讲 一 讲

达内童程

讲 一 讲

动态规划-数的划分

（NOIp2001）
将整数n分成k份，且每份不能为空。例如：n=7，k=3，下面三种分法被认为是相同的。

1，1，5; 1，5，1; 5，1，1;

问有多少种不同的分法。

【输入】

n，k (6<n<=200，2<=k<=6)

【输出】

一个整数，即不同的分法。

【样例】

输入： 7 3

输出： 4

讲 一 讲

动态规划-数的划分

有几个枚举量我们就设几个状态，
这道题目明显n和k都是枚举量，要求的几种方案是值，

因此建立起递推状态：f[i,j]表示将i分成j份的方案数。

状态转移方程

f[i,j]就可以由f[x,y]等其他状态得来，
最后我们要求得f[n,k]。

讲一讲

动态规划-传纸条（noip2008）

小渊和小轩是好朋友也是同班同学，他们在一起总有谈不完的话题。一次素质拓展活动中，班上同学安排做成一个 m 行 n 列的矩阵，而小渊和小轩被安排在矩阵对角线的两端，因此，他们就无法直接交谈了。幸运的是，他们可以通过传纸条来进行交流。纸条要经由许多同学传到对方手里，小渊坐在矩阵的左上角，坐标(1,1)，小轩坐在矩阵的右下角，坐标 (m,n)。从小渊传到小轩的纸条只可以向下或者向右传递，从小轩传给小渊的纸条只可以向上或者向左传递。在活动进行中，小渊希望给小轩传递一张纸条，同时希望小轩给他回复。班里每个同学都可以帮他们传递，但只会帮他们一次，也就是说如果此人在小渊递给小轩纸条的时候帮忙，那么在小轩递给小渊的时候就不会再帮忙。反之亦然。

讲 一 讲

动态规划-传纸条（noip2008）

还有一件事情需要注意，全班每个同学愿意帮忙的好感度有高有低（注意：小渊和小轩的好心程度没有定义，输入时用 0 表示），可以用一个 0-100 的自然数来表示，数越大表示越好心。小渊和小轩希望尽可能找好心程度高的同学来帮忙传纸条，即找到来回两条传递路

径，使得这两条路径上同学的好心程度只和最大。现在，请你帮助小渊和小轩找到这样的两条路径。

输入的第一行有 2 个用空格隔开的整数 m 和 n，表示班里有 m 行 n （1<=m,n<=50）。

接下来的 m 行是一个 m*n 的矩阵，矩阵中第 i 行 j 列的整数表示坐在第 i 行 j 列的学生的好心程度。每行的 n 个整数之间用空格隔开。

讲一讲

动态规划-传纸条（noip2008）

输出共一行，包含一个整数，表示来回两条路上参与传递纸条的学生的好心程度之和的最大值。

3 3
0 3 9
2 8 5
5 7 0

输出：

34

讲一讲

动态规划-背包

背包九讲

01背包

完全背包

多重背包

混合背包

分组背包

……

讲一讲

动态规划-01背包

N个物品，每个物品体积vi，价值wi，求在包总体积V的情况下，放物品的最大价值。

采药

讲 一 讲

动态规划-01背包

辰辰是个天资聪颖的孩子，他的梦想是成为世界上最伟大的医师。为此，他想拜附近最有威望的医师为师。医师为了判断他的资质，给他出了一个难题。医师把他带到一个到处都是草药的山洞里对他说：“孩子，这个山洞里有一些不同的草药，采每一株都需要一些时间，每一株也有它自身的价值。我会给你一段时间，在这段时间里，你可以采到一些草药。如果你是一个聪明的孩子，你应该可以让采到的草药的总价值最大。”

输入格式：

输入文件medic.in的第一行有两个整数T（1 <= T <= 1000）和M（1 <= M <= 100），用一个空格隔开，T代表总共能够用来采药的时间，M代表山洞里的草药的数目。接下来的M行每行包括两个在1到100之间（包括1和100）的整数，分别表示采摘某株草药的时间和这株草药的价值。

