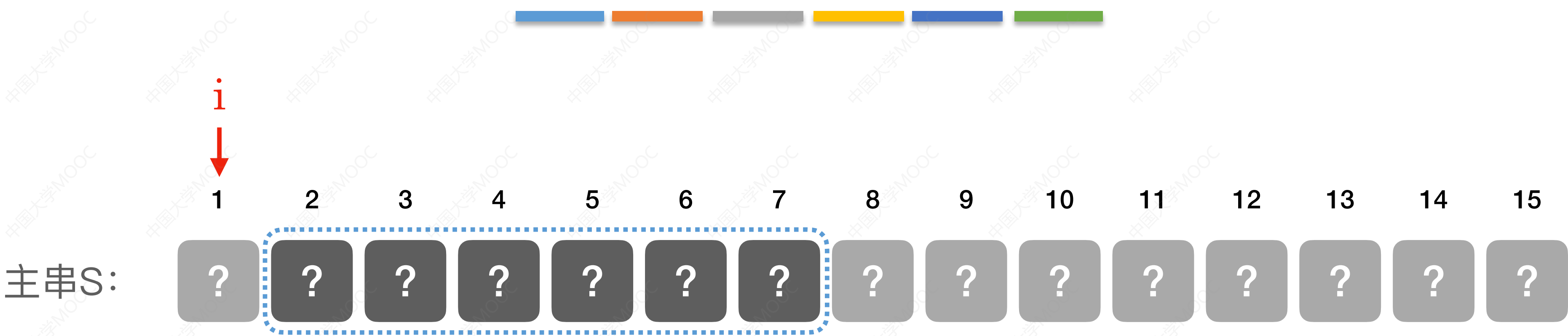
如果其他位置不匹配呢？



对于模式串 T = 'abaabc'，当第1个元素匹配失败时？怎么搞？

匹配下一个相邻子串

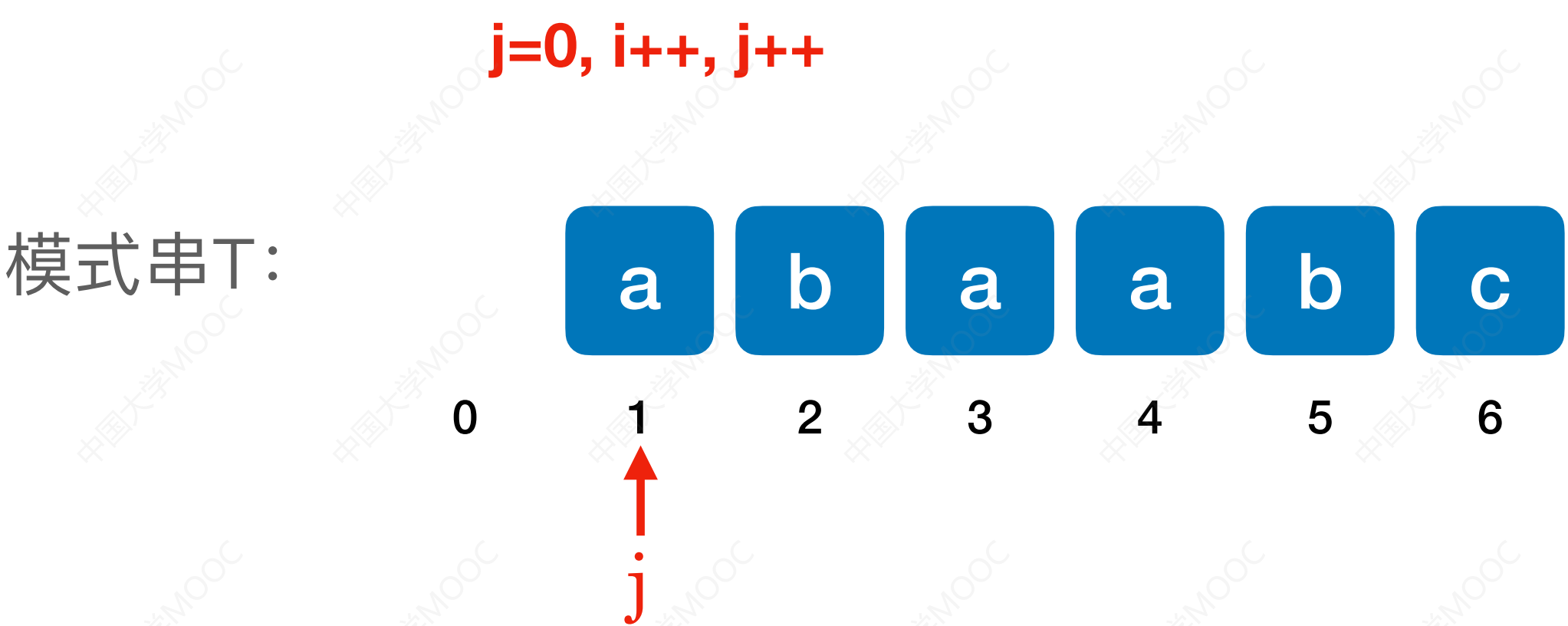
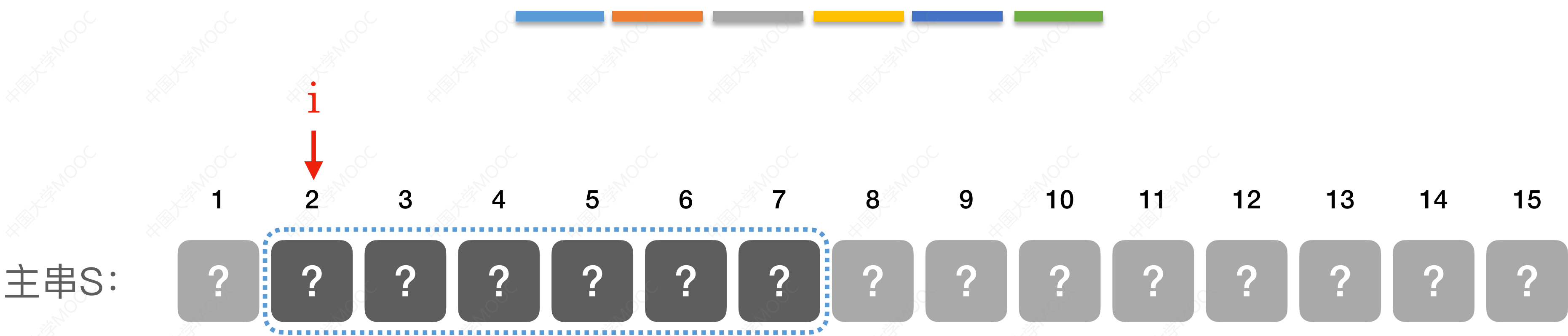
如果其他位置不匹配呢？



对于模式串 T = 'abaabc'，当第1个元素匹配失败时？怎么搞？

匹配下一个相邻子串

如果其他位置不匹配呢？



对于模式串 T = 'abaabc'，当第1个元素匹配失败时？怎么搞？

匹配下一个相邻子串

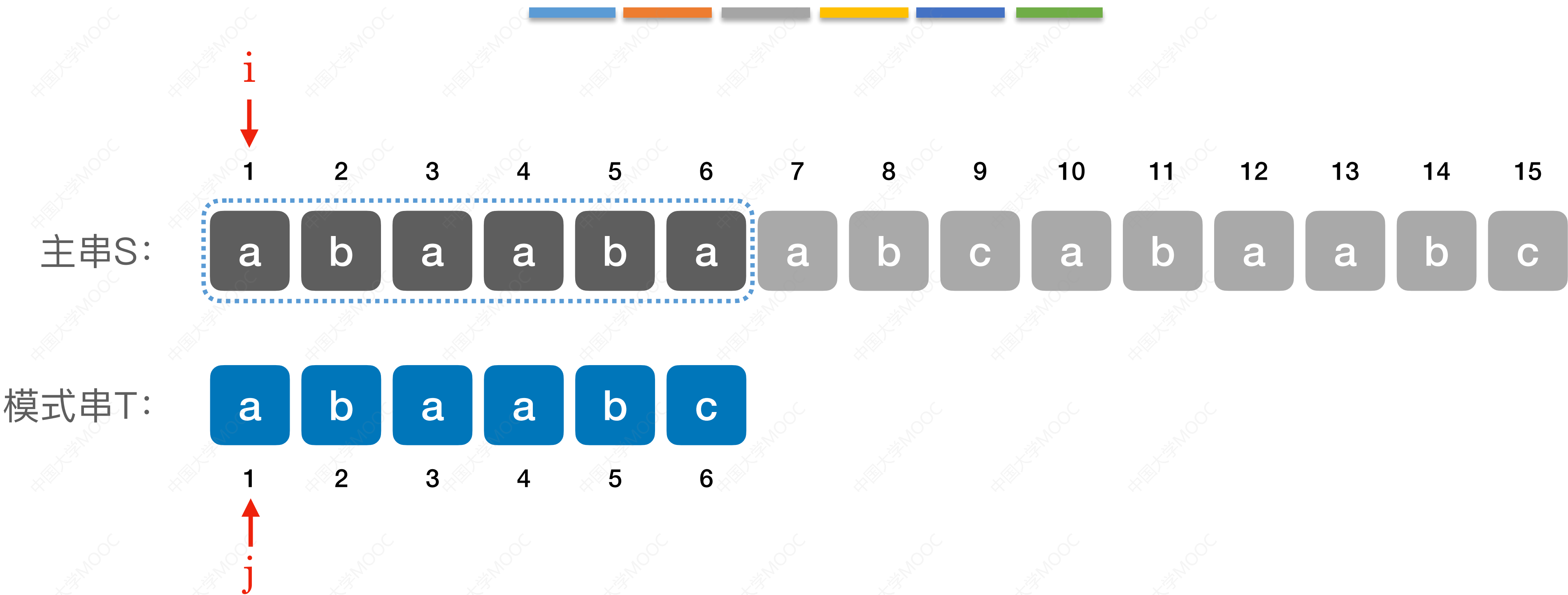
结论



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

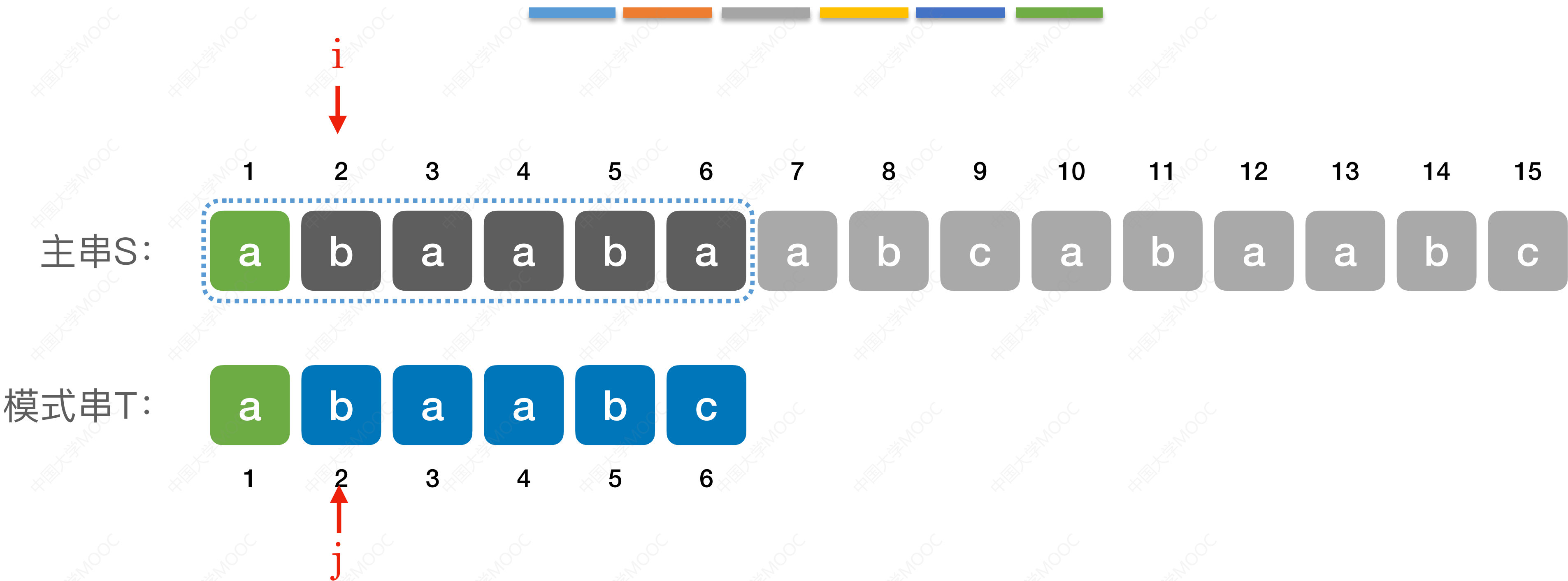
再整一次上一节的例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

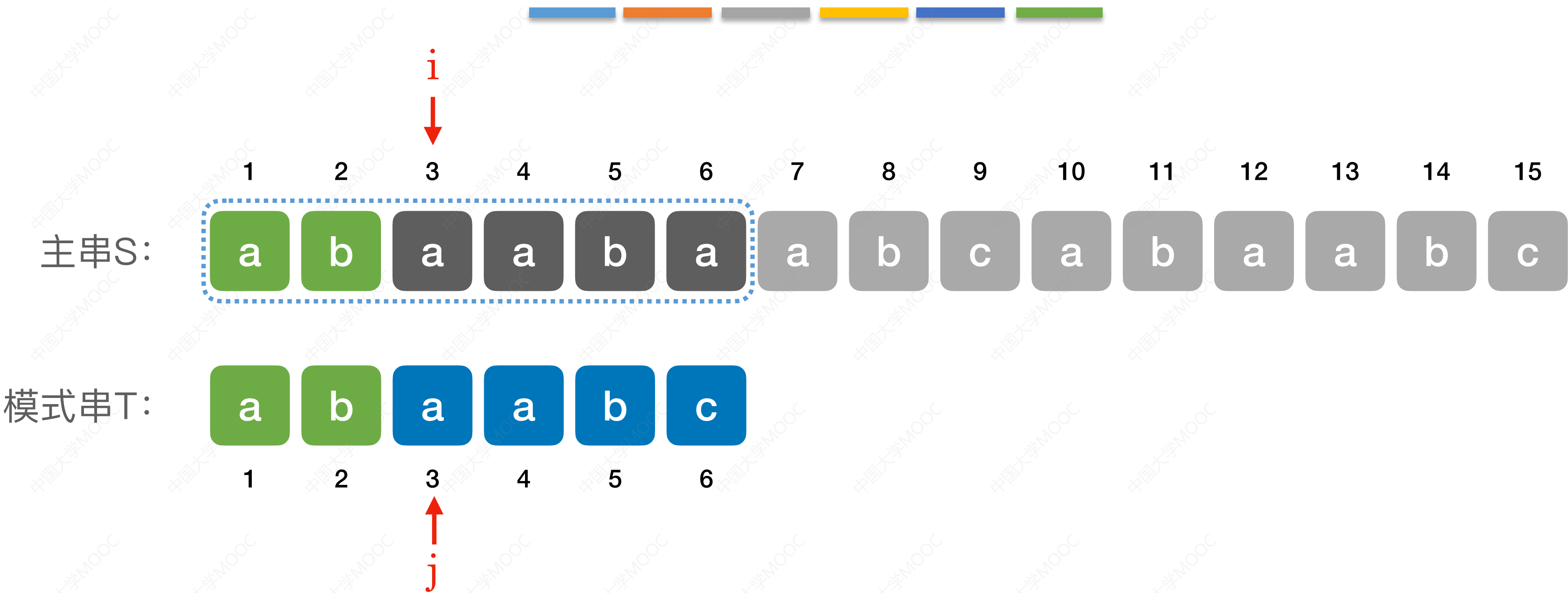
再整一次上一节的例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

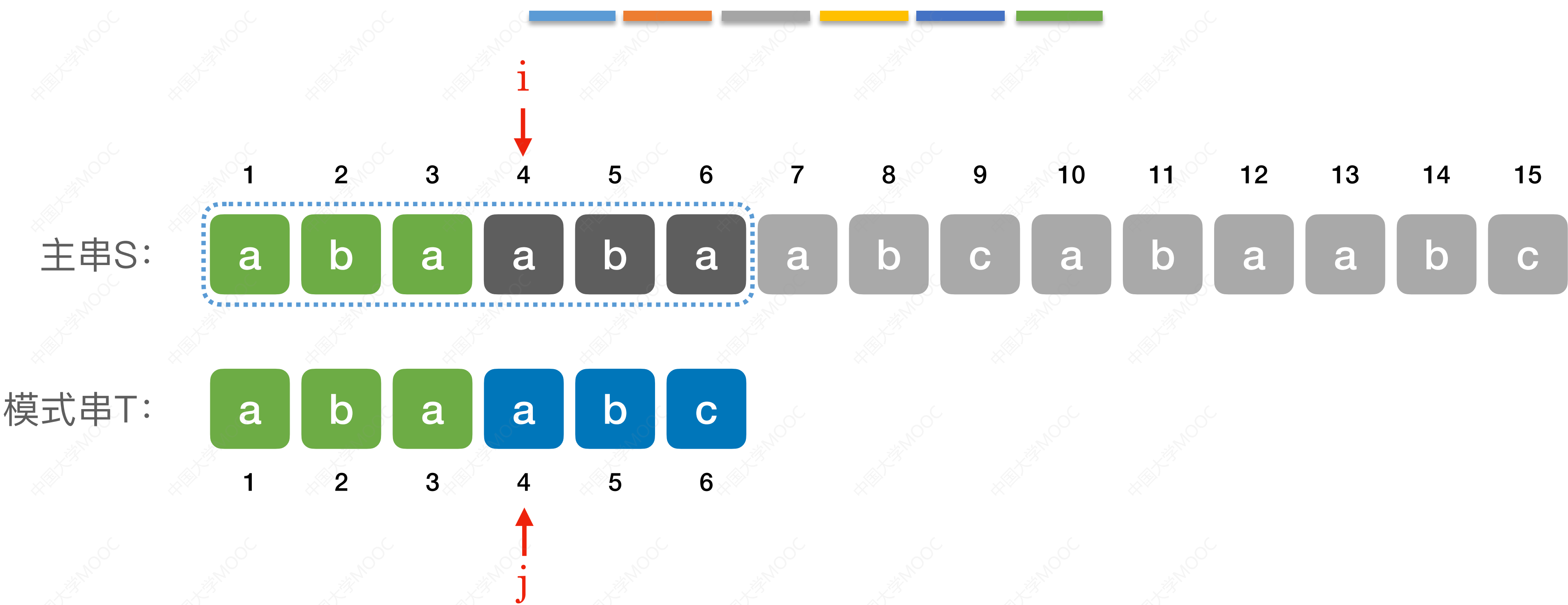
再整一次上一节的例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

