整数划分模型

- 一类整数划分问题

- 对于一个正整数n：

- 0.求把n划分成k个非负整数的方案数?
- 1.求把n划分成k个正整数的方案数?
- 2.求把n划分成k个互不相同正整数的方案数?
- 3.求把n划分成k个不大于m的互不相同正整数的方案数?
- 4.求把n划分成k个奇数的方案数?

0.求把n划分成k个非负整数的方案数?

- 是不是很熟悉?
- POJ 1664 放苹果

- dfs+剪枝
- 递归+记忆化
- 递推/dp

0.求把n划分成k个非负整数的方案数?

- 递推/dp

- 有0？ //放苹果 有空盘子
- 没0？ //放苹果 没空盘子

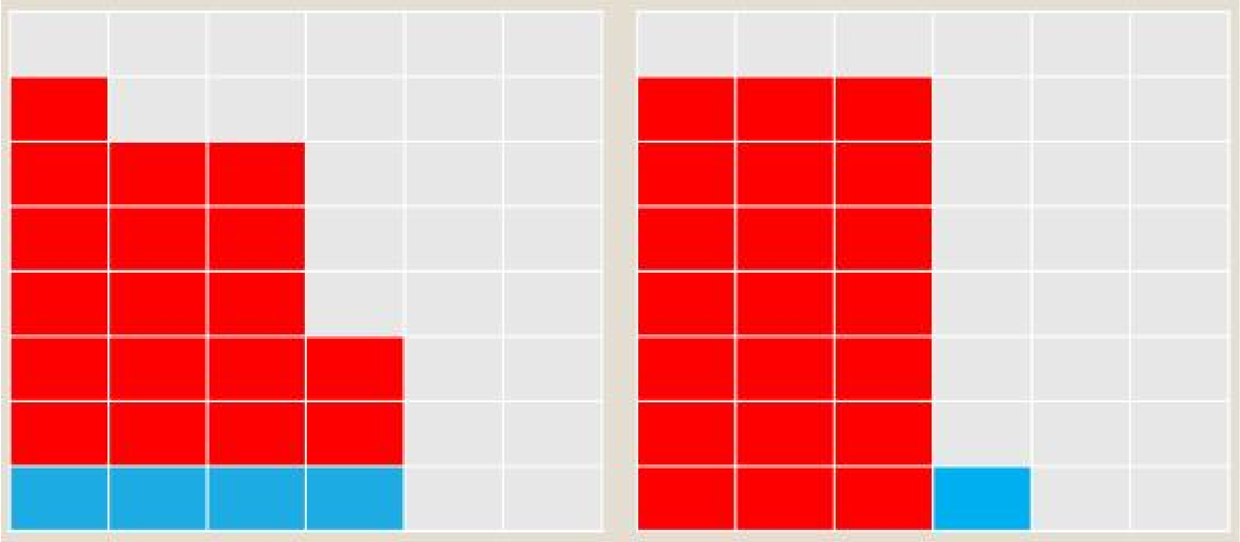
- dp[i][j] = dp[i][j-1] + dp[i-j][j]

1.求把n划分成k个正整数的方案数

- 暴力dp
- 设dp[i][j][sum]前i个数，选了j个数，和为sum的方案数是多少，答案就是dp[n][k][n]，考虑这一个数选几个来转移即可。这个状态是n*k*n的转移O(sum/i)。均摊O(n*k*n*log(n))复杂度有些高。
- 本质是个完全背包的，那种写法的话可以做到O(n*k*n)。

1.求把n划分成k个正整数的方案数

- 我们直接设dp[i][j]表示把i划分成j个正整数的方案数。



- 有1?
- 没1?

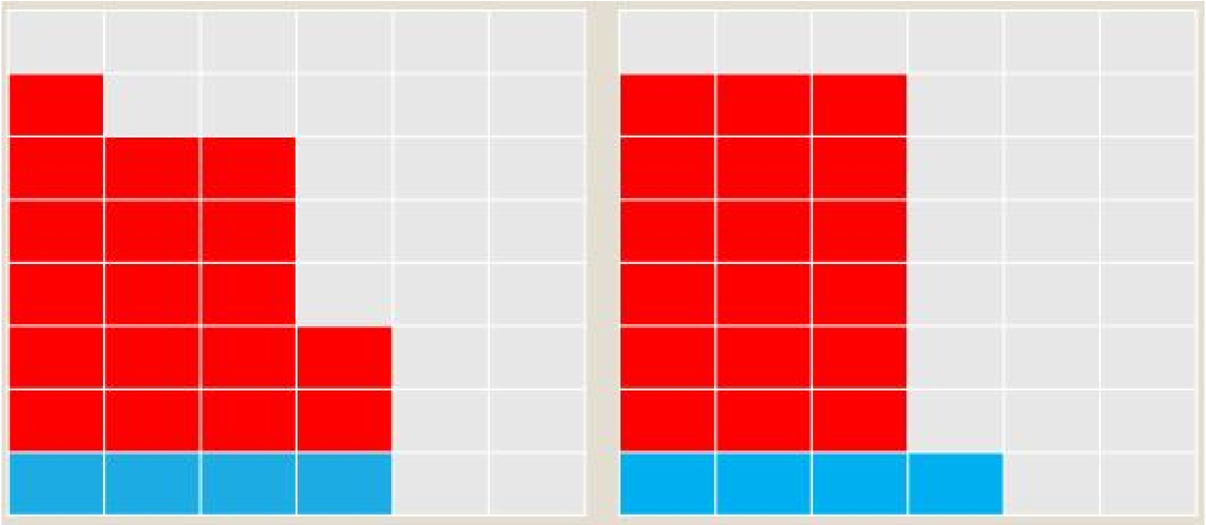
- 我们可以得到dp[i][j]=dp[i-j][j]+dp[i-1][j-1]。

2.求把n划分成k个互不相同正整数的方案数

- 大同小异。
- 暴力dp:设dp[i][j][sum] 前i个数，选了j个数，和为sum的方案数是多少，答案就是dp[n][k][n]，考虑下一个数选不选来转移即可。
- 本质是个0/1背包， 复杂度O(n*k*n)

2.求把n划分成互不相同的k个正整数的方案数

- 还是直接设dp[i][j]表示把i划分成j个数的方案数



- 有1?
- 没1?

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

3.求把n划分成k个不大于m的互不相同的正整数的方案数

- 背包dp的方法同上。
- N=20,k=4,m=6的不合法方案如图。
- dp[i][j]=
- dp[i-j][j]+dp[i-j][j-1]-dp[i-(m+1)][j-1]
- 减的这一项就是超过线的那种情况。

4.求把n划分成k个奇数的方案数

- 可重复数字纯奇纯偶划分
- g[i][j]:将i划分为j个偶数
- f[i][j]:将i划分为j个奇数
- g[i][j] = f[i - j][j]
- f[i][j] = f[i - 1][j - 1] + g[i - j][j]

- 不可重复的类似