{{ self.title() }}

题目简述

求从 0 到 e 且分别至少间隔 T 秒经过每个 x_i 两次的最短用时。

子任务设计

$T=1$ 是送分的。$T=3$ 的部分也比较简单，可能有助于选手想到正解。

大部分 NOIP 一等奖选手应该能独立写出本题的 $O\left(n^2\right)$ 的 DP，故前面的 40 分主要是给没想出 DP 的选手爆搜乱搞/状压用的。下文中将不具体写出此类算法。

本题标算做法是线性的，但考虑到要卡掉 $O\left(n\log n\right)$ 做法可能比较麻烦，故直接出到 $n\leq 100000$ 放 $O\left(n\log n\right)$ 过。

算法1：$T=1$

可以证明，当 $T=1$ 时，每次到达一个景点时直接停留 $T=1$ 秒可以取到理论最短用时 $E+nT=n+E$。

故直接输出即可。可以拿到 20 分。

算法2：$T=3$

考虑到 T 很小，我们推测可能这一部分也会有较好的性质。

感性考虑，当 $T=3$ 时我们经过一个景点很久再回来，似乎并没有在这个景点停留 T 秒再前进更优秀。

理性分析一下我们的发现。假设在 t_0 时刻第一次到达 x_i（$i\neq n$），我们有两种方案：

1. 直接前往 x_{i+1} 再返回到 x_i，在 $t_0+\max\left\{2\left(x_{i+1}-x_i\right),T\right\}$ 时刻参观 x_i，并在 $t_0+\left(x_{i+1}-x_i\right)+\max\left\{2\left(x_{i+1}-x_i\right),T\right\}$ 时刻到达 x_{i+1}（此时已经可以参观 x_{i+1}）；
2. 等到 t_0+T 时刻参观 x_i，再在 $t_0+\left(x_{i+1}-x_i\right)+T$ 到达 x_{i+1}，等到 $t_0+\left(x_{i+1}-x_i\right)+2T$ 参观再往前或直接往前。

如果 1. 比 2. 严格更优，则必有 $2\left(x_{i+1}-x_i\right)\leq T$ 或 $\begin{cases}2\left(x_{i+1}-x_i\right)>T\\2\left(x_{i+1}-x_i\right)<2T\end{cases}$，整理得 $x_{i+1}-x_i<T=3$ 即 $x_{i+1}-x_i\leq 2$。

因此如果前进了 3 单位或更多的距离再返回，肯定不如等 T 秒再前进来的优。

如果连续有多个景点距离不超过 2，则需要贪心分段。

总复杂度为 $O\left(n\right)$，可以拿到 20 分。

算法3：$O\left(n^2\right)$ DP

有了上面的算法 2，我们不难发现一个性质：最优解中，n 个景点可以被分为若干段，且每一段都是先从该段坐标最小的一个景点走到最大的一个景点，回到最小的景点并参观，然后走到最大的景点并参观经过的所有景点。换句话说，如果需要返回，则一定要参观之前所有未参观的景点，否则不优。

记 $f\left(i\right)$ 表示参观完 i 以及 i 之前的所有景点，所需的最少时间。由于从 0 到 E 这段路程是必须走的，我们可以从 f 中将这一部分扣去，最后再加上 E。

转移方程为 $f(i)=\min_{0\le j<i}\{f(j)+\max\{2(x_i-x_{j+1}),T\}\}$，其中 $\max\{2(x_i-x_{j+1}),T\}$ 表示两次参观 x_{j+1} 的时间间隔。初始条件 $f(0)=0$。

直接做 $O(n^2)$ DP 就可以拿到 70 分。

算法4：优化 DP

由于转移方程中有 $\min$ 套 $\max$ 的部分，我们考虑对 $\max$ 的两部分拆开来分析。

当 $2(x_i-x_{j+1})\ge T$ 时，我们只需要求 $\min\{f(j)+2(x_i-x_{j+1})\}=2x_i+\min\{f(j)-2x_{j+1}\}$，而满足 $2(x_i-x_{j+1})\ge T$ 的下标一定是一段前缀，故统计前缀最小值即可。

当 $2(x_i-x_{j+1})<T$ 时，我们只需要求 $\min\{f(j)+T\}=T+\min f(j)$。满足条件的 j 是 $[1,i)$ 的一段后缀。可以证明，f 值是单调递增的，故用单调性直接找出分界点即可。总复杂度为 $O(n)$，可以通过本题。

如果没有发现 f 的单调性，也可以用数据结构来维护，复杂度为 $O(n\log n)$，同样可以通过本题。