# N - 这是一道比CCCC简单题更有想象力的中档题

##### Time Limit: 9000/3000MS (Java/Others)     Memory Limit: 65535/65535KB (Java/Others)

Submit Status

在2550年，CCCC已经成为了一场和ICPC一样正式的大型比赛

UESTC在那年派出了一只n个人组成的参赛队参加比赛

2050年的CCCC赛制有了改变，这n个人可以坐在一起讨论问题，一起写代码！

CCCC嘛，前面都是些不着边际的奇怪题目，一直到了最后的终极大BOSS题目

最后大boss的AC代码，将由很多队员一起写成！

已知n个参赛队员，对于第i个队员，每写一行代码，就会留下 aiai 个bug

最后一题需要写m行代码，请安排各个队员写的代码行数（显然要非负），使得整个代码的bug数不超过b个

然后，在ACM玄学之神的保佑下，这份不超过b个bug的代码就能AC了！

问你有多少种不同的安排方案可以写出一份AC代码，要求方案数对mod取模

## Input

第一行，n,m,b,mod，1<=n,m<=500,0<=b<=500,1<=mod<=1e9+7

接下来一行n个数，分别是 a1a1到 an,an, 0<= aiai <=500

## Output

对mod取模之后的方案数

## Sample input and output

| **Sample Input** | **Sample Output** |
| --- | --- |
| 3 3 3 100  1 1 1 | 10 |

题意：

N件物品，每件体积为ai,每件可以选无数次，选m件使得总体积不超过b，问总共多少种方法。

题解：

用dp[i][j][k]表示前i行代码由前j个人写错误数量不超过k个的方案数，我们可以很容易的得到递推公式：

dp[i][j][k]=dp[i][j-1][k](0<=k<a[j])

dp[i][j][k]=dp[i][j-1][k]+dp[i-1][j][k-a[j]]

(a[j]<=k<=b)

其中，dp[i][j][k]由第i行代码选择第j个人写和选择前j-1个人来写两部分方案数量之和来得到。时间复杂度O(nmb)，复杂度较高，不过由于n,m,b都很小，且程序时限较大，这样的方法不会超时。

还有一点需要注意，此题n最大为500，三维数组会导致空间不足，我们需要利用滚动数组降低空间复杂度。