

假设序列为 $a[1:n]$ (index 方便起见写作 $1 \rightarrow n$)，原问题可以转化为求 n 个以 $a[i]$ ($1 \leq i \leq n$) 结尾的 LIS 长度，再取这 n 个结果的最大值。

还设计 $res[n] = \{1\}$ 储存结果，每个结果 ≥ 1 (至少可以取自己)。

$$res[i] = \begin{cases} \max(res[j]) + 1, & \text{若 } j < i, a[j] < a[i] \\ \text{不变}, & \text{其他 } \forall j < i, a[j] \geq a[i] \end{cases}$$

(所有符合条件的 $res[j]$ ($a[j] < a[i]$))

代码如： $\angle res[n] = \{1\}$ // 初始状态

for i in (1, n):

for j in (1, i-1)

if ($a[j] < a[i]$)

$res[i] = \max(res[i], res[j] + 1)$

$ans = \max(ans, res[i])$ // 忘记 ans 初始化了，假设已初 $ans = 0$

return ans

明显双重循环效率为 $O(n^2)$ 。

例 $a = [3, 4, 7, 9, 5, 1, 2, 8]$

$res = [1, 1, 1, 1, 1, 1, 1, 1]$

$[1, 2, 1, 1, 1, 1, 1, 1]$

$[1, 2, 3, 1, 1, 1, 1, 1]$

$[1, 2, 3, 4, 3, 1, 1, 1]$

$[1, 2, 3, 4, 3, 1, 1, 1]$

$[1, 2, 3, 4, 3, 1, 2, 1]$

$[1, 2, 3, 4, 3, 1, 2, 4]$

ans = 4



序序列为 $a[n]$ 可以建一个 $low[n] = \{INF\}$ 。初始化 low 表示空，因为不可能取到
 $low[i]$ 表示 LIS 为 i 的最小结尾元素。 $low[i]$ 越小，后续 LIS 越可能
 长。遍历一遍 $a[n]$ 插入 $low[n]$ 后， $low[n]$ 的长度即为 $\max LIS$ 。

代码如下

(若 $a[i]$ 比 low 中所有元素都大，插入 $a[i]$ ， $len++$
 否则找到第 low 中第 1 个大于 $a[i]$ 的位置用 $a[i]$ 替换
 以维护 low 性质！)

$ans = 1$

for i in $(1, n)$

if ($a[i] > low[ans]$) // 补充定义 $INF < 任何值$

$low[ans++] = a[i]$

else

$low[Search(low, ans-1, a[i])] = a[i]$
 (防越界)

return $ans-1$ (len)

其中

int Search (int* a, int right, int x)

{ $left = 1$

while ($left \leq right$)

$mid = (left + right) / 2$

if ($a[mid] \leq x$)

$left = mid + 1$

else

$right = mid - 1$

return left



插入 n 次, 每次查找操作代价 $O(\log n)$, 总代价 $O(n \log n)$.

例 (同前).

$a = [3, 4, 7, 9, 5, 1, 2, 8]$.

$res = [\infty, \infty, \infty, \infty, \infty, \infty, \infty, \infty]$. 方便起见用 "x" 替 " ∞ ".

↓

$[3, x, x, x, x, x, x, x]$.

↓

$[3, 4, x, x, x, x, x, x]$.

↓

$[3, 4, 7, x, x, x, x, x]$.

↓

$[3, 4, 7, 9, x, x, x, x]$.

↓

$[3, 4, 5, 9, x, x, x, x]$.

↓

$[3, 4, 5, 9, x, x, x, x]$.

↓

$[1, 2, 5, 9, x, x, x, x]$.

↓

$[1, 2, 5, 9, x, x, x, x]$.

$\Rightarrow ans = 4$.

