



$m=0$

最大子段和。



$m=1$

枚举两个位置交换，然后最大子段和。





$n=100, m=10$

- 发现其实最后选择的一定是一个子段。
- 枚举最后的答案子段，之后我们还可以使用操作优化答案，不难想到其实就是把我们的子段中最小的 m 个换出去，换外面最大的 m 个。
- 给两个部分排序解决这个问题。
- $O(n^3 \log n)$





$n=300, m=10$

- 这个是非常有意思的做法。
- 我们可以给数组先排序放在b数组里，且记录每个数字的原下标，然后按上个包的做法，不过我们改一下排序的思路。我们在有序的b数组中从头到尾访问一遍，把下标在枚举区间的数字按顺序加入数组，这样可以实现 $O(N)$ 得到区间内外数组的有序序列。
- $O(n^3)$





$n=2000, m=50$ (选学)

- 做法1：发现每次枚举区间加一个数，外部删一个数。有一种东西叫multiset，可以 $\log n$ 维护加数和删数，且可以保证数组的有序性和使用迭代器按顺序枚举数组。
- 复杂度 $O(n^2 \log n)$ ，不过听说会被卡常。
- 做法2：发现其实只要维护 m 个数就行了。内部我们写插排维护一个长度为 m 的数组。其实这个思想也可以做到 n 三方。
- 外部怎么办呢？维护一个有序的链表！删除可以做到 $O(1)$ ，且因为这个链表我们不会动下标，可以通过桶去快速在原序号和排序后点的序号间互相切换。
- 时间复杂度 $O(n^2 m)$