

0-1背包

第一节（免费试听）

讲师：炼药师

我是班主任sunny,
加我领取课程福利哦



加班主任，进班级答疑群
快速获取面试资料/课程福利



关注公众号，了解大厂资讯

版权声明

九章的所有课程均受法律保护，不允许录像与传播录像
一经发现，将被追究法律责任和赔偿经济损失

- 重要性：1.动态规划中的重难点，2面试中的常考点
- 什么是背包问题
- 主要分类：0-1背包，完全背包，多重背包

- 爆搜，贪心的思路与局限
- 动态规划算法实例与代码

0-1 背包 爆搜

举例：背包容量 $m = 10$

物品大小 $A = [2, 3, 5, 7]$

物品价值 $V = [1, 5, 2, 4]$

爆搜解法：分别枚举每一个物体取或者不取，1代表取，0代表不取

状态	容量	价值	状态	容量	价值	状态	容量	价值	状态	容量	价值
0000	0	0	0100	3	5	1000	2	1	1100	5	6
0001	7	4	0101	10	9	1001	9	5	1101	12	×
0010	5	3	0110	8	7	1010	7	3	1110	10	8
0011	12	×	0111	15	×	1011	14	×	1111	17	×

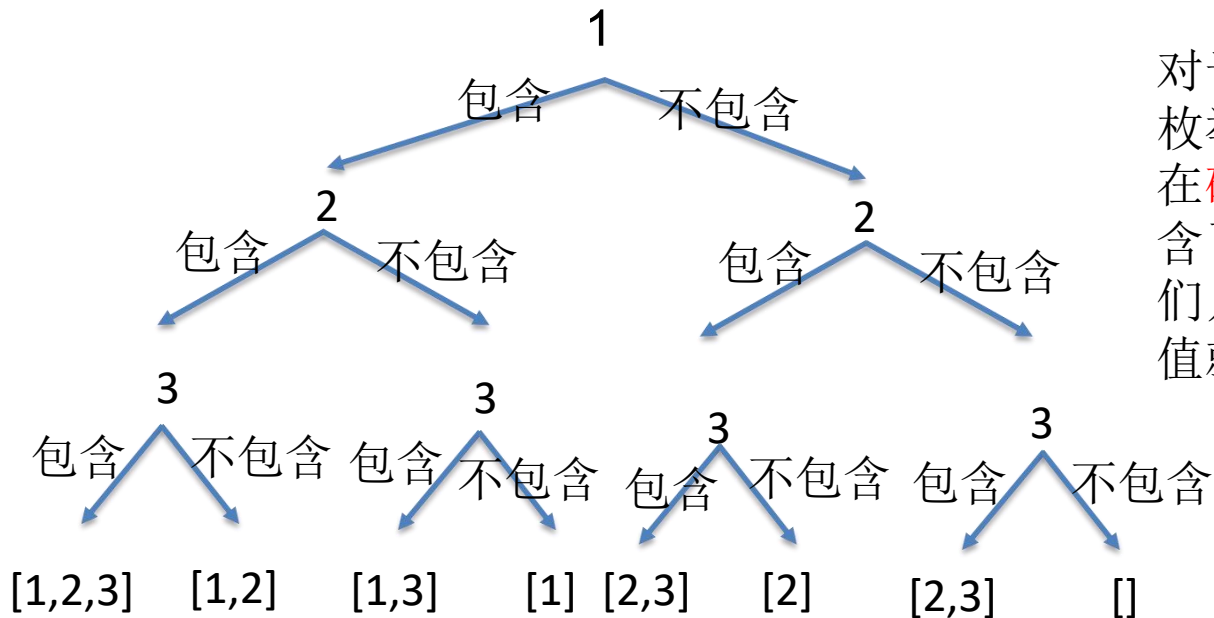
- 取价值最高:
- $m=2, A = [1, 1, 2], V = [2, 2, 3]$
- 贪心答案: 3, 正确答案: 4

- 取重量最轻
- $m=2, A = [1, 1, 2], V = [1, 1, 3]$
- 贪心答案: 2, 正确答案: 3

- 取单位价值最高
- $m=3, A = [1, 1, 3], V = [2, 2, 5]$
- 贪心答案: 4, 正确答案: 5

所有的贪
心，都是
错误的！！！！

爆搜算法的局限：



对于前*i*个物体，每个都枚举了取或者不取。
在**确定**容量*j*下，可能包含了**不同的情况**，但我们只需要知道**最好**的价值就可以了

0-1 背包 动态规划实现:

举例1: 背包容量 $m = 10$

物品大小 $A = [2, 3, 5, 7]$

物品价值 $V = [1, 5, 2, 4]$

使用数组来记录取前 i 个物品, 在容量 j 的情况下能取的最大价值

i/j	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0	0	0
1	0										
2	0										
3	0										
4	0										

0-1 背包 动态规划实现:

i / j	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	1	1	1	1	1	1	1	1
2	0	0	1	5	5	6	6	6	6	6	6
3	0	0	1	5	5	6	6	6	7	7	8
4	0	0	1	5	5	6	6	6	7	7	9

$dp[i][j]$ 表示前*i*个物体，在容量*j*的情况下，能取到的最大价值

如果取第*i*个物体，价值为 $dp[i - 1][j - A[i]] + V[i]$ ($j - A[i] > 0$)

如果不取第*i*个物体，价值为 $dp[i - 1][j]$

状态转移: $dp[i][j] = \max(dp[i - 1][j - A[i]] + V[i], dp[i - 1][j])$

0-1 背包 动态规划实现:

举例2: 背包容量 $m = 8$

物品大小 $A = [2, 3, 4, 5]$

物品价值 $V = [3, 4, 5, 6]$

i / j	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	0	3	3	3	3	3	3	3
2	0	0	3	4	4	7	7	7	7
3	0	0	3	4	5	7	8	9	9
4	0	0	3	4	5	7	8	9	10

作业: LintCode563 <https://www.lintcode.com/problem/backpack-v/>



扫描二维码关注微信/微博
获取最新面试题及权威解答

微信: [ninechapter](#)

知乎专栏: <http://zhuannlan.zhihu.com/jiuzhang>

微博: <http://www.weibo.com/ninechapter>

官网: www.jiuzhang.com