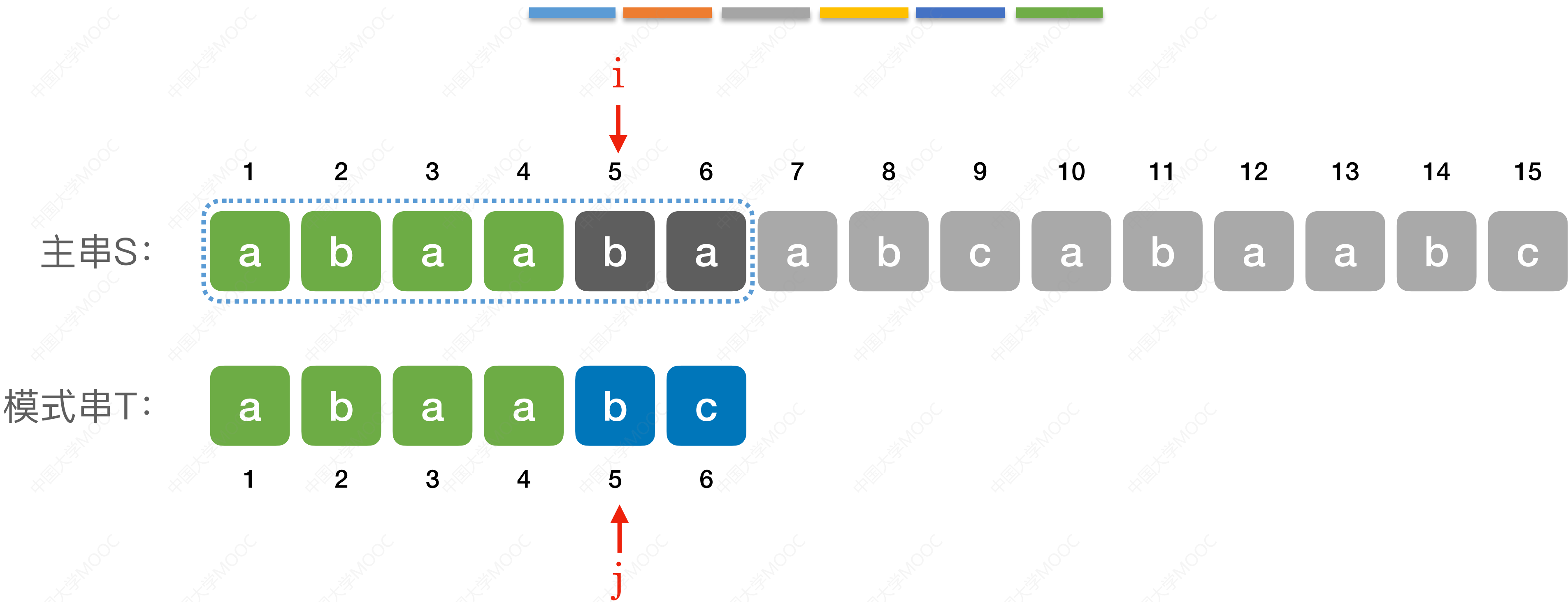
再整一次上一节的例子



对于模式串 $T = 'abaabc'$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

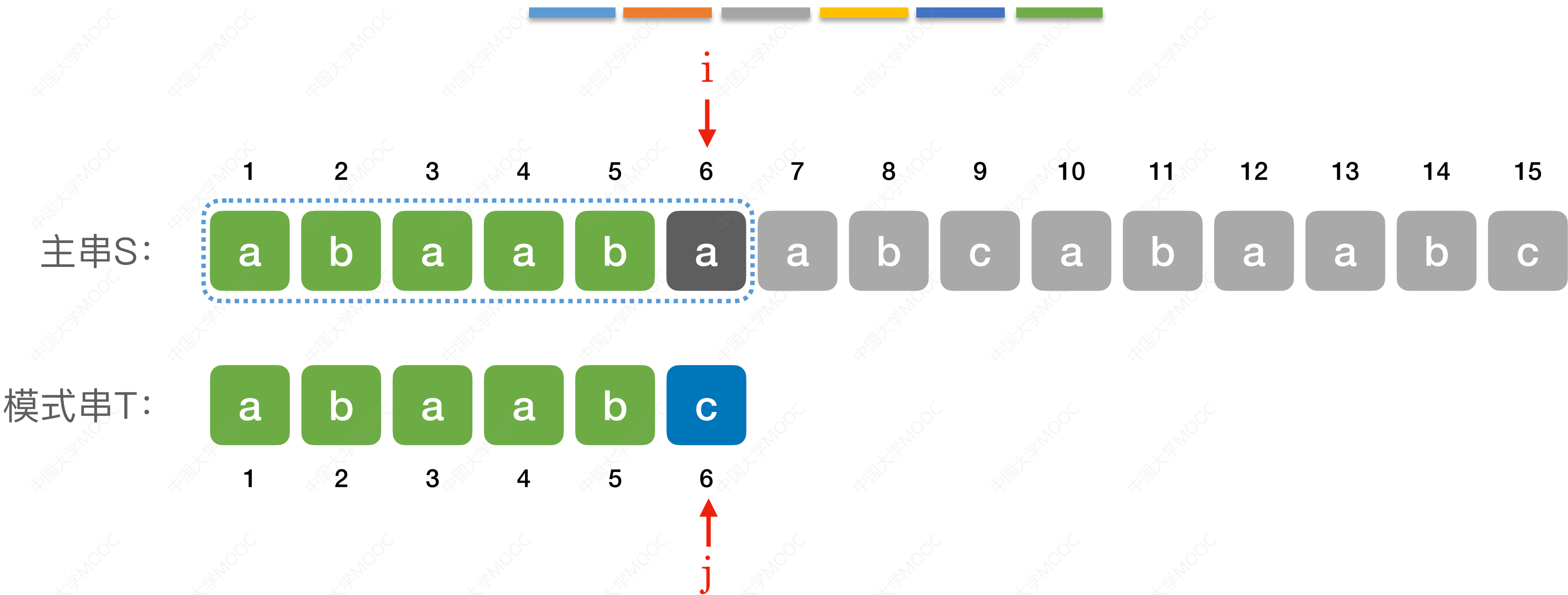
再整一次上一节的例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

再整一次上一节的例子



对于模式串 T = 'abaabc'

当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=3

当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=2

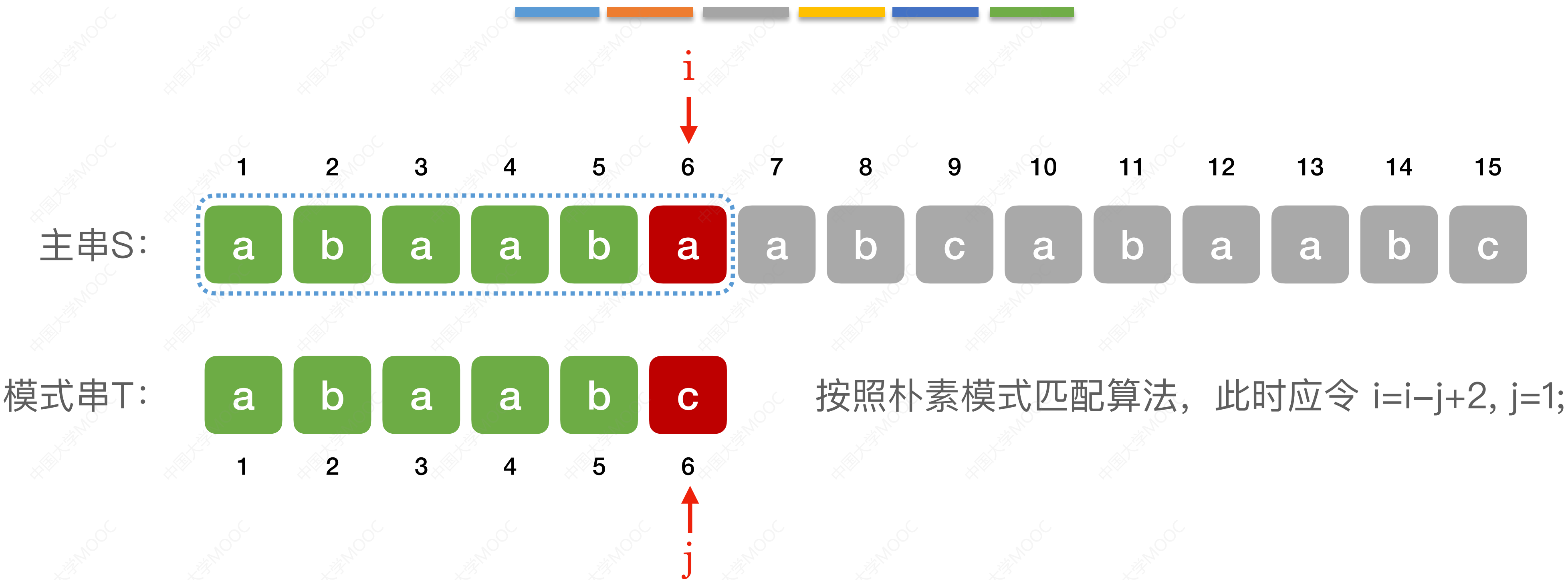
当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=2

当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=1

当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=1

当第1个元素匹配失败时, 匹配下一个相邻子串, 令 j=0, i++, j++

再整一次上一节的例子



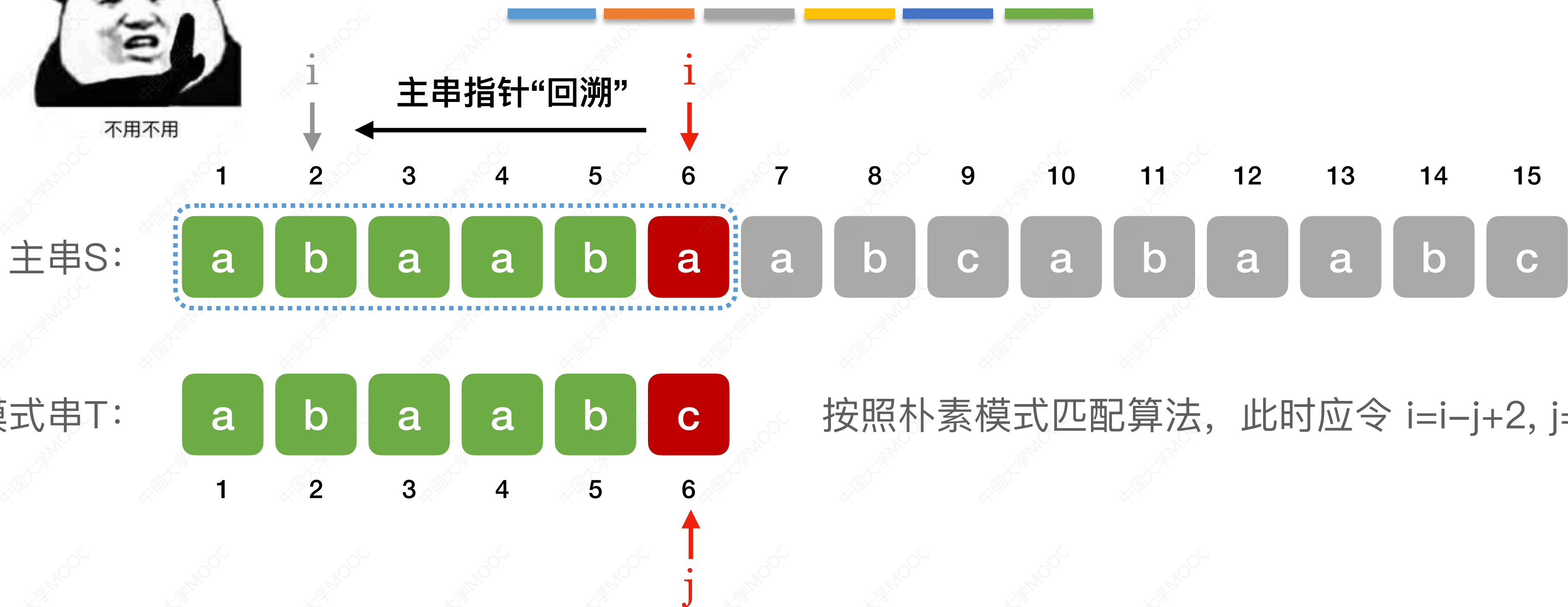
对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0$, $i++$, $j++$



不用不用

再整一次上一节的例子

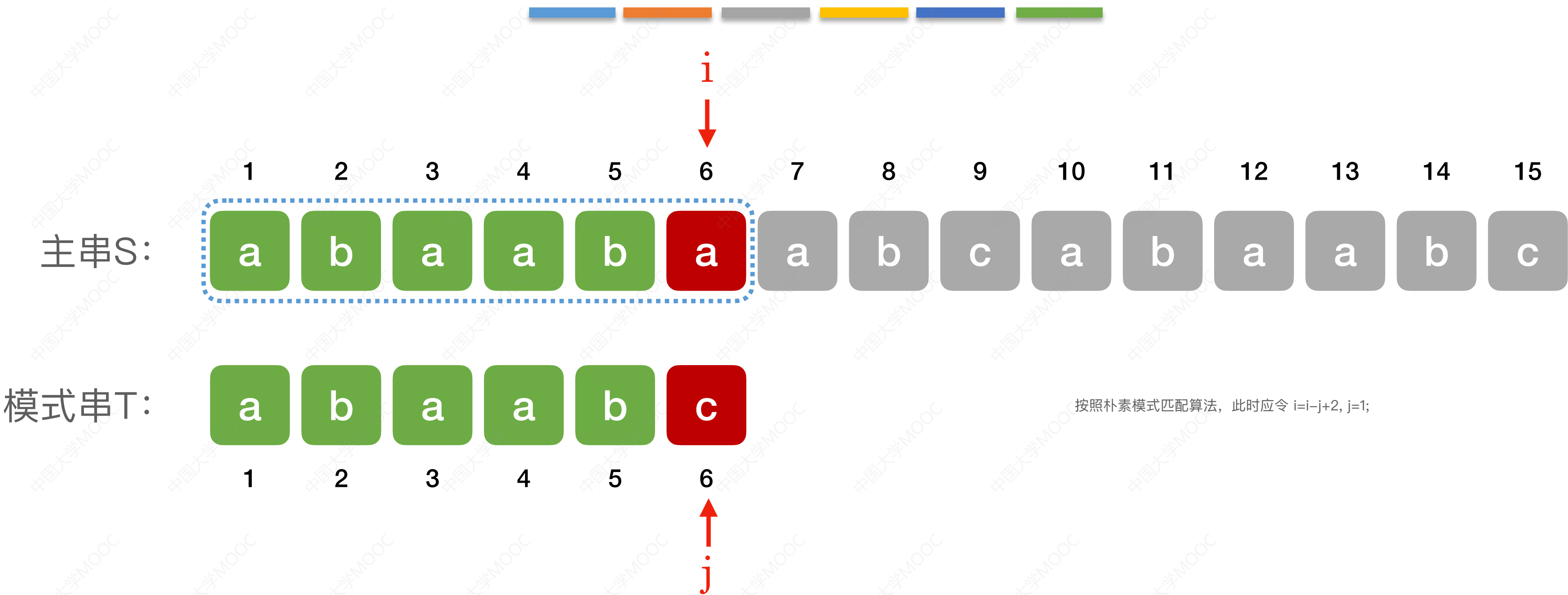


按照朴素模式匹配算法，此时应令 $i=i-j+2$, $j=1$;

