



综合动态规划-作业讲评

wrpwlp
洛谷网校

课前提示

- 上课的时候专心听讲解，**不要跟着老师抄代码**，下课后独立完成。
- 不使用 AI 做题，AI 会做不等于自己会。
- 不抄袭题解（含对照题解抄一遍），抄对不等于会做。

看完题解后，关闭题解独立练习。

练习中途遇到问题，应当分析题目及自己的思路，而非回忆题解或再次参考题解。

T659611 【第七周】求和

在动态规划的习题中，我们很少操作小数，我们在这个作业题里弥补大家没操作小数的遗憾。

定义 $S_n = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$ 。

可以看到，随着 n 的增加， S_n 会不断变大。

现给出一个整数 k ，要求计算出一个最小的 n ，使得 $S_n > k$ 。

对于 100% 的数据， $1 \leq k \leq 16$ 。

T659611 【第七周】求和

模拟题意，输出答案即可。

T659613 【第七周】 分解

给你一个正整数 n ，你能求出将其分解成若干个素数之和的方案总数吗。

- 对于 100% 的数据， $1 \leq n \leq 10^3$ 。

答案可能超过 `int` 范围。

T659613 【第七周】 分解

首先先使用筛法或者暴力枚举把算出来哪些数是质数。

把数看成重量， n 看成容量，问题转化为完全背包

复杂度为 $O(\text{质数个数} \times n)$

T659615 【第七周】 多跳一次

你要爬上共有 n 阶的楼梯。每次可以爬 1 阶或 2 阶。此外，你最多一次可以使用“特权跳”爬 3 阶。请计算共有多少种不同的爬法。

对于 100% 的数据，满足 $1 \leq n \leq 40$ 。

T659615 【第七周】 多跳一次

如果没有“特权”，这就是斐波那契数列。

考虑到“特权”，设 $f[i][0|1]$ 表示跳到第 i 个位置，目前有/没有使用“特权”，转移即可。

T659618 【第七周】交替

给定仅由小写字母 a 、 b 组成的字符串 s 。你可以把任意位置的字符改成 a 或 b ，每次修改花费 1。

把 s 改成 交替串（相邻字符不同）的最少修改次数是多少？（例如 $abab\dots$ 或 $baba\dots$ 都是交替串。）

对于 100% 的数据， $1 \leq |s| \leq 10^5$

T659618 【第七周】交替

其实是放进来诈骗的，提示大家有简单贪心做法的时候不要硬来dp。

你当然可以设 $f[i][0|1]$ 类似的状态表示考虑前 i 位置的最优解，但是没有必要。

这个序列要么 a 开头，要么 b 开头，综合两种情况取最小值即可。

T659620 【第七周】双倍

给定一个长度为 n 的整数序列 a_i , 要求从中选出两个不相交、且中间至少隔了一个位置的长度至少为 1 的连续子段 (序列中连续且非空的一段), 使得这两个子段中的整数的总和最大, 并输出所求的两个子段中的整数的总和。

对于 100% 的数据, 有 $N \leq 10^6$ 。

数据保证运算过程中的所有数值均在 64 位有符号整型的表示范围内, 换言之, 记得开 `long long` !

T659620 【第七周】双倍

如果只需要选一段，这就是最大子段和问题。

最大子段和的做法就是维护一个 $f[i] = \max(a[i], f[i - 1] + a[i])$

这里有两段怎么处理呢？

T659620 【第七周】双倍

枚举分割点，把序列分成两半，把两半的最大子段和加起来即可。

所以我们需要预处理前缀/后缀的子段和，两者复杂度都是 $O(n)$

P1616 疯狂的采药

有 m 种草药，一共可以用来采药的时间是 t 。采摘一份第 i 种草药需要花费时间 a_i ，同时可以获得价值 b_i 。**每种草药都有无数份。**

请问你最多可以采价值为多少的草药？

$$1 \leq m \leq 10^4, 1 \leq t \leq 10^7, m \times t \leq 10^7, 1 \leq a_i, b_i \leq 10^4$$

P1616 疯狂的采药

考虑认为时间就是背包容量，可以发现这和 01 背包问题很相似，唯一的区别是，这里每个物品有无数个。而在 01 背包中，每个物品只有一个。

套用完全背包即可。

P1757 通天之分组背包

自 01 背包问世之后，小 A 对此深感兴趣。一天，小 A 去远游，却发现他的背包不同于 01 背包，他的物品大致可分为 k 组，每组中的物品相互冲突，现在，他想知道最大的利用价值是多少。

$0 \leq m \leq 1000, 1 \leq n \leq 1000, 1 \leq k \leq 100, a_i, b_i, c_i$ 在 int 范围内。

