

DS–Autumn 2023 — Lab 1 Solutions

王子隆, SID 2221411126

2023 年 11 月 16 日

摘要

1. 问题描述

假设有一个能装入总体积为 T 的背包和 n 件体积分别为 w_1, w_2, \dots, w_n 的物品，能否从 n 件物品中挑选若干件恰好装满背包，即使 $w_1 + w_2 + \dots + w_m = T$ ，要求找出所有满足上述条件的解。

例如：当 $T = 10$ ，各件物品的体积1,8,4,3,5,2时，可找到下列4组解：

(1,4,3,2)

(1,4,5)

(8,2)

(3,5,2)。

2. 实现提示

可利用回溯法的设计思想来解决背包问题。首先，将物品排成一列，然后，顺序选取物品装入背包，若已选取第 i 件物品后未滿，则继续选取第 $i+1$ 件，若该件物品“太大”不能装入，则弃之，继续选取下一件，直至背包装满为止。

如果在剩余的物品中找不到合适的物品以填满背包，则说明“刚刚”装入的物品“不合适”，应将它取出“弃之一边”，继续再从“它之后”的物品中选取，如此重复，直到求得满足条件的解，或者无解。

由于回溯求解的规则是“后进先出”，自然要用到“栈”。

进一步考虑：如果每件物品都有体积和价值，背包又有大小限制，求解背包中存放物品总价值最大的问题解——最优解或近似最优解。

1. 数据结构实现

(a) 功能实现

(b) 应用

1.1 abc

Solutions:

1. 1 ArrayDeque.java

2. 2