对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0$, $i++$, $j++$

再整一次上一节的例子



对于模式串 $T = 'abaabc'$

当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$

当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$

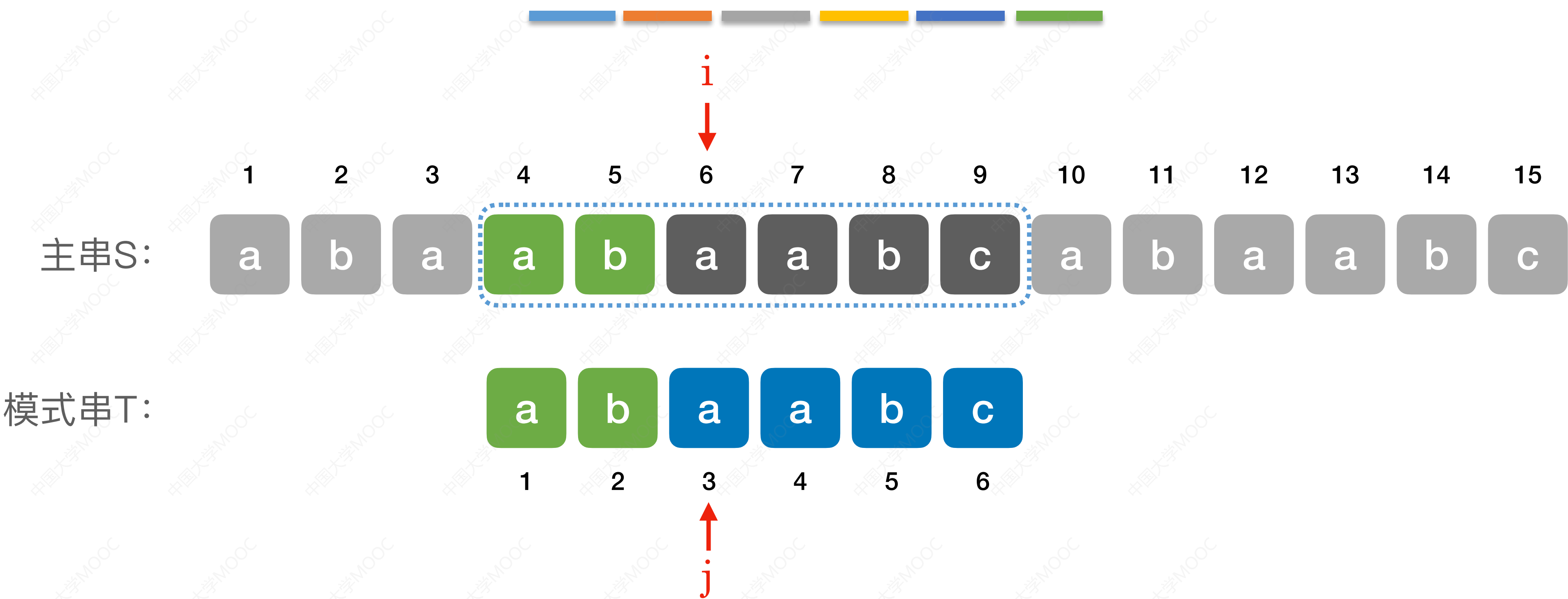
当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$

当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$

当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$

当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

再整一次上一节的例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

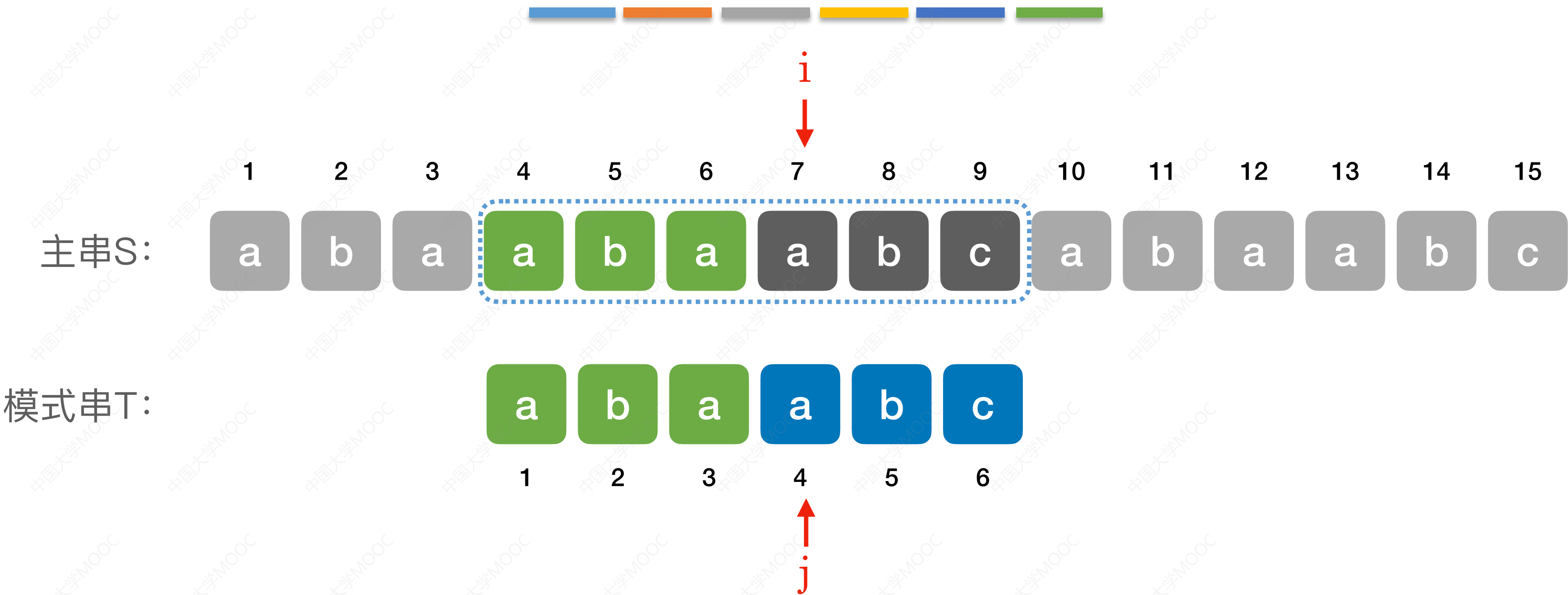
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

再整一次上一节的例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

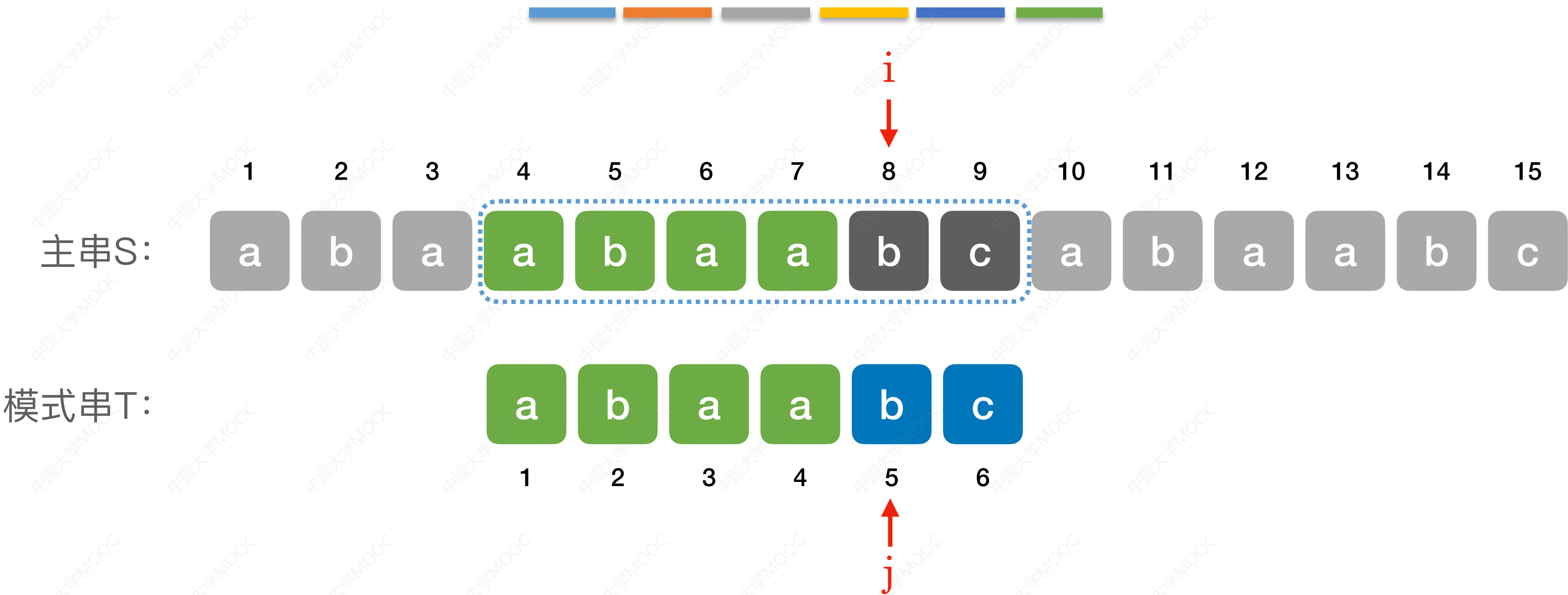
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

再整一次上一节的例子



对于模式串 $T = 'abaabc'$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

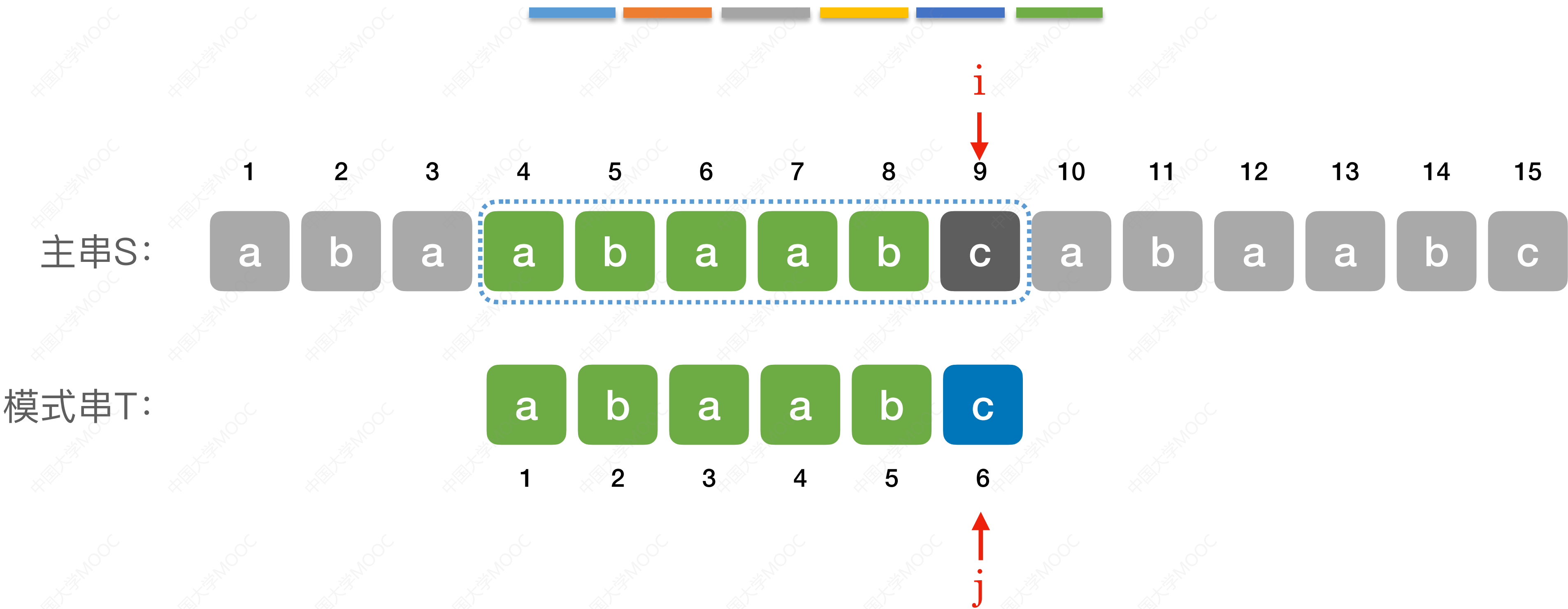
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

再整一次上一节的例子



对于模式串 $T = 'abaabc'$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

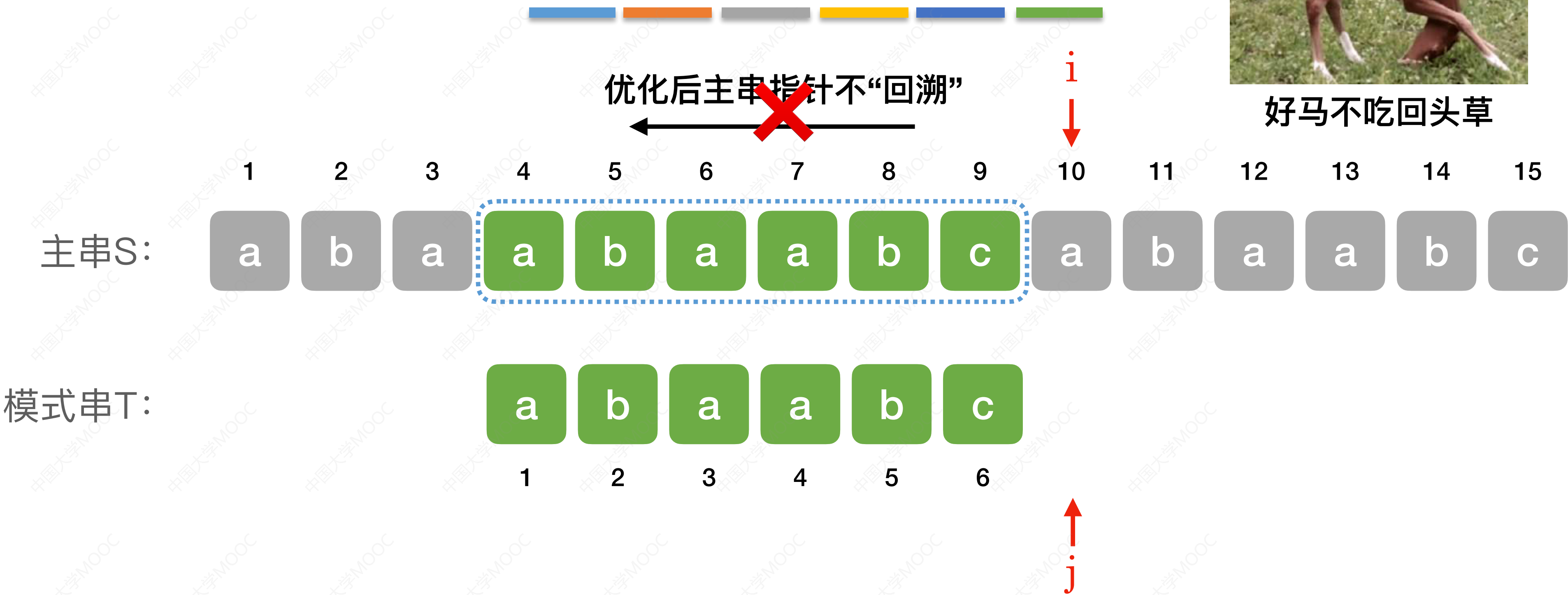
当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

再整一次上一节的例子



好马不吃回头草



掌声送给最棒的你

对于模式串 $T = 'abaabc'$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

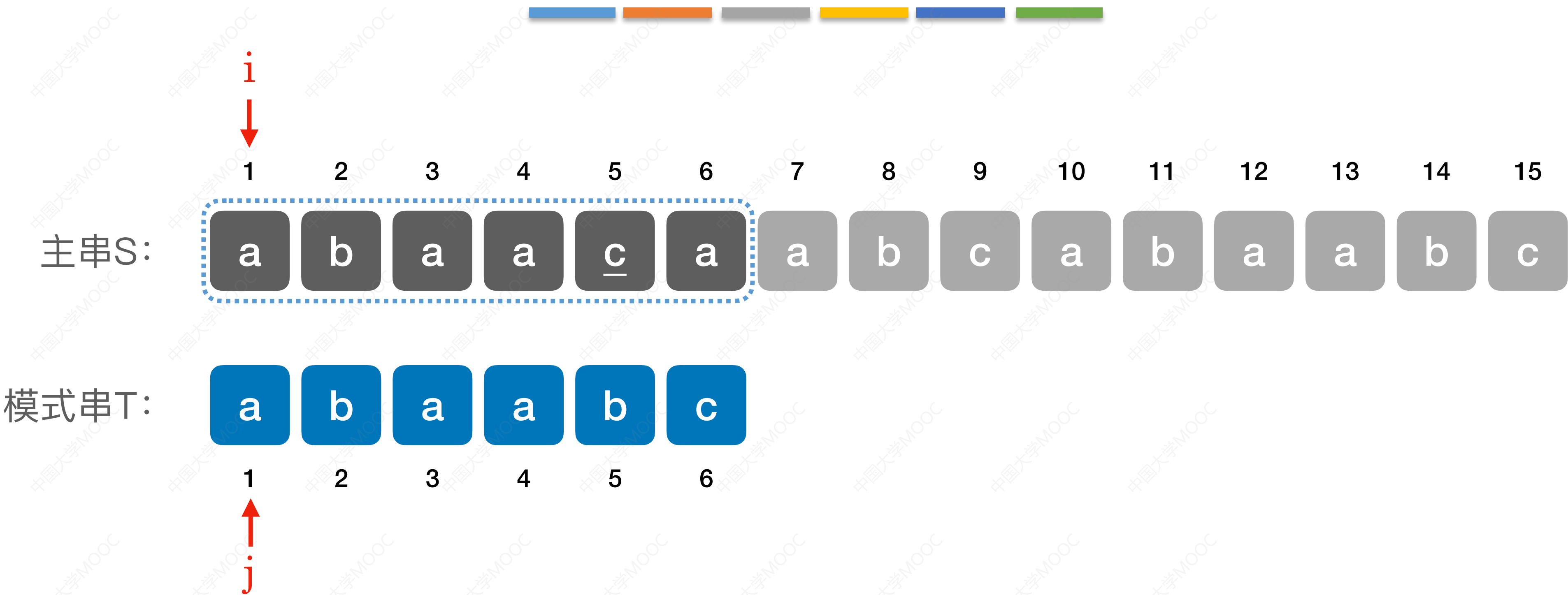
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