P1757 通天之分组背包

这是一个分组背包，看起来我们没学过，但是事实上，分组背包和 01 背包区别并不大。

转移 01 背包的时候，我们需要确保状态从“没有选第 i 个物品”转移过来，分组的时候，我们就确保状态从“没有选第 i 组物品”这个状态转移过来。

设 $f[i][j]$ 表示考虑前 i 组物品，总重量为 j 的最大价值，转移即可。

P7774 [COCI 2009/2010 #2] KUTEVI

给定 N 个角（第 i 个角记作 a_i ），作为初始角，另给定 M 个角（第 i 个角记作 b_i ），作为目标角。

请求出对于每个 b_i ，它是否能被若干个 a_i 之间的加、减运算得到。

注意同一个 a_i 可以用多次，也可以不用。

$$1 \leq N, M \leq 10, 0 < a_i, b_i < 360.$$

P7774 [COCI 2009/2010 #2] KUTEVI

其实题意就是有 n 个数，这 n 个数你可以自由加或者减，运算过程对 360 取模，问能凑出来给定的 m 个数中的多少个。一个数可以用多次。

P7774 [COCI 2009/2010 #2] KUTEVI

这事实上是一个普通的背包问题，只是每个数可以加也可以减。

直接背包即可。

注意：这里题解的滚动数组写法直接理解是有问题的，因为往大/小都有转移，直接滚动数组不对。

P7774 [COCI 2009/2010 #2] KUTEVI

这事实上是一个普通的背包问题，只是每个数可以加也可以减。

直接背包即可。

注意：这里题解的滚动数组写法直接理解是有问题的，因为往大/小都有转移，直接滚动数组不对。

P1521 求逆序对

求长度为 N , 逆序对个数恰好为 K 的排列个数。

$$N \leq 1000, K \leq N(N - 1)/2$$

P1521 求逆序对

设 $f[i][j]$ 表示由 1 到 i 组成的所有排列中，逆序对个数为 j 的排列个数。

考虑把数字 i 插入到之前的排列中。将数字 i 插入到序列中会产生一些逆序对。

$$\begin{aligned} f[i][j] &= \sum_{k=0}^{i-1} f[i-1][j-(i-1-k)] \\ &= \sum_{k=j-i+1}^j f[i-1][k] \end{aligned}$$

P1521 求逆序对

考虑前缀和优化。

$$\begin{aligned} f[i][j] &= \sum_{k=0}^{i-1} f[i-1][j-(i-1-k)] \\ &= \sum_{k=j-i+1}^j f[i-1][k] \end{aligned}$$

我们可以在从 $i-1$ 递推到 i 的时候处理 $f[i-1]$ 的前缀和。这样就可以简单做到平方复杂度了。

P2066 机器分配

总公司拥有高效设备 M 台，准备分给下属的 N 个分公司。各分公司若获得这些设备，可以为国家提供一定的盈利。问：如何分配这 M 台设备才能使国家得到的盈利最大？求出最大盈利值。其中 $M \leq 15$, $N \leq 10$ 。分配原则：每个公司有权获得任意数目的设备，但总台数不超过设备数 M 。

输入格式

第一行有两个数，第一个数是分公司数 N ，第二个数是设备台数 M 。

接下来是一个 $N \times M$ 的矩阵，表明了第 i 个公司分配 j 台机器的盈利。

最大盈利值相同时，要求编号小的公司分得设备尽可能少。

输出格式

第一行为最大盈利值。

接下来 N 行为第 i 分公司分 x 台。

P2066 机器分配

普通的背包问题，但是要求输出中间过程。

还是设 $f[i][j]$ 表示考虑前 i 个公司，已经用 j 个设备。

但是要求输出方案，并且编号小的公司分得少，怎么办？

P2066 机器分配

```
for (int i = n, j = m; i; i--) {
    for (int k = 0; k <= j; k++) {
        if (f[i][j] == f[i - 1][j - k] + a[i][k]) {
            way[i] = k;
        }
    }
    j -= way[i];
}
for (int i = 1; i <= n; i++)
    cout << i << ' ' << way[i] << '\n';
```

P2904 [USACO08MAR] River Crossing S

农夫约翰以及他的 $N(1 \leq N \leq 2500)$ 头奶牛打算过一条河，但他们所有的渡河工具，仅仅是一个木筏。

由于奶牛不会划船，在整个渡河过程中，约翰必须始终在木筏上。在这个基础上，木筏上的奶牛数目每增加 1，FJ 把木筏划到对岸就得花更多的时间。

当约翰一个人坐在木筏上，他把木筏划到对岸需要 $M(1 \leq M \leq 1000)$ 分钟。当木筏搭载的奶牛数目从 $i - 1$ 增加到 i 时，约翰得多花 $M_i(1 \leq M_i \leq 1000)$ 分钟才能把木筏划过河（也就是说，船上有 1 头奶牛时，约翰得花 $M + M_1$ 分钟渡河；船上有 2 头奶牛时，时间就变成 $M + M_1 + M_2$ 分钟。后面的以此类推）。那么，约翰最少要花多少时间，才能把所有奶牛带到对岸呢？当然，这个时间得包括约翰一个人把木筏从对岸划回来接下一批的奶牛的时间。