输出格式：

输出文件medic.out包括一行，这一行只包含一个整数，表示在规定的时间内，可以采到的草药的最大总价值。

讲 一 讲

动态规划-01背包

输入样例：

70 3

71 100

69 1

1 2

输出样例：

3

讲 一 讲

动态规划-完全背包

N个物品，每个物品体积vi（每个物品有无数个），价值wi，求在包总体积V的情况下，放物品的最大价值。

根据第i种物品放多少件进行决策，所以状态转移方程为：

$$F[i][j] = Max\{F[i-1][j-k*C[i]]+k*W[i]\}, \quad 0 \le k*C[i] \le j$$

其中F[i-1][j-K*C[i]]+K*W[i]表示前i-1种物品中选取若干件物品放入剩余空间为j-K*C[i]的背包中所能得到的最大价值加上k件第i种物品。

讲 一 讲

动态规划-多重背包

多重背包问题的思路跟完全背包的思路非常类似，只是k的取值是有限制的，每件物品的数量是有限制，状态转移方程为：

$$dp[i][v] = \max\{dp[i - 1][v - k * c[i]] + w[i] \mid 0 <= k <= n[i]\}$$

讲 一 讲

动态规划-多重背包

多重背包转换成01背包

用二进制

将第i种物品的k个分成若干件物品，其中每件物品有一个系数，这件物品的费用和价值均是原来的费用和价值乘以这个系数。使这些系数分别为1,2,4,...,2^(k-1),n[i]-2^k+1，且k是满足n[i]-2^k+1>0的最大整数。例如，如果n[i]为13，就将这种物品分成系数分别为1,2,4,6的四件物品。

讲 一 讲

动态规划-背包例题

n个正整数a1,a2...an。选出若干个数，和为m。求多少种方案。n<100,m<100000.ai<1000.

讲 一 讲

动态规划-背包例题

n个正整数a1,a2...an。选出若干个数，和为m。求多少种方案。n<100,m<100000.ai<1000.

拓展：每个数字可以用无限次……

讲一讲

动态规划-分组背包

有N件物品和一个容量为V的背包。

第i件物品的费用是c[i]，价值是w[i]。

这些物品被划分为若干组，每组中的物品互相冲突，最多选一件。

求解将哪些物品装入背包可使这些物品的费用总和不超过背包容量，且价值总和最大。

讲 一 讲

动态规划-分组背包

分析：

每组物品有若干种策略：是选择本组的某一件，还是一件都不选。

也就是说设f[k][v]表示前k组物品花费费用v能取得的最大权值，则有：

f[k][v]=max{f[k-1][v],f[k-1][v-c[i]]+w[i]|物品i属于组k}。

最终可用一维数组实现，伪代码如下

顺序递推

Sample Problem15

加分二叉树（NOIp2003）

【问题描述】

设一个n个节点的二叉树tree的中序遍历为（l,2,3,…,n），其中数字1,2,3,…,n为节点编号。每个节点都有一个分数（均为正整数），记第j个节点的分数为di，tree及它的每个子树都有一个加分，任一棵子树subtree（也包含tree本身）的加分计算方法如下：

subtree的左子树的加分× subtree的右子树的加分＋subtree的根的分数

若某个子树为主，规定其加分为1，叶子的加分就是叶节点本身的分数。不考虑它的空

子树。

试求一棵符合中序遍历为（1,2,3,…,n）且加分最高的二叉树tree。要求输出；

（1）tree的最高加分

（2）tree的前序遍历

【输入格式】

第1行：一个整数n（n＜30），为节点个数。

第2行：n个用空格隔开的整数，为每个节点的分数（分数＜100）。

讲 一 讲

顺序递推

因为此题给我们的是中序遍历，所以如果a[i]为根，那么a[1]~a[i-1]就是这棵树的左子树，a[i+1]~a[n]就是这棵树的右子树。同样的如果j是a[1]~a[i-1]这棵子树的根，那么a[1]~a[j-1]就是左子树中的左子树，a[j+1]~a[i-1]就是左子树中的右子树，依次类推。

这样，我们只要枚举i~j这个区间和枚举这个区间的根k就可以了。至于路径，记录一下就可以了。

```
For i:=1 To n-1 Do
For j:=1 To n-i Do
For k:=j To j+i Do
If f[j,k-1]*f[k+1,j+i]+a[k]>f[j,j+i] Then
    f[j,j+i]:=f[j,k-1]*f[k+1,j+i]+a[k];
```

讲 一 讲

End

Thank you