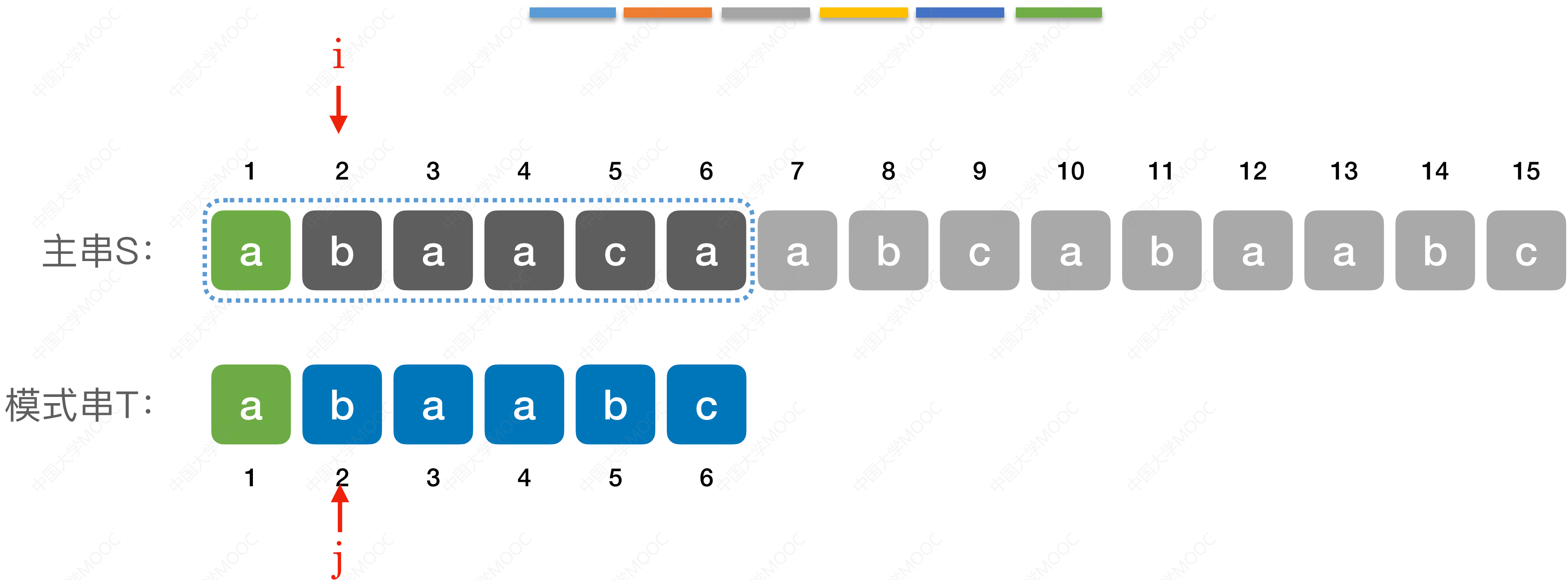
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

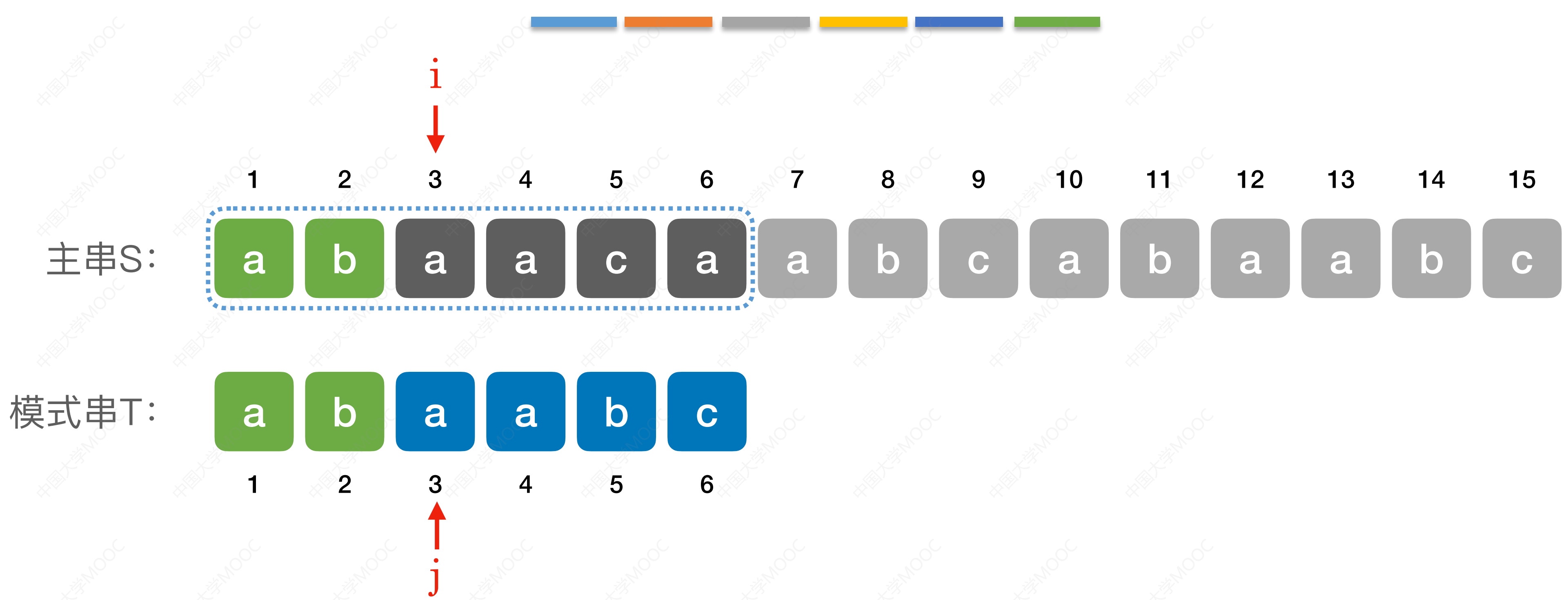
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

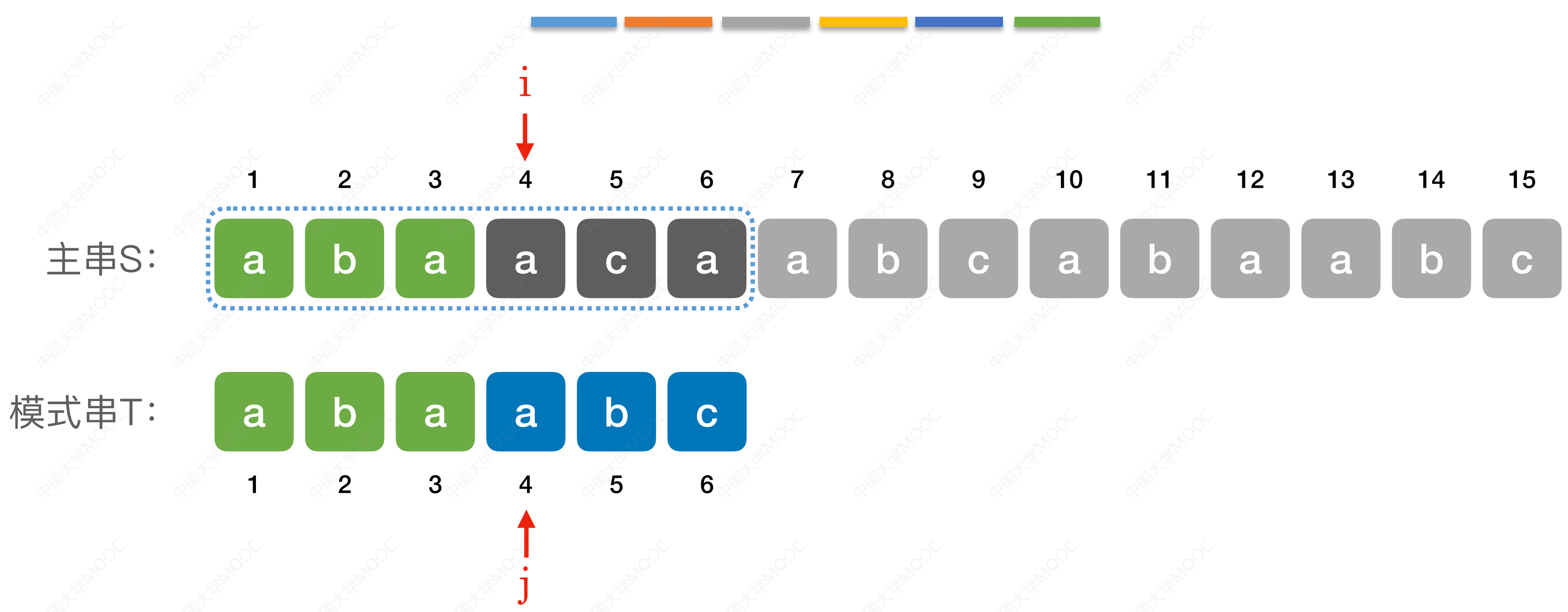
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

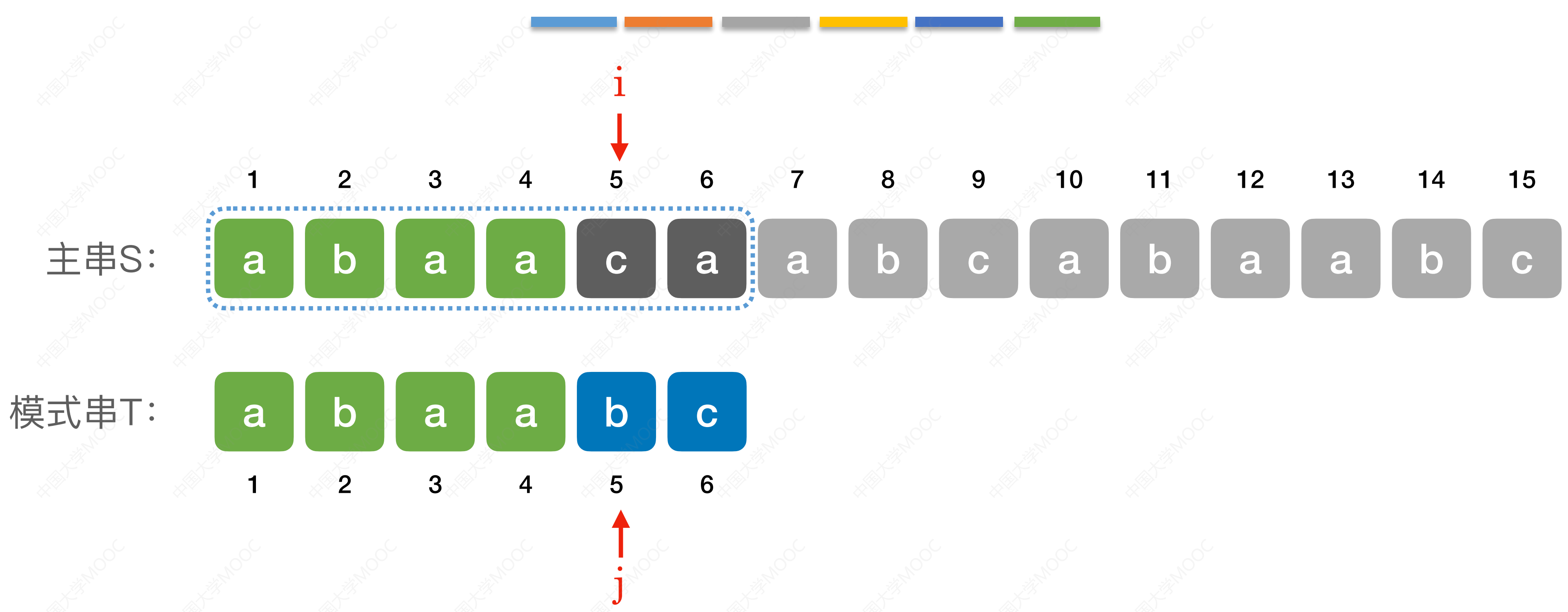
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

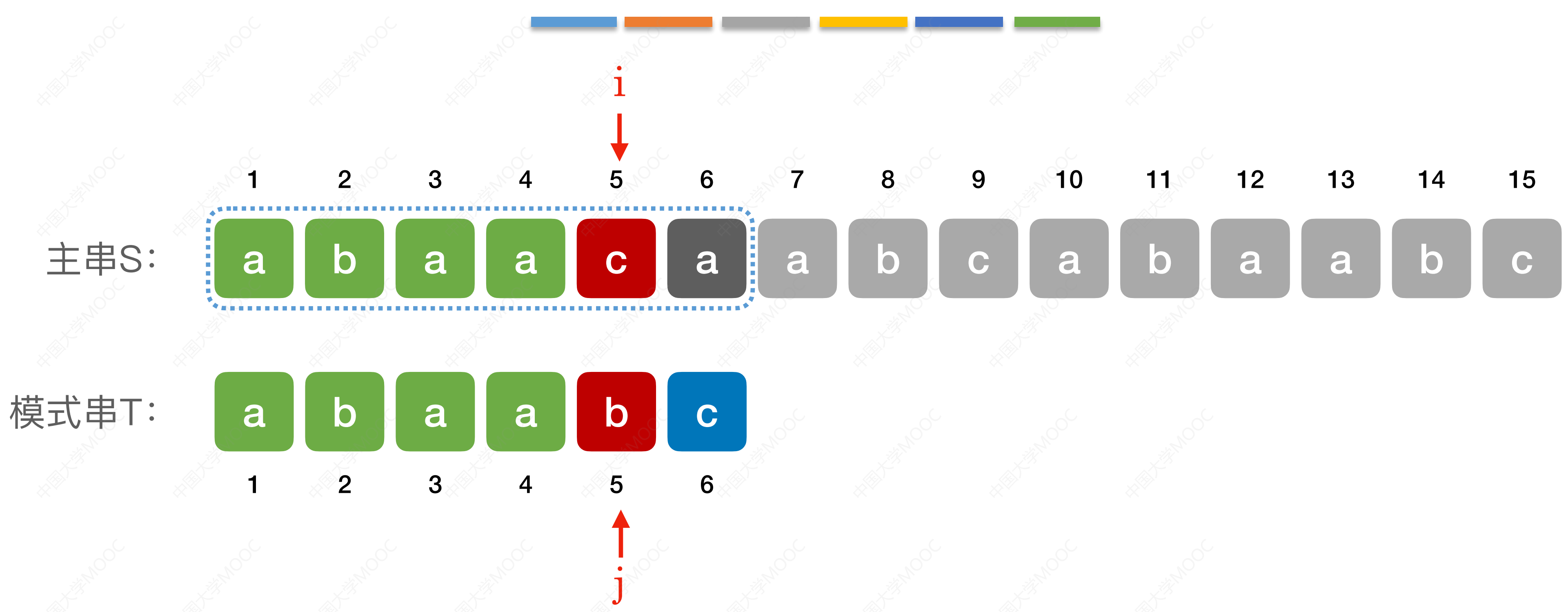
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=3$
- 当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=2$
- 当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 $j=1$
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 $j=0, i++, j++$

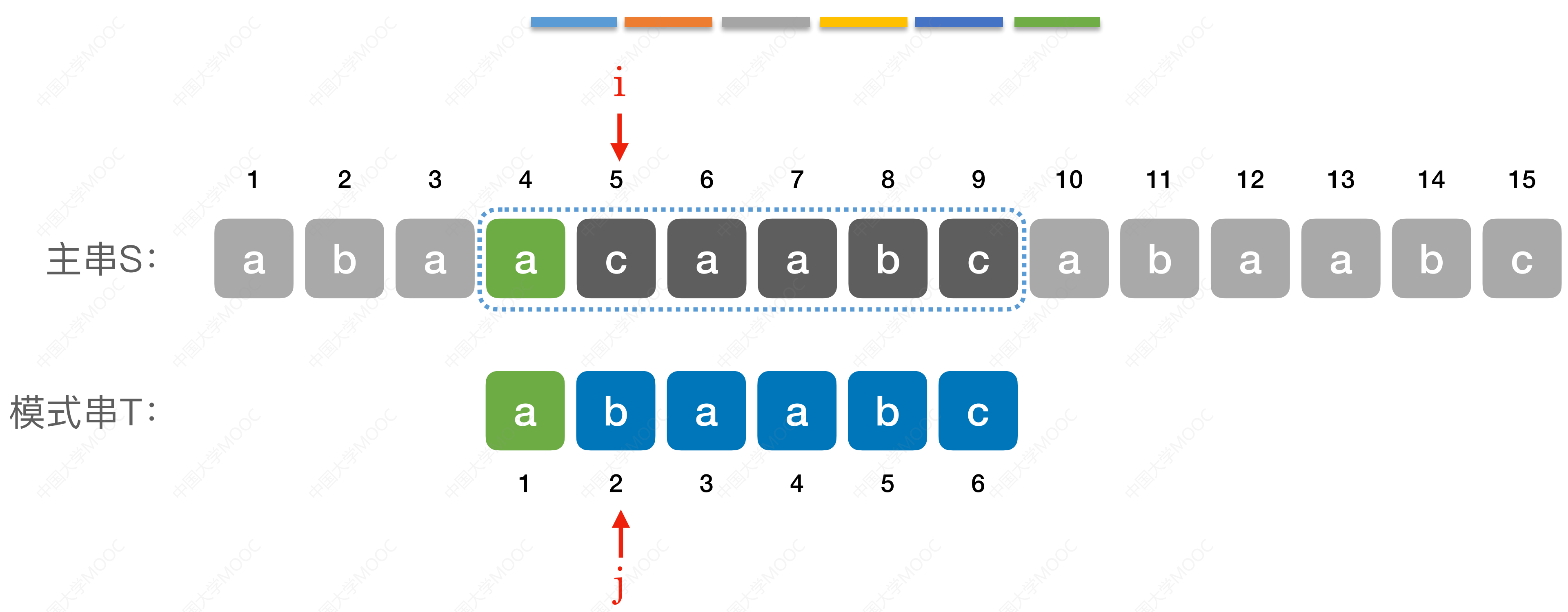
改造一哈例子



对于模式串 $T = 'abaabc'$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

改造一哈例子



对于模式串 T = 'abaabc'