P2904 [USACO08MAR] River Crossing S

普通的完全背包问题。

预处理一下一次运 i 头牛的代价，简单转移即可。

P1156 垃圾陷阱

卡门——农夫约翰极其珍视的一头 Holsteins 奶牛——已经落到了“垃圾井”中。“垃圾井”是农夫们扔垃圾的地方，它的深度为 D ($2 \leq D \leq 100$) 英尺。

卡门想把垃圾堆起来，等到堆得与井深同样高或比井深更高（即，垃圾高度总和 $\geq D$ ）时，她就能逃出井外了。另外，卡门可以通过吃一些垃圾来维持自己的生命。

每个垃圾都可以用来吃或堆放，并且堆放垃圾不用花费卡门的时间。

假设卡门预先知道了每个垃圾扔下的时间 t ($1 \leq t \leq 1000$)，以及每个垃圾堆放的高度 h ($1 \leq h \leq 25$) 和吃进该垃圾能增加维持生命的时间 f ($1 \leq f \leq 30$)，要求出卡门最早能逃出井外的时间，假设卡门当前体内有足够的持续 10 小时的能量，如果卡门 10 小时内（不含 10 小时，维持生命的时间同）没有进食，卡门就将饿死。特别地，若体力值为 0 时吃下垃圾或逃出井外也不会饿死。

第一行为两个整数， D 和 G ($1 \leq G \leq 100$)， G 为被投入井的垃圾的数量。

第二到第 $G + 1$ 行每行包括三个整数： T ($1 \leq T \leq 1000$)，表示垃圾被投进井中的时间； F ($1 \leq F \leq 30$)，表示该垃圾能维持卡门生命的时间；和 H ($1 \leq H \leq 25$)，该垃圾能垫高的高度。

P1156 垃圾陷阱

可以看到，垃圾要么被堆起来，要么被吃掉。我们按时间顺序来考虑每个物品。

考虑把高度看成容量，这就是一个变体的背包问题。

具体的，考虑设 $f[i][j]$ 表示考虑前 i 个垃圾，堆的高度是 j 当前生命值最大是多少。

转移考虑每个垃圾是被吃掉还是堆起来即可。

复杂度 $O(GD)$

P1063 [NOIP 2006 提高组] 能量项链

在 Mars 星球上，每个 Mars 人都随身佩带着一串能量项链。在项链上有 N 颗能量珠。能量珠是一颗有头标记与尾标记的珠子，这些标记对应着某个正整数。并且，对于相邻的两颗珠子，前一颗珠子的尾标记一定等于后一颗珠子的头标记。因为只有这样，通过吸盘（吸盘是 Mars 人吸收能量的一种器官）的作用，这两颗珠子才能聚合成一颗珠子，同时释放出可以被吸盘吸收的能量。如果前一颗能量珠的头标记为 m ，尾标记为 r ，后一颗能量珠的头标记为 r ，尾标记为 n ，则聚合后释放的能量为 $m \times r \times n$ (Mars 单位)，新产生的珠子的头标记为 m ，尾标记为 n 。

需要时，Mars 人就用吸盘夹住相邻的两颗珠子，通过聚合得到能量，直到项链上只剩下一颗珠子为止。显然，不同的聚合顺序得到的总能量是不同的，请你设计一个聚合顺序，使一串项链释放出的总能量最大。

第一行是一个正整数 N ($4 \leq N \leq 100$)，表示项链上珠子的个数。第二行是 N 个用空格隔开的正整数，所有的数均不超过 1000。第 i 个数为第 i 颗珠子的头标记 ($1 \leq i \leq N$)，当 $i < N$ 时，第 i 颗珠子的尾标记应该等于第 $i + 1$ 颗珠子的头标记。第 N 颗珠子的尾标记应该等于第 1 颗珠子的头标记。

至于珠子的顺序，你可以这样确定：将项链放到桌面上，不要出现交叉，随意指定第一颗珠子，然后按顺时针方向确定其他珠子的顺序。

[NOIP 2006 提高组] 能量项链

首先这是一个环形问题，先考虑序列的情况。

设 $dp[l][r]$ 表示这个区间合并以后得到的最大能量。

$$dp[l][r] = \max(dp[l][k] + dp[k+1][r] + a[l] \times a[k+1] \times a[r+1])$$

[NOIP 2006 提高组] 能量项链

考虑到这是环形问题，运用断环为链技巧，倍长数组。

复杂度 $O(n^3)$