

北京大学 2014 年研究生算法课第 4 次作业

发布时间：2014 年 11 月 17 日

截止时间：2014 年 12 月 1 日课前

注意事项：

- 作业应独立完成，严禁抄袭。作业必须使用统一规定的模板。
- 在截止日期那天，直接把纸质版的作业交给任课老师。
- 如果因生病等特殊原因不能按时完成作业的，那么应在截止日期前一天向任课老师请假。
- 作业和考试中的动态规划算法设计都要说明状态转移方程、转移条件、初始状态、返回值、时间复杂度以及空间复杂度。

动态规划

1. 题目来源：《算法设计》第六章动态规划第 2 题

题目描述：

假设你正在管理一个专业计算机黑客团队，每周你必须选择一项工作让团队来承担，工作有两类，低压的和高压的。如果你在第 i 周选择了一项低压的工作，那么你将得到 l_i 的收益，如果你选择高压的工作，你将得到 h_i 的收益。但是，需要注意的是为了在第 i 周取得高压的工作，第 $i-1$ 周必须什么也不做。第一周可以直接选择高压工作。

因此，给定 n 周的序列，请你给出一个“高压”，“低压”，“无”来组成的序列，使得这 n 周你的收益最大。

问题：给定一组值 l_1, l_2, \dots, l_n ，和 h_1, h_2, \dots, h_n ，请找出一个最大价值的计划。

例如：假设 $n=4$ ，且 l_i 和 h_i 的价值给在下表，那么最大价值的计划将是在周 1 选择“无”，周 2 选择高压，周 3 和周 4 选择低压。这个计划的价值是 $0+50+10+10=70$

	周一	周二	周三	周四
l	10	1	10	10
h	5	50	5	1

(a) 通过给出一个没有得到正确答案的实例证明下述算法不能正确求解这个问题。

For 迭代 $i=1$ to n

```

    If  $h_{i+1} > l_i + l_{i+1}$  then
        输出“在周 i 选择不工作”
        输出“在周 i+1 选择一项高压的工作”
        继续迭代 i+2
    Else
        输出“在周 i 选择一项低压的工作”
        继续迭代 i+1
    Endif
End

```

为了避免数组溢出问题，当 $i > n$ 时我们定义 $h_i = l_i = 0$

在你的例子中，说出什么是正确的答案且上述算法找到的是什么？

(b) 给出一个有效的算法，输出关于 l_1, l_2, \dots, l_n 和 h_1, h_2, \dots, h_n 的值并且返回一个最优计划的价值。

2. 题目来源：《算法设计》第六章动态规划第 9 题

题目描述：

你有一个高性能计算系统给，对于 n 天中的每天，你都会得到一定的数据量来处理，在第 i 天，为你提供的数据量是 x_i TB。每处理 1TB，你将会接收到固定的收益。没有处理的数据以后也不能再处理了。

由于你的计算系统的能力的限制，它一天只能处理一定数量 TB 的数据。而且这个系统处理能力在逐渐下降，比如第一天处理能力是 s_1 ，第二天就是 s_2 ，第 n 天是 s_n ，而且 $s_1 > s_2 > \dots > s_n > 0$ 。(当然，不管第 i 天你的系统运行多块，你最多处理 x_i TB 的数据)。为了使系统恢复性能，你可以重启机器，但在你重启的那一天，它将不能工作。

问题：给定接下去 n 天的有效数据量 x_1, x_2, \dots, x_n ，并且给定 s_1, s_2, \dots, s_n ，表达该系统的特征，选择你将要在哪些天重启系统，以使得处理的数据量达到最大。

例：假设 $n=4$ ， x_i 与 s_i 的值由下表给定。

	天 1	天 2	天 3	天 4
--	-----	-----	-----	-----

x	10	1	7	7
s	8	4	2	1

最好的解将是只在第二天重启；按照这种方式，你在第一天处理 8TB，接着在第 2 天处理 0TB，第三天 7TB，第 4 天 4TB，总计 19TB。

(1) 给出一个具有下述性质的实例：

- 存在数据剩余，即对每个 i ， $x_i > s_i$ 。
- 最优解重启动次数最少两次。

此外，对这个例子，你应该说出这个最优解是什么，你不必提供它是最优的证明。

(2) 给出一个有效的算法，对输入 x_1, x_2, \dots, x_n 和 s_1, s_2, \dots, s_n 的值并且返回由最优解处理的 TB 总数。

3. 题目来源：《算法设计》第六章动态规划第 20 题

题目描述：

假设快到期末了，你选修了 n 门课程，而且每门都有一道题目没有完成。每个课程将按照下面的登记进行评分：等级是正整数，从 1 到 g ($g > 1$)，等级越高越好，你的目标是在你的 n 个课题上的平均分最高。

你一共有 H 个小时来完成所有的课程 ($H > n$, H 是正整数)，每门课程投入的时间和得到的分数都有一个函数 h 。如果你在第 i 门课程上投入 h 个小时，那么你将得到一个 $f_i(h)$ 的成绩 (f_i 函数是非降的)。

这个问题就可以描述为：给定这些函数 $\{f_i\}$ ，请决定你在每个课程上花费多少小时 (整数) 以使得你的平均成绩最高。注：时间复杂度应该是一个 n, g, H 的多项式。