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=3

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=2

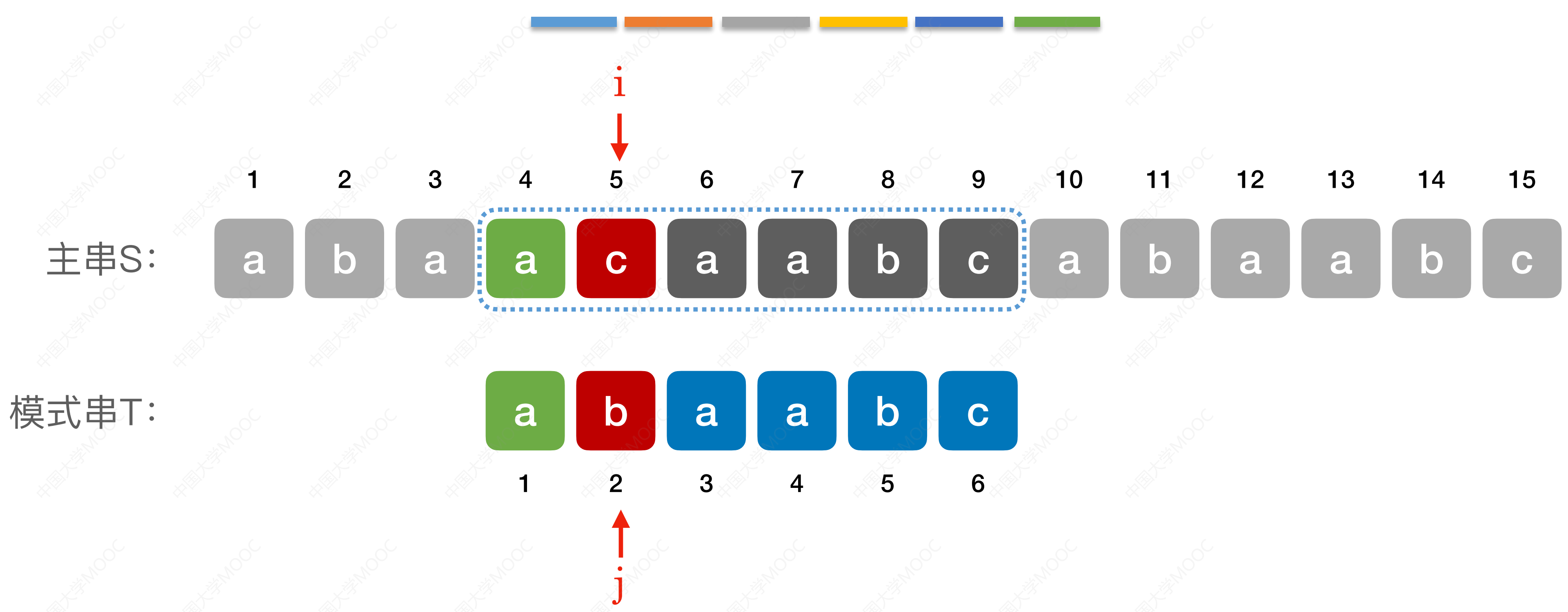
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=2

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=1

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=1

当第1个元素匹配失败时，匹配下一个相邻子串，令 j=0, i++, j++

改造一哈例子



对于模式串 T = 'abaabc'

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=3

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=2

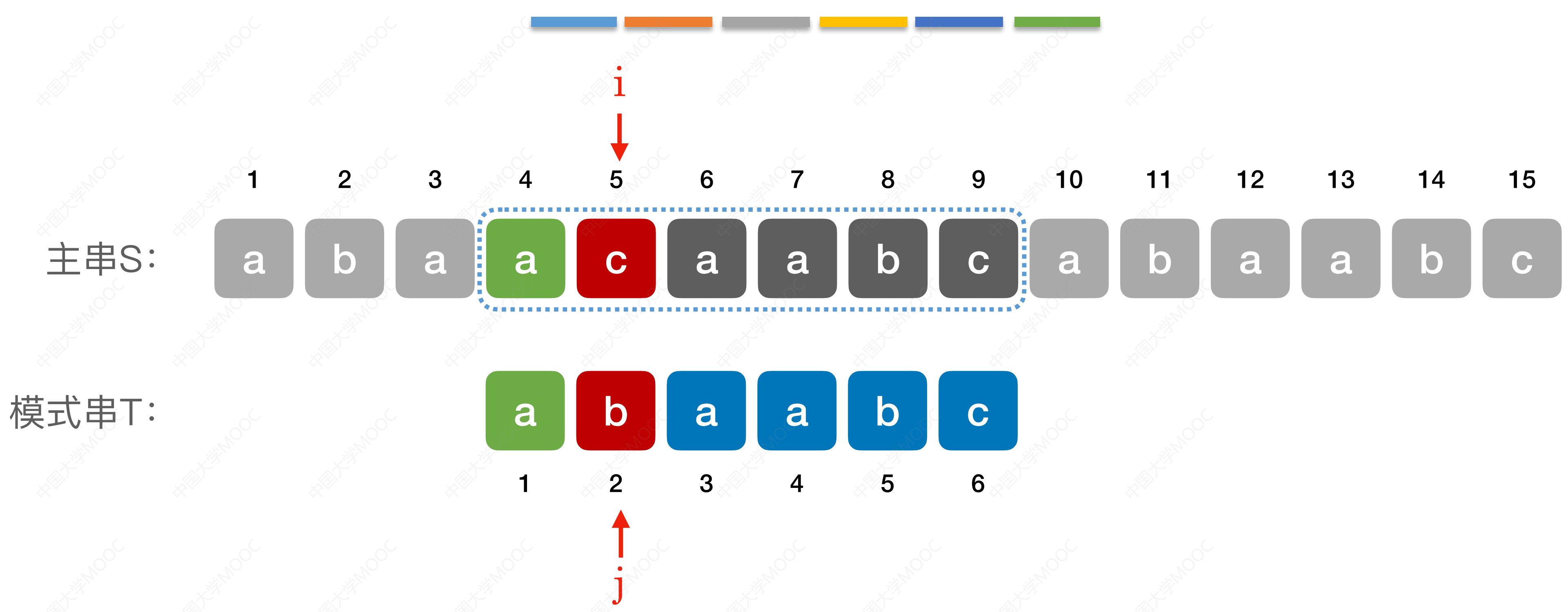
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=2

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=1

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 j=1

当第1个元素匹配失败时，匹配下一个相邻子串，令 j=0, i++, j++

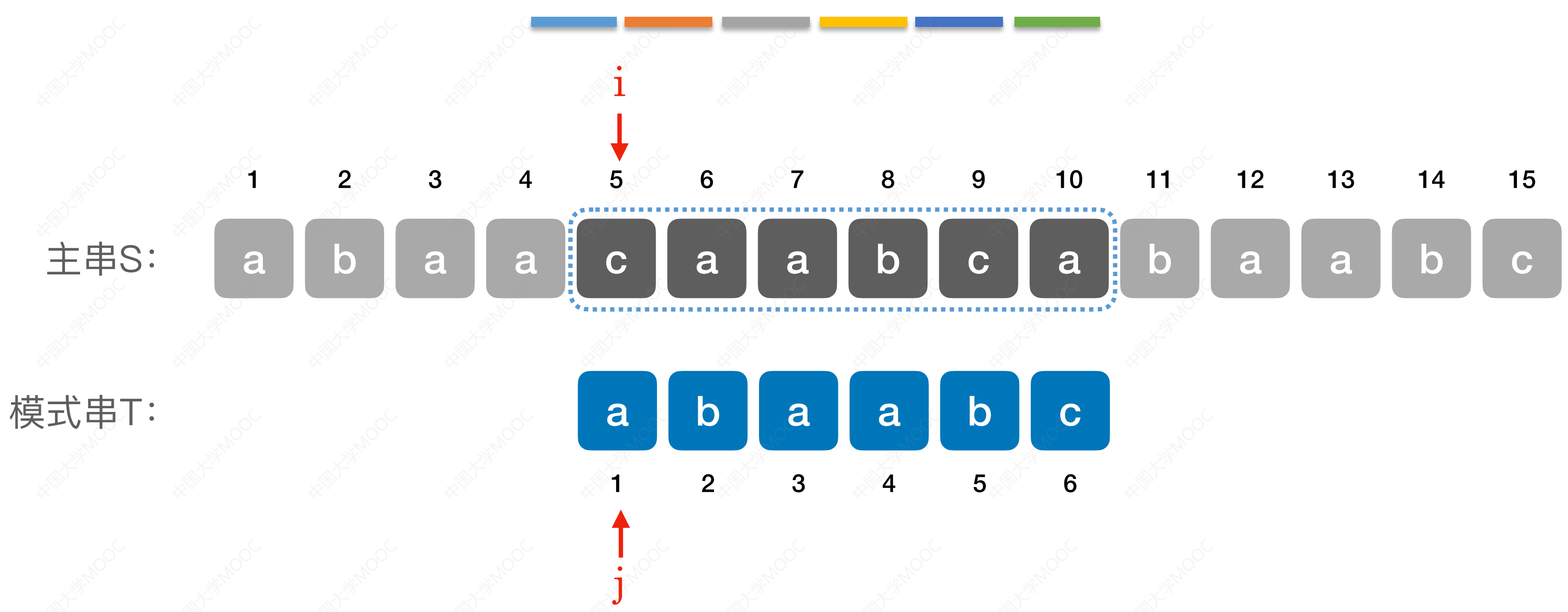
改造一哈例子



对于模式串 $T = 'abaabc'$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

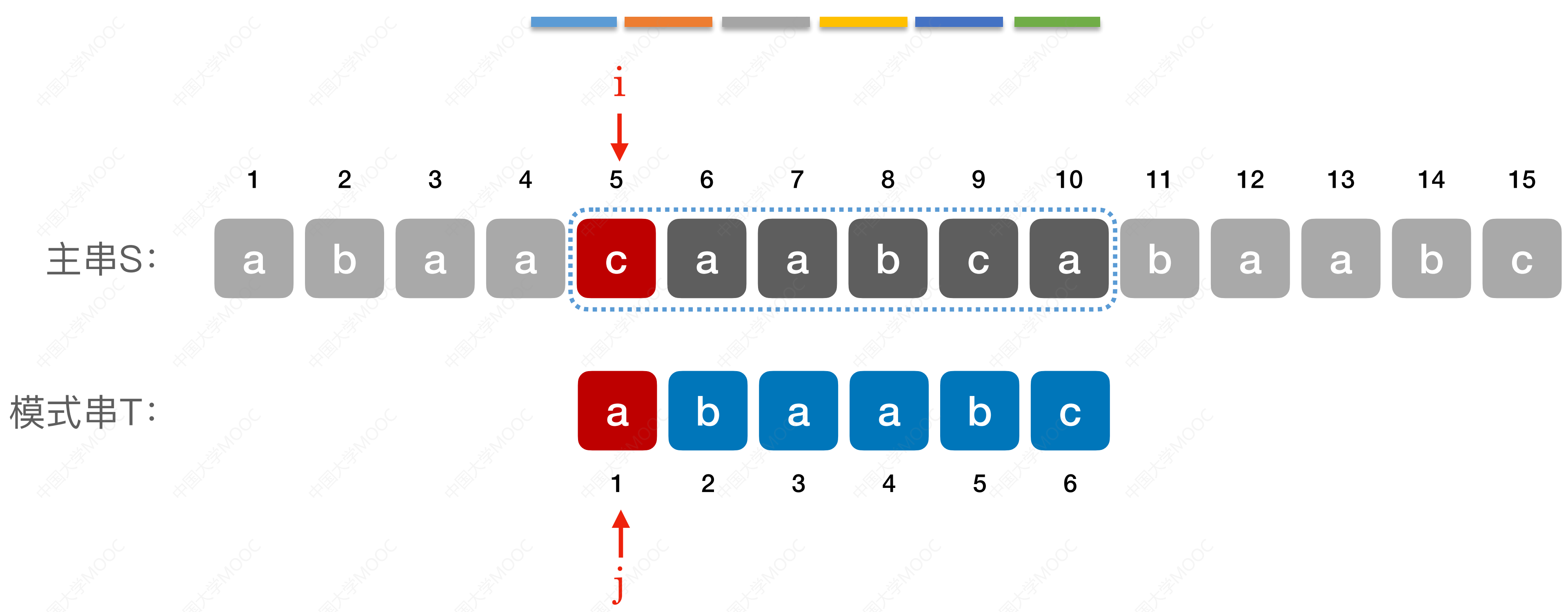
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时, 可令主串指针 **i 不变**, 模式串指针 **j=3**
- 当第5个元素匹配失败时, 可令主串指针 **i 不变**, 模式串指针 **j=2**
- 当第4个元素匹配失败时, 可令主串指针 **i 不变**, 模式串指针 **j=2**
- 当第3个元素匹配失败时, 可令主串指针 **i 不变**, 模式串指针 **j=1**
- 当第2个元素匹配失败时, 可令主串指针 **i 不变**, 模式串指针 **j=1**
- 当第1个元素匹配失败时, 匹配下一个相邻子串, 令 **j=0, i++, j++**

改造一哈例子



对于模式串 $T = 'abaabc'$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

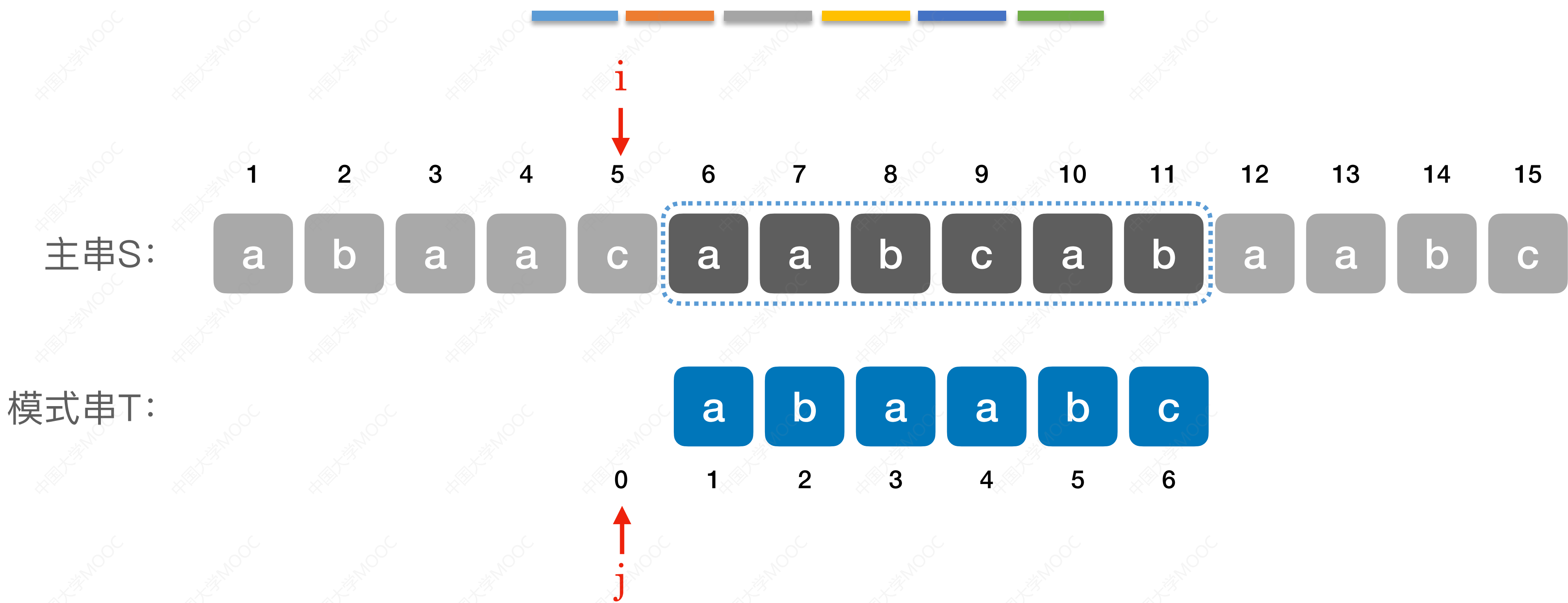
当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

王道考研/CSKAOYAN.COM

改造一哈例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$

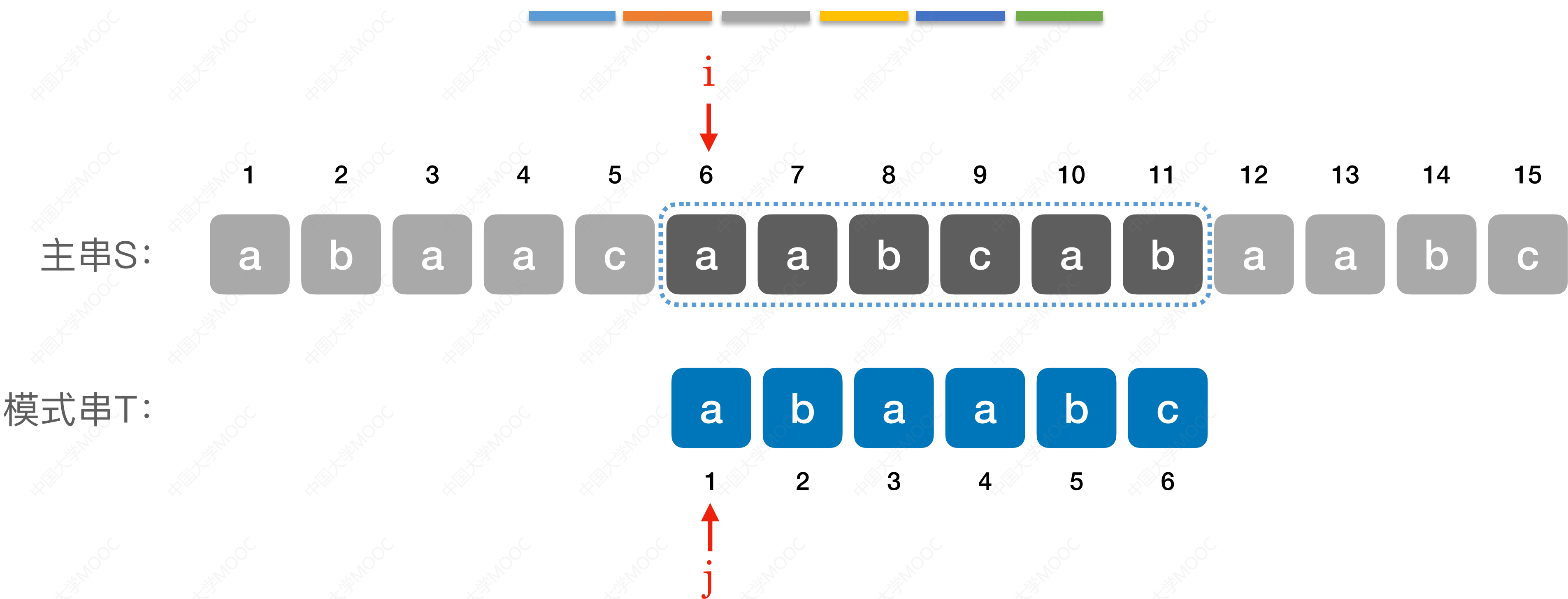
当第4个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

改造一哈例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=3**

当第5个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=2**

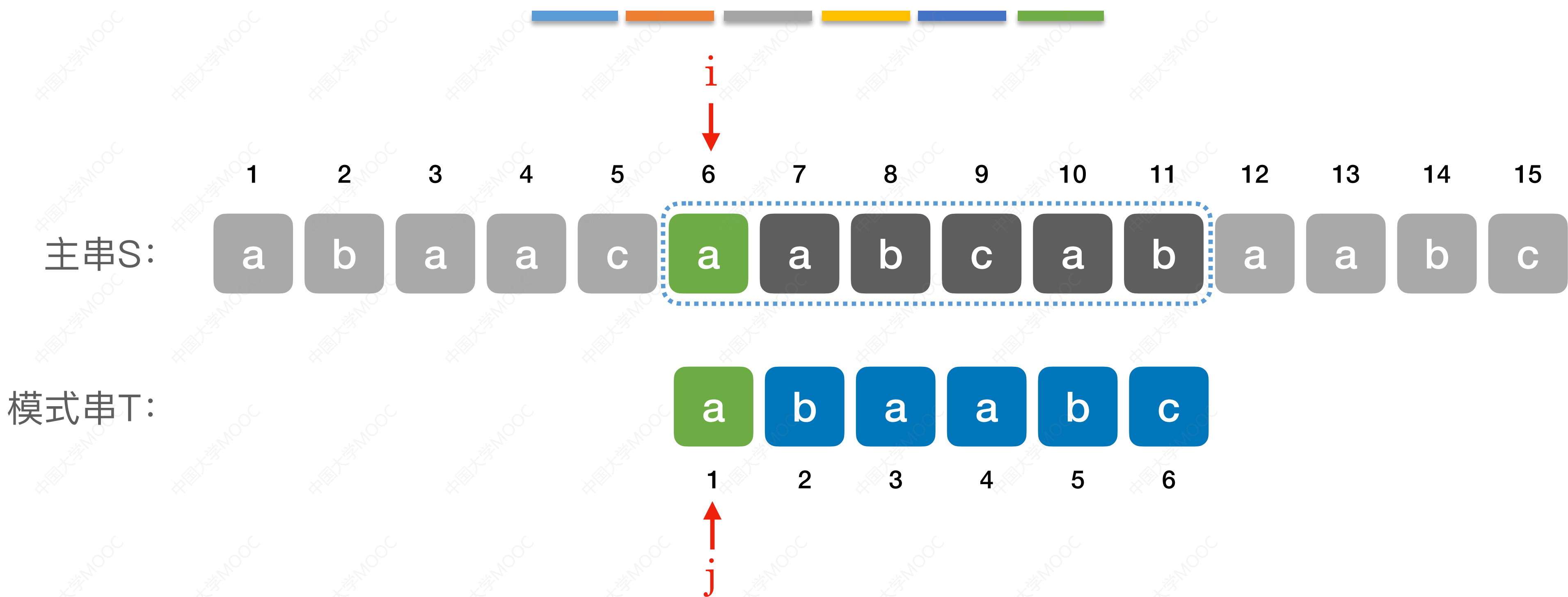
当第4个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=2**

当第3个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=1**

当第2个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=1**

当第1个元素匹配失败时，匹配下一个相邻子串，令 **j=0, i++, j++**

改造一哈例子

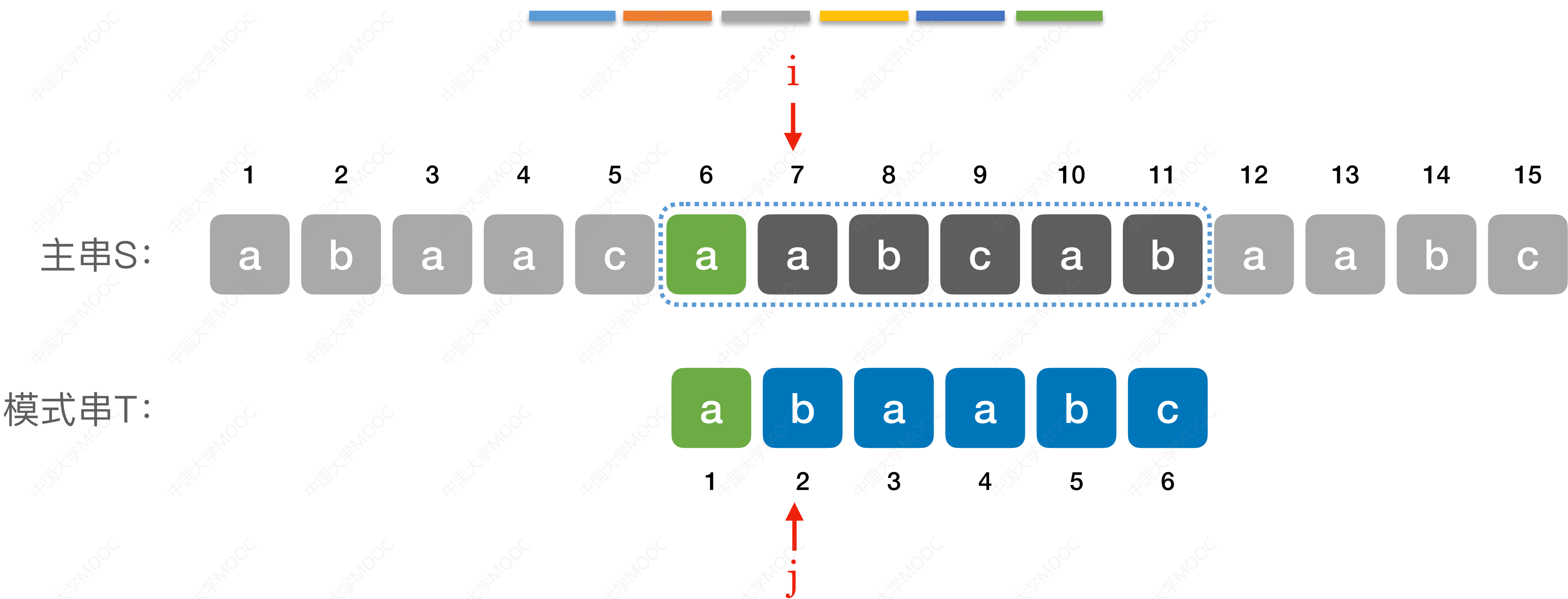


对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

王道考研/CSKAOYAN.COM

改造一哈例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=3**

当第5个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=2**

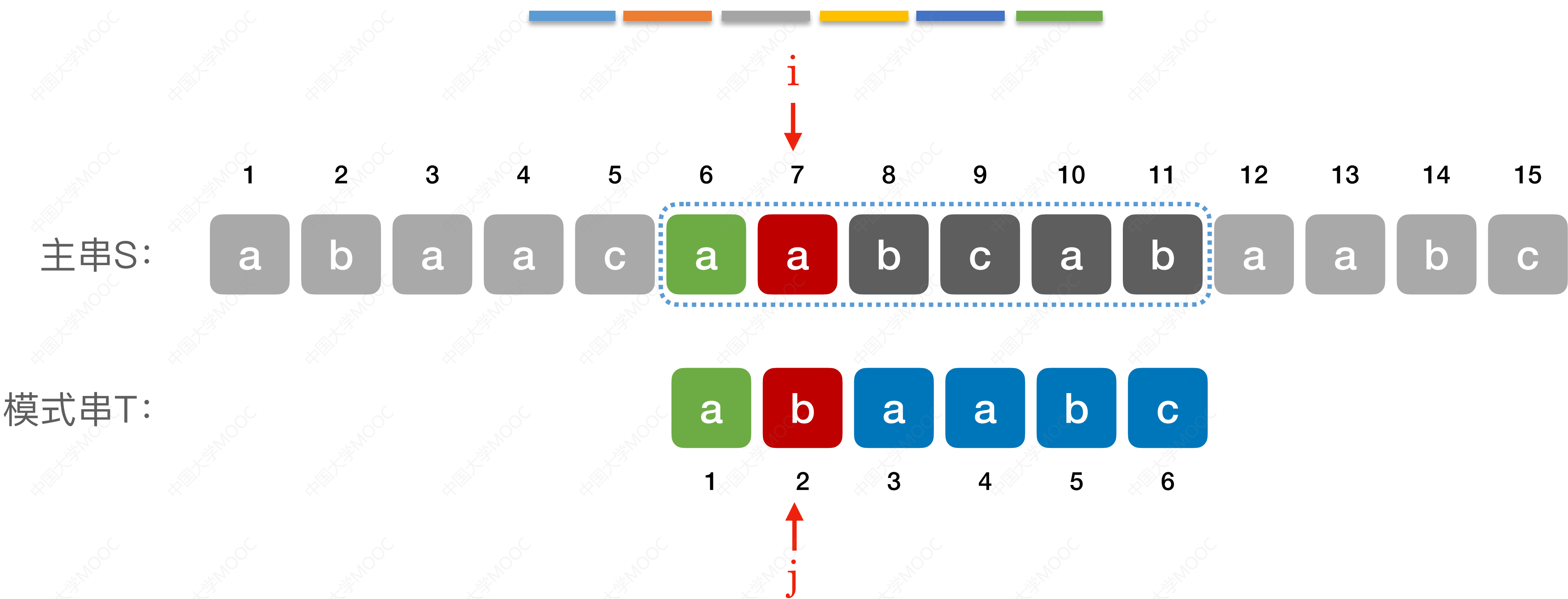
当第4个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=2**

当第3个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=1**

当第2个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=1**

当第1个元素匹配失败时，匹配下一个相邻子串，令 **j=0, i++, j++**

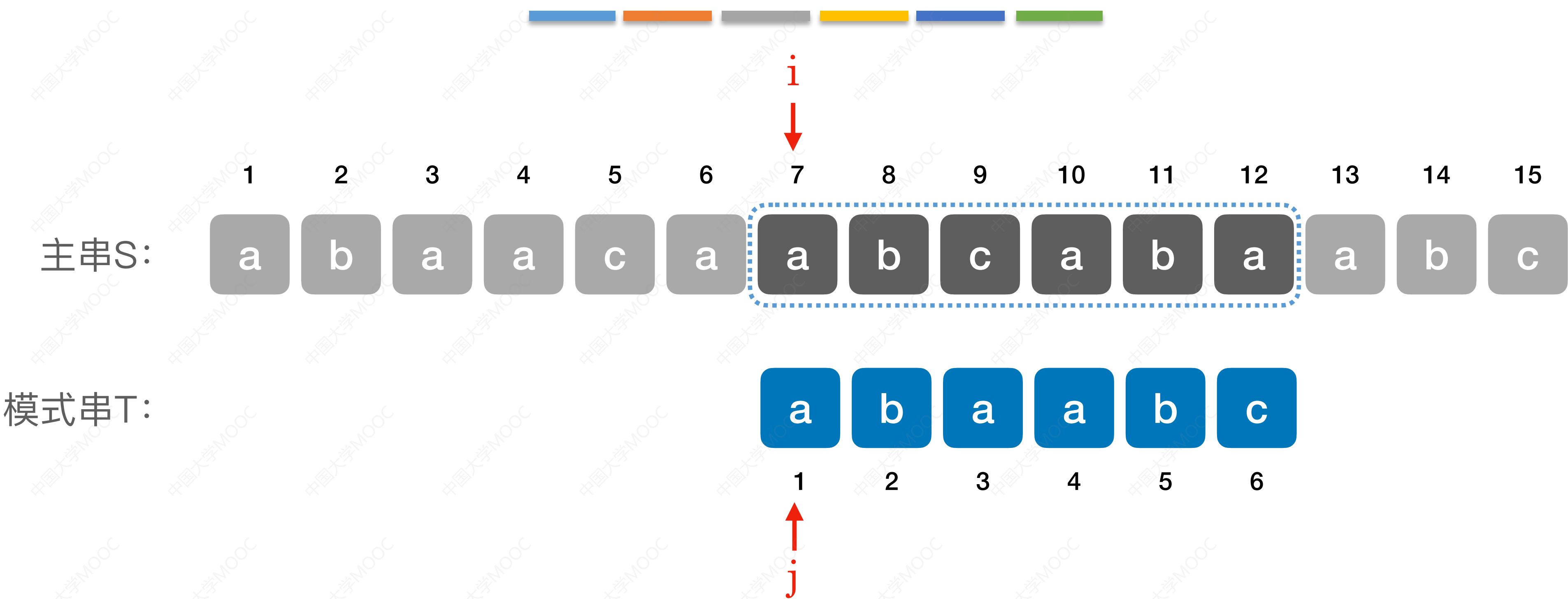
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

改造一哈例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$

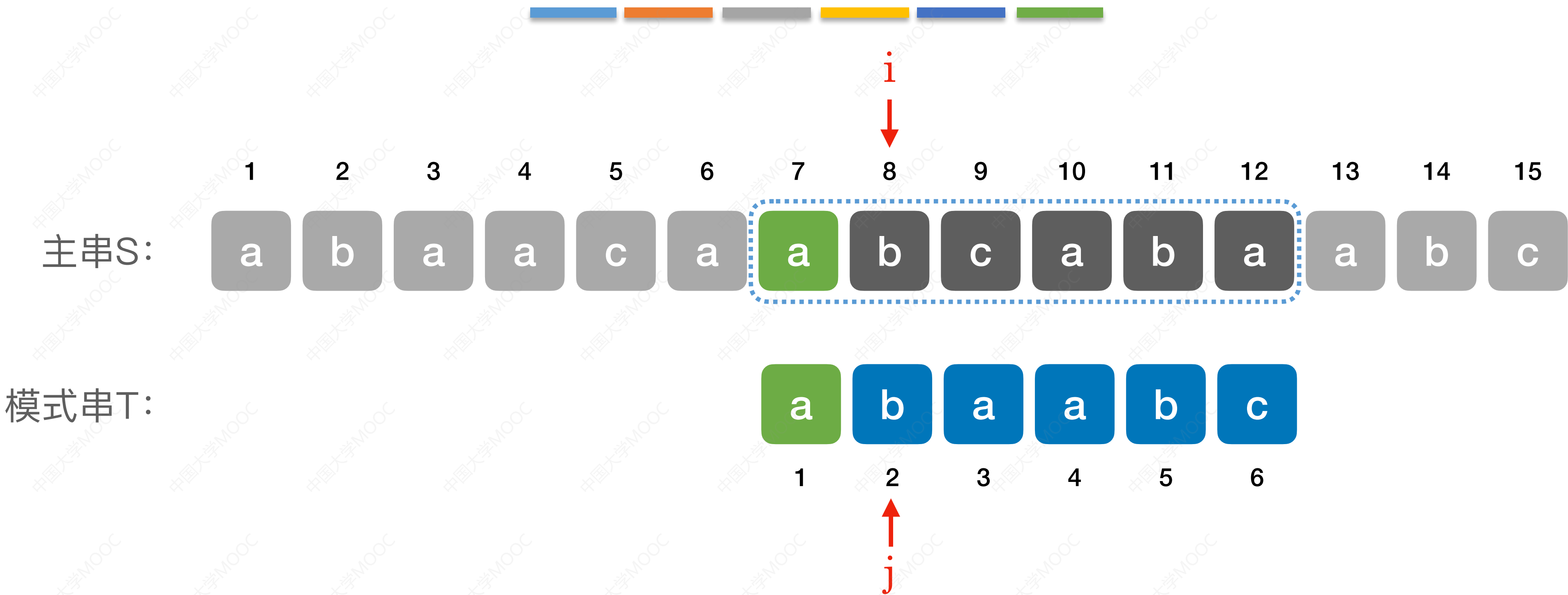
当第4个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

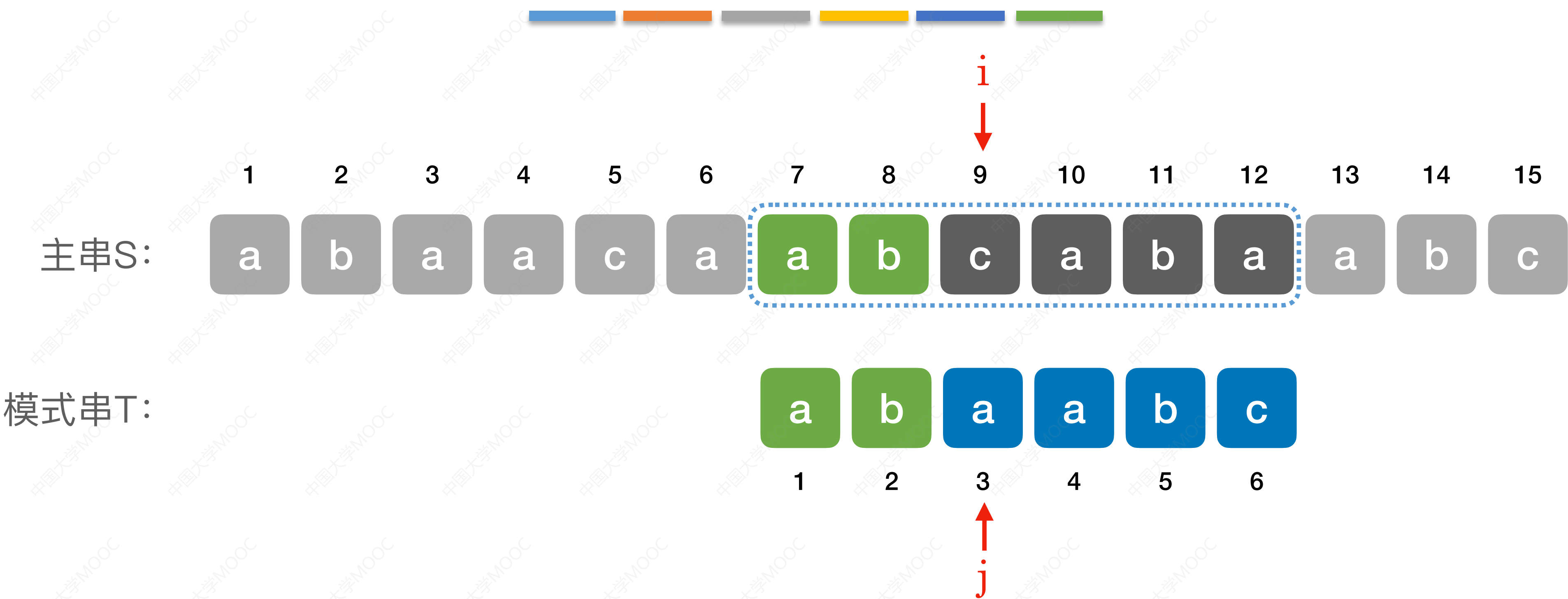
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

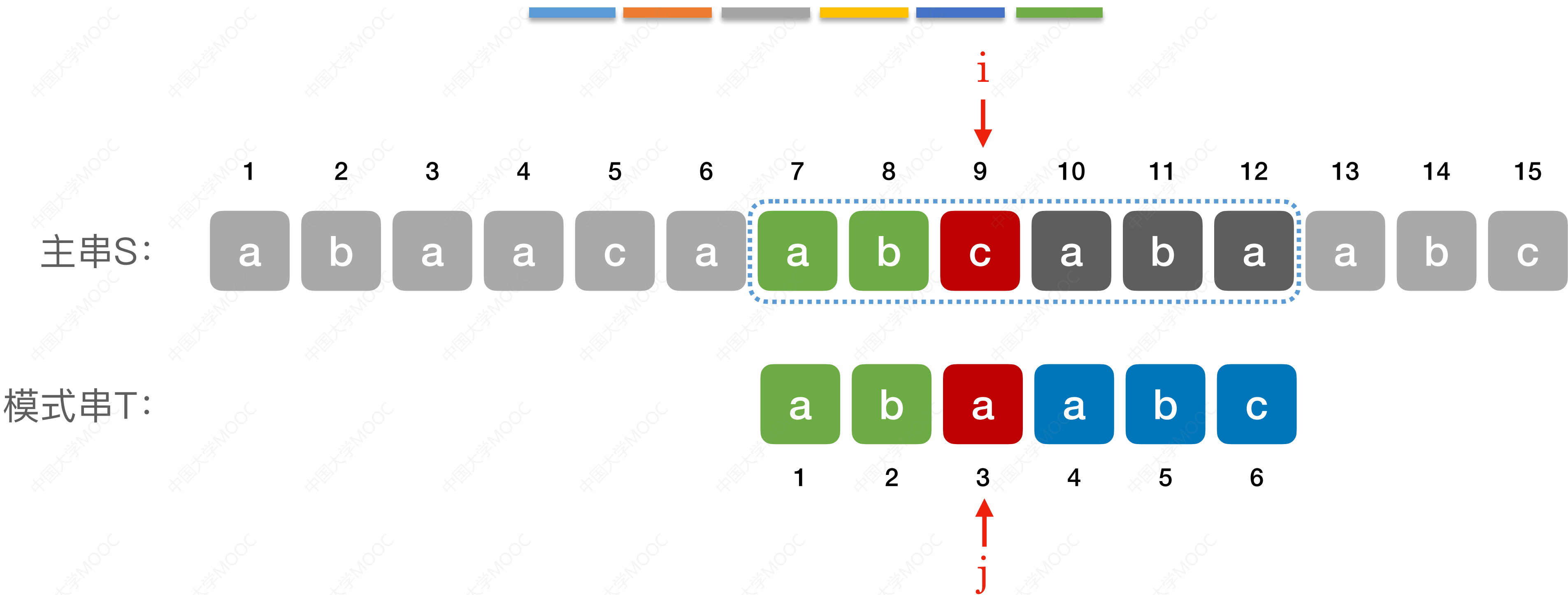
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

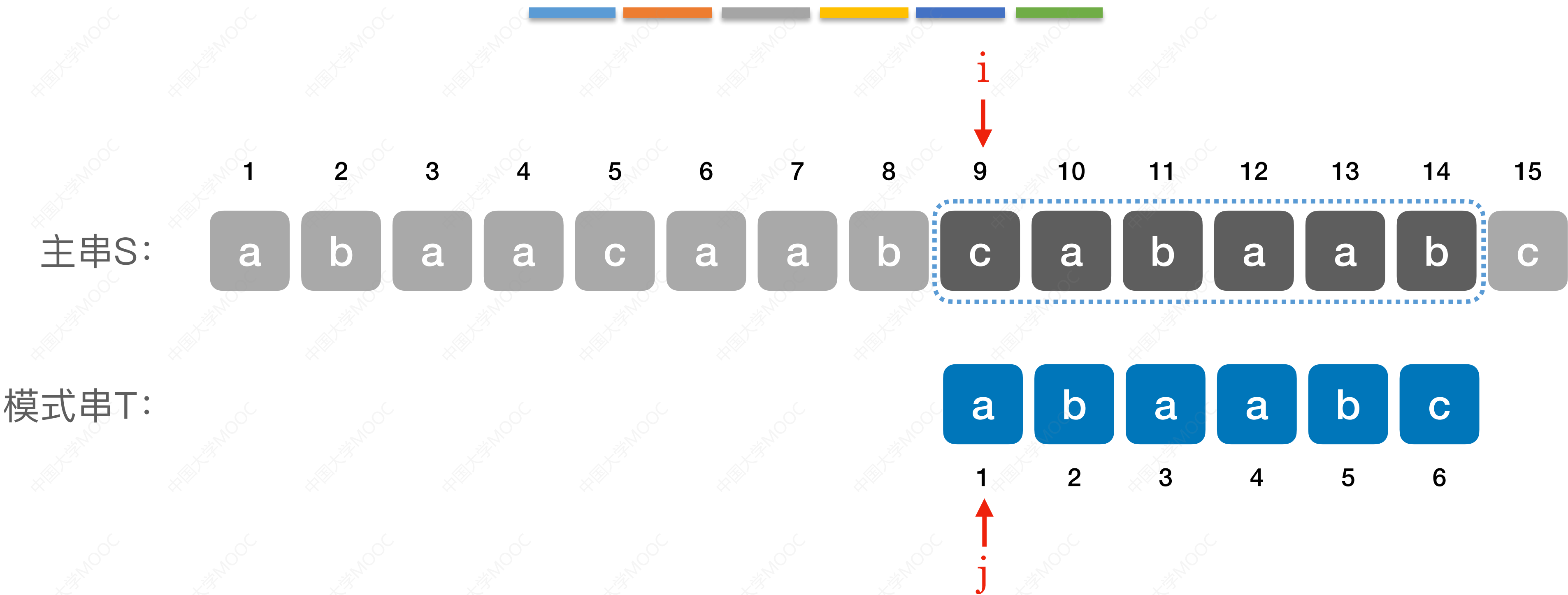
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

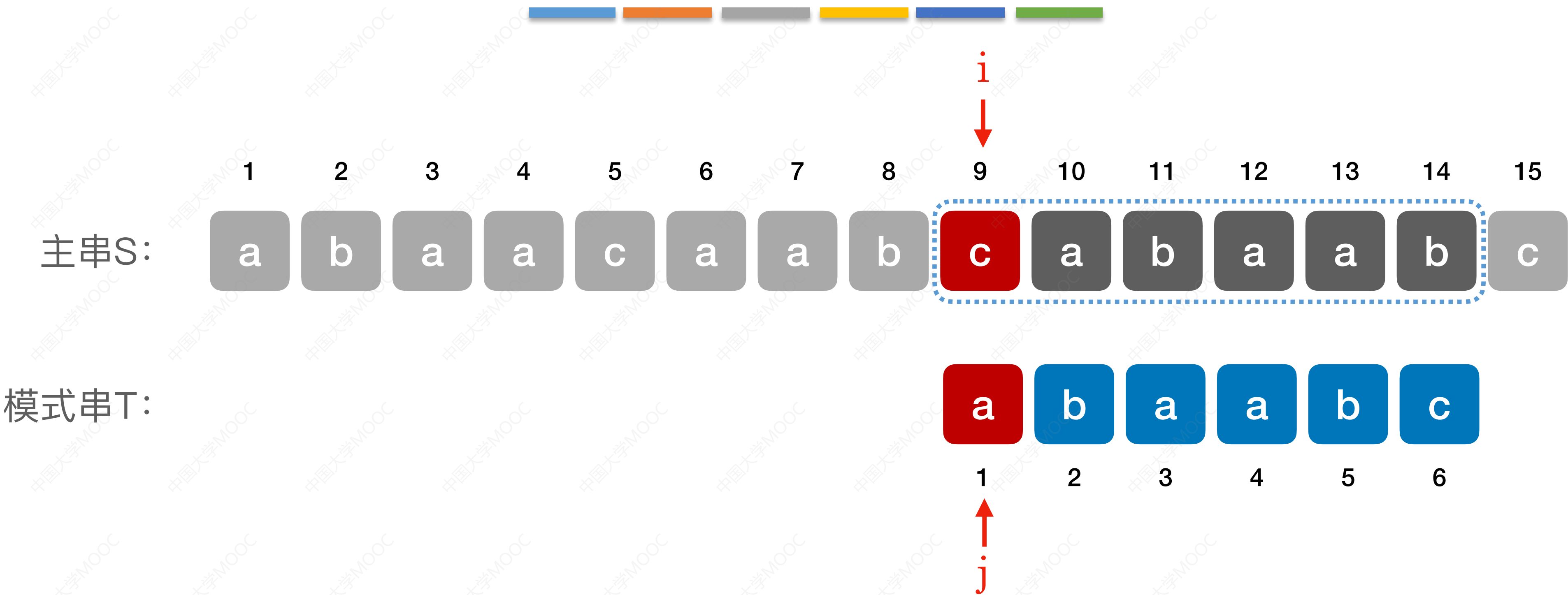
改造一哈例子



对于模式串 $T = \text{'abaabc'}$

- 当第6个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=3$
- 当第5个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$
- 当第4个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=2$
- 当第3个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$
- 当第2个元素匹配失败时，可令主串指针 *i* 不变，模式串指针 $j=1$
- 当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

改造一哈例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

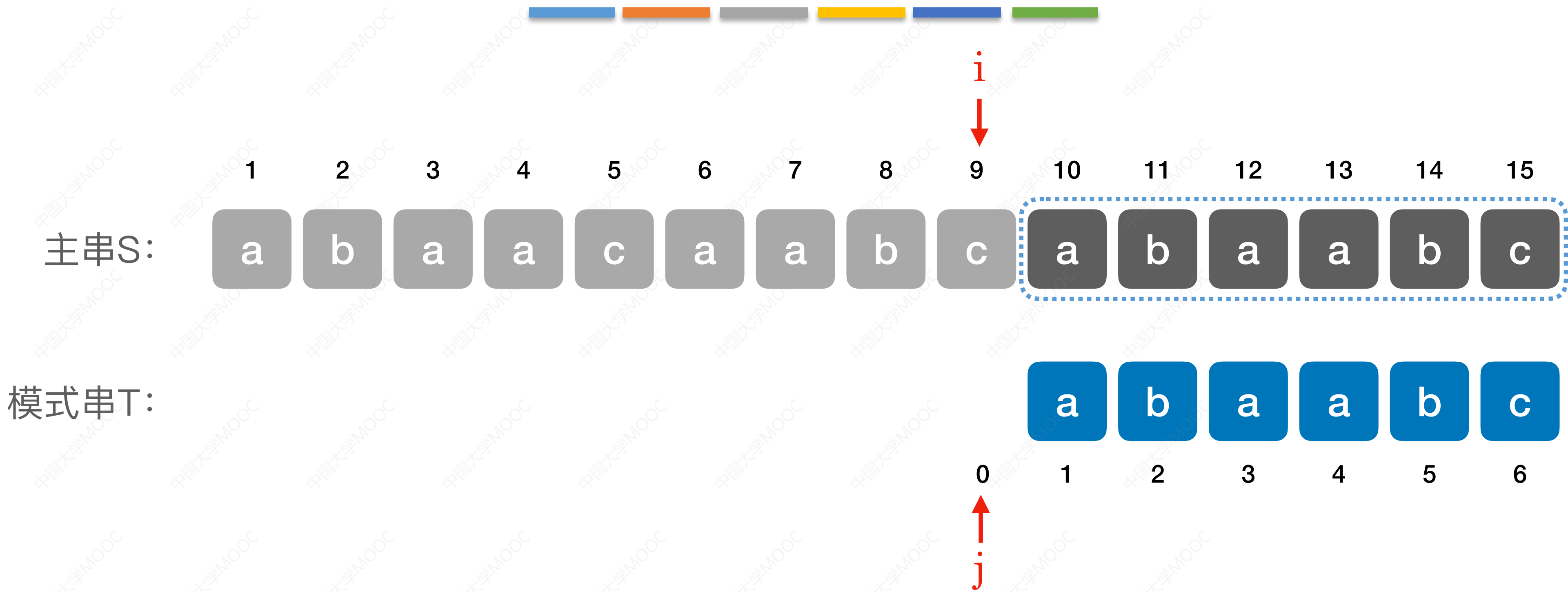
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

改造一哈例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

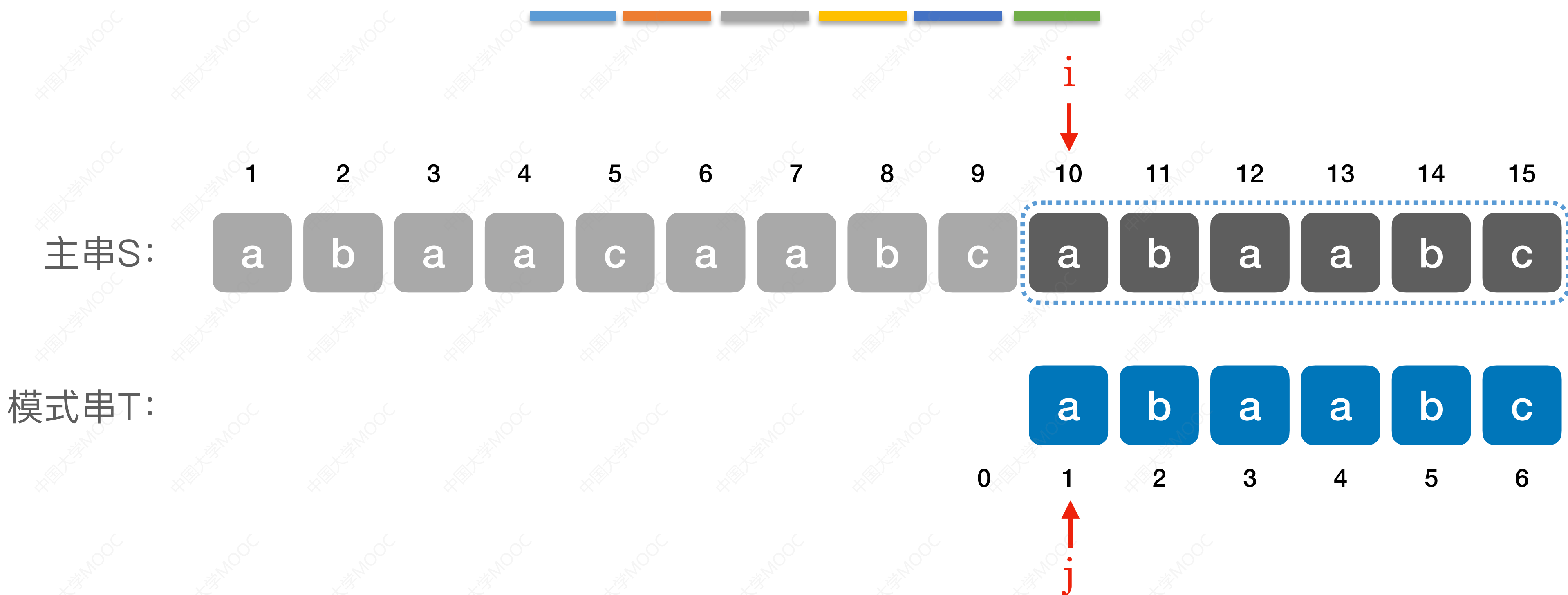
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

改造一哈例子



对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

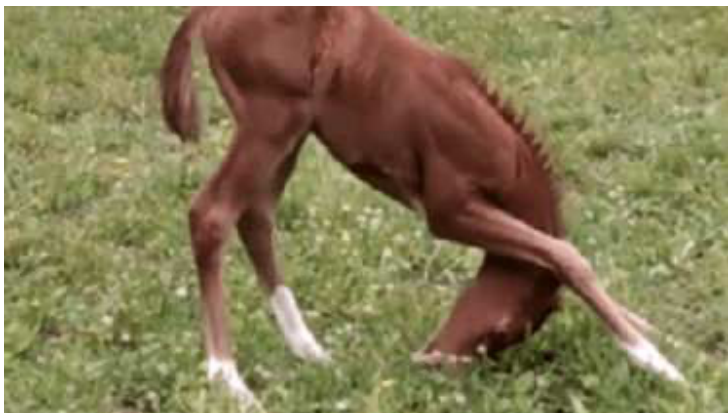
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

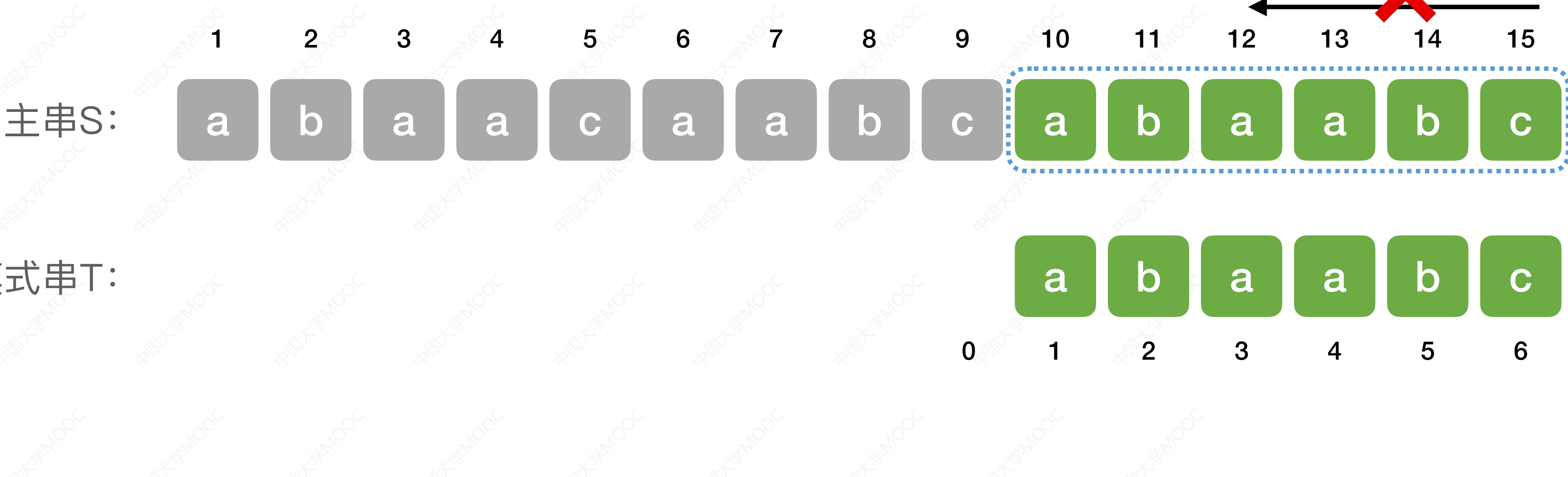
当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

改造一哈例子



好马不吃回头草
优化后主串指针不“回溯”



对于模式串 $T = 'abaabc'$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$

当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

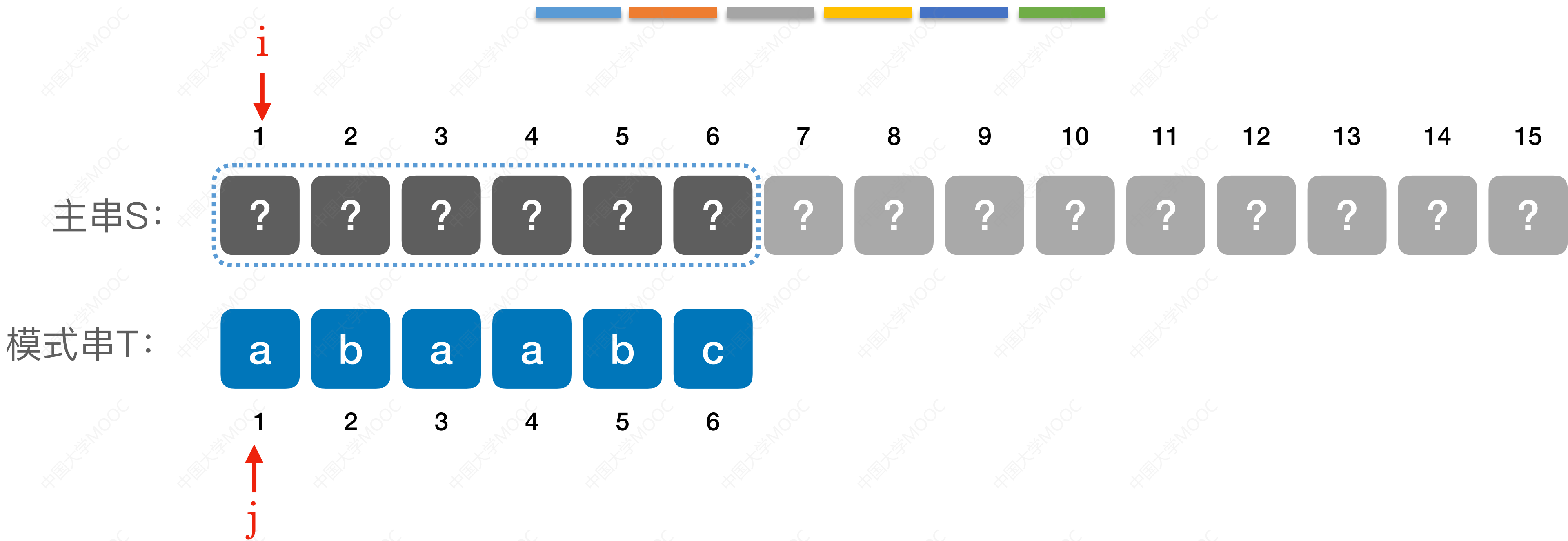
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$

当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$

当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

怎么用代码实现这个处理逻辑？



next[0]	next[1]	next[2]	next[3]	next[4]	next[5]	next[6]
	0	1	1	2	2	3

```
if (S[i] != T[j]) j=next[j];  
if (j==0) { i++; j++ }
```

对于模式串 T = 'abaabc'

当第6个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=3

当第5个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=2

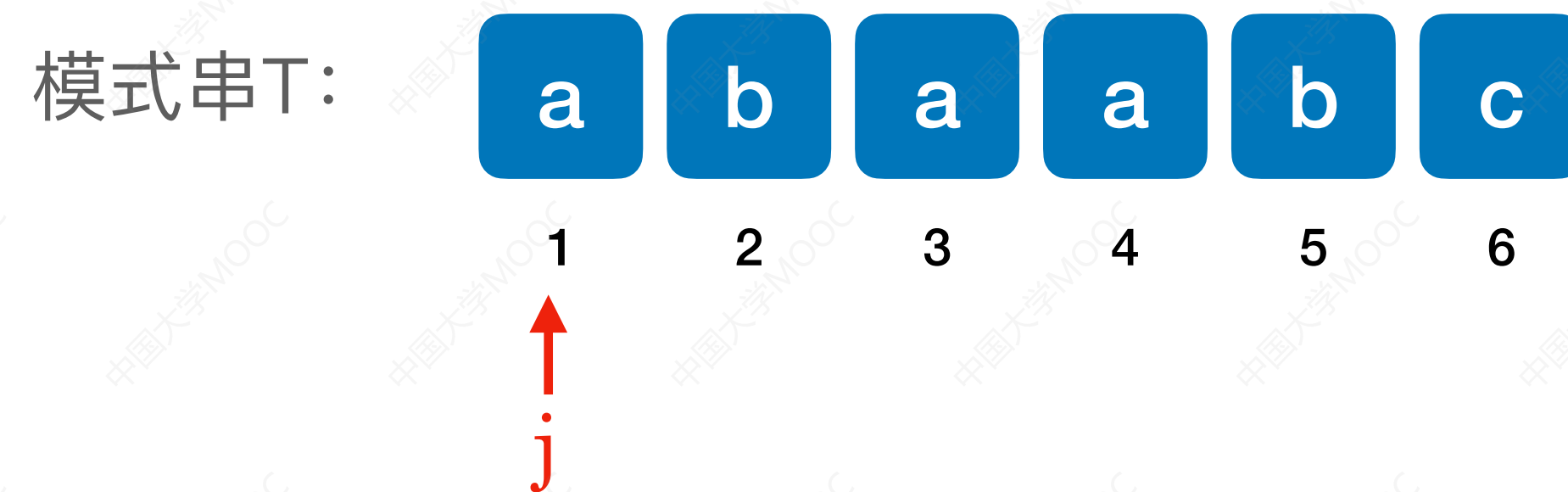
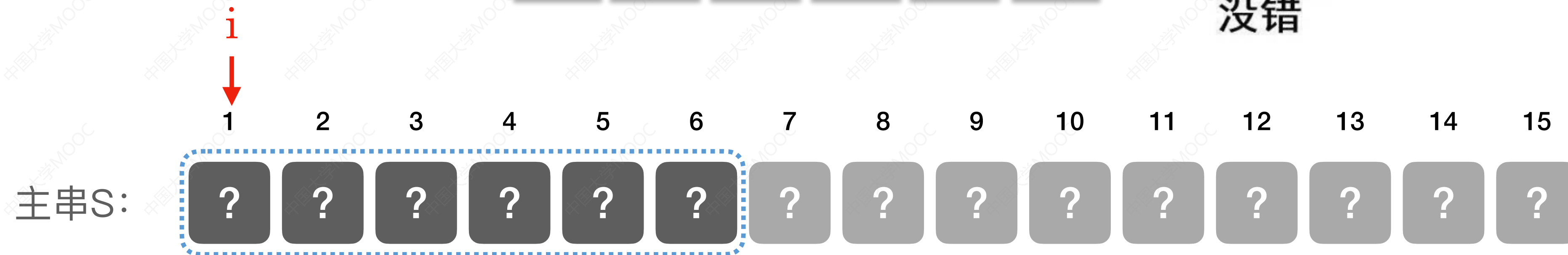
当第4个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=2

当第3个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=1

当第2个元素匹配失败时, 可令主串指针 i 不变, 模式串指针 j=1

当第1个元素匹配失败时, 匹配下一个相邻子串, 令 j=0, i++, j++

KMP算法



next数组:

next[0]	next[1]	next[2]	next[3]	next[4]	next[5]	next[6]
	0	1	1	2	2	3

next数组只和短短的模式串有关，和长长的主串无关

```
if (S[i] != T[j]) j=next[j];  
if (j==0) { i++; j++ }
```

对于模式串 T = 'abaabc'

当第6个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=3**
当第5个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=2**
当第4个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=2**
当第3个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=1**
当第2个元素匹配失败时，可令主串指针 **i 不变**，模式串指针 **j=1**
当第1个元素匹配失败时，匹配下一个相邻子串，令 **j=0, i++, j++**

KMP算法



小老弟往后稍稍

根据模式串T，求出 next 数组

利用next数组进行匹配
(主串指针不回溯)

next数组:

next[0]	next[1]	next[2]	next[3]	next[4]	next[5]	next[6]
	0	1	1	2	2	3

```
if (S[i] != T[j])    j=next[j];  
if (j==0)    { i++; j++; }
```

next数组只和短短的模式串有关，和长长的主串无关

对于模式串 $T = \text{'abaabc'}$

当第6个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=3$
当第5个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
当第4个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=2$
当第3个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
当第2个元素匹配失败时，可令主串指针 i 不变，模式串指针 $j=1$
当第1个元素匹配失败时，匹配下一个相邻子串，令 $j=0, i++, j++$

KMP算法

利用next数组进行匹配
(主串指针不回溯)

next数组:

next[0]	next[1]	next[2]	next[3]	next[4]	next[5]	next[6]
	0	1	1	2	2	3

if (S[i] != T[j]) j=next[j];
if (j==0) { i++; j++ }

```
int Index_KMP(SString S,SString T,int next[]){  
    int i=1, j=1;  
    while(i<=S.length&& j<=T.length){  
        if(j==0 || S.ch[i]==T.ch[j]){  
            ++i;  
            ++j;  
            //继续比较后继字符  
        }  
        else{  
            j=next[j];  
            //模式串向右移动  
        }  
        if(j>T.length)  
            return i-T.length;  
        //匹配成功  
        else  
            return 0;  
    }  
}
```

朴素模式匹配 v.s. KMP算法

```
int Index(SString S, SString T){
    int i=1, j=1;
    while(i<=S.length && j<=T.length){
        if(S.ch[i]==T.ch[j]){
            ++i; ++j; //继续比较后继字符
        }
        else{
            i=i-j+2;
            j=1;
            //指针后退重新开始匹配
        }
    }
    if(j>T.length)
        return i-T.length;
    else
        return 0;
}
```

匹配失败时，主串
指针 i 不回溯

匹配失败时，主串
指针 i 疯狂回溯

```
int Index_KMP(SString S, SString T, int next[]){
    int i=1, j=1;
    while(i<=S.length && j<=T.length){
        if(j==0 || S.ch[i]==T.ch[j]){
            ++i;
            ++j;
            //继续比较后继字符
        }
        else{
            j=next[j];
            //模式串向右移动
        }
    }
    if(j>T.length)
        return i-T.length;
    else
        return 0;
}
```

朴素模式匹配算法，最坏时间复杂度 $O(mn)$

KMP算法，最坏时间复杂度 $O(m+n)$

其中，求 next 数组时间复杂度 $O(m)$
模式匹配过程最坏时间复杂度 $O(n)$

KMP算法

根据模式串T，求出 next 数组

利用next数组进行匹配
(主串指针不回溯)

KMP算法精髓：利用好已经匹配过的模式串的信息

next数组:

next[0]	next[1]	next[2]	next[3]	next[4]	next[5]	next[6]
	0	1	1	2	2	3

if (S[i] != T[j]) j=next[j];

if (j==0) { i++; j++ }

KMP算法，最坏时间复杂度 $O(m+n)$

其中，求 next 数组时间复杂度 $O(m)$
模式匹配过程最坏时间复杂度 $O(n)$

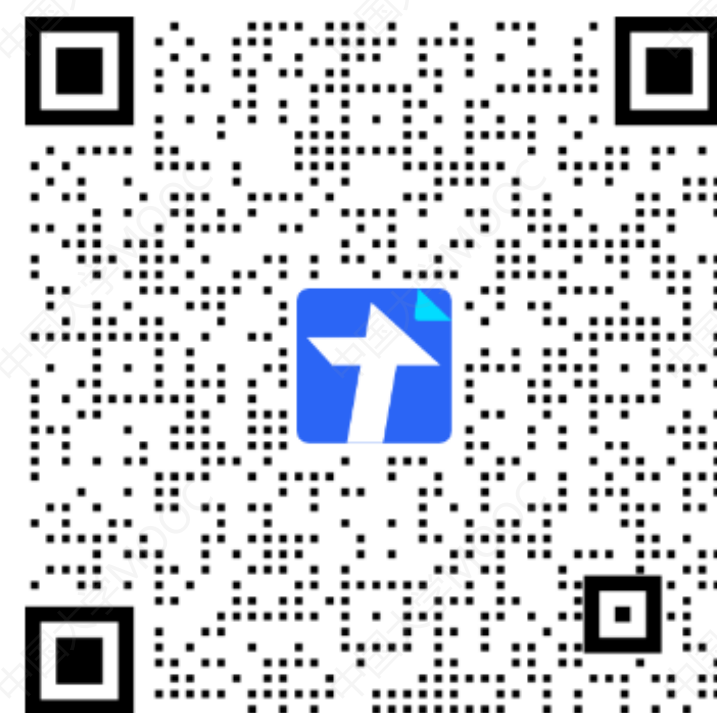
```
int Index_KMP(SString S, SString T, int next[]){
    int i=1, j=1;
    while(i<=S.length&& j<=T.length){
        if(j==0 || S.ch[i]==T.ch[j]){
            ++i;
            ++j;           //继续比较后继字符
        }
        else
            j=next[j];     //模式串向右移动
    }
    if(j>T.length)
        return i-T.length; //匹配成功
    else
        return 0;
}
```

欢迎大家对本节视频进行评价~



学员评分：4.2.2_1 KMP算法

扫一扫二维码打开或分享给好友



— 腾讯文档 —

可多人实时在线编辑，权限安全可控



公众号：王道在线



b站：王道计算机教育



抖音：王道计算机考研