

提交答案题的一些技巧

宁波市镇海中学 梁晏成

2018 年 7 月 1 日

Overview

1 什么是提交答案题

2 一些小技巧

3 造计算机类提答

- APIO2017 考拉
- NOI2016 旷野大计算

4 复杂问题的特殊解类提答

- 分析数据
- 寻找通用解

5 总结

什么是提交答案题

一般的传统题，都是要求写一个满足题目条件的程序，同时它不依赖于输入的一些特殊性质

提答题则是给出问题的输入数据，只要求给出结果的一类问题。

提答题的问题一般比较复杂，有些需要通过输入数据的特殊性质设计针对的算法，有些则是利用题目给定的一些“元件”，实现一个能够完成某个功能的流程，还有的就是直接根据数据进行构造

相比于传统题，提交答案题对“分析输入数据”提出了比较高的要求

一些小技巧

- 可以用一个程序或者脚本同时测试10个数据点

一些小技巧

- 可以用一个程序或者脚本同时测试10个数据点
- 注意备份输出文件，以及造输出文件的程序，以备后面类似问题的使用

一些小技巧

- 可以用一个程序或者脚本同时测试10个数据点
- 注意备份输出文件，以及造输出文件的程序，以备后面类似问题的使用
- 在程序中利用checker实现检验解的过程

一些小技巧

- 可以用一个程序或者脚本同时测试10个数据点
- 注意备份输出文件，以及造输出文件的程序，以备后面类似问题的使用
- 在程序中利用checker实现检验解的过程
- 首先看完所有问题，根据性价比选择做题顺序

一些小技巧

- 可以用一个程序或者脚本同时测试10个数据点
- 注意备份输出文件，以及造输出文件的程序，以备后面类似问题的使用
- 在程序中利用checker实现检验解的过程
- 首先看完所有问题，根据性价比选择做题顺序
- 搜索或其余运行时间较长的算法，考虑输出进度方便估计时间

造计算机类提答

在提答题中，其中的一类是给定一些元件，你需要将它们组建起来来实现某个功能。

这类问题需要首先分析元件的性质，再针对各个问题具体分析。

APIO2017 考拉

有 N 个点排成一行，所有点的权值构成一个1-100的排列（你不知道这个排列）。你和考拉分别有 W 颗红石子和蓝石子，你们可以玩一个游戏，具体步骤如下：

- 你将所有 W 颗红石子分配给所有的点，并将分配结果告诉考拉
 - 考拉将它的所有 W 颗蓝石子分配给所有的点，并将分配结果告诉你
 - 如果一个点上的蓝石子个数 $>$ 红石子个数，那么卡拉将会得到这个点的权值
 - 卡拉会优先最大化它得到的点权和，如果有多解则最大化得到的点的个数，如果还有多解那么它会选择任意一个解
- 你需要在一定的步数内完成一些任务
- ~~讲道理这个题是一个交互题~~

分析与准备

对于一些地位相同的点，我们优先考虑给它们分配一样的石子

分析与准备

对于一些地位相同的点，我们优先考虑给它们分配一样的石子

显然需要用checker来验证，那么准备输入时考虑直接给第 i 个点分配 i 的权值，不影响结果且方便调试

任务一

任务一： $N = W = 100$ ，求出权值最小的点，返回下标
要求：次数 ≤ 2

任务一

任务一： $N = W = 100$ ，求出权值最小的点，返回下标
要求：次数 ≤ 2

随便选一个点放上一颗石子，那么考拉最多只能选99个点，
那么没被选的就是最小的

任务二

任务二： $N = W = 100$ ，求出权值最大的点，返回下标
要求：次数 ≤ 13 ， ≤ 4 为满分

任务二

任务二: $N = W = 100$, 求出权值最大的点, 返回下标

要求: 次数 ≤ 13 , ≤ 4 为满分

方法一: 选择十个点并且每个点放十个石子, 那么考拉最多从中选一个, 11次比较可以找打最大值

任务二

任务二: $N = W = 100$, 求出权值最大的点, 返回下标
要求: 次数 ≤ 13 , ≤ 4 为满分

方法一: 选择十个点并且每个点放十个石子, 那么考拉最多从中选一个, 11次比较可以找打最大值

方法二: 假设当前有 x 个备选, 那么每个点放 $\lfloor \frac{100}{x} \rfloor$ 个石子
淘汰尽量多的备选

经检验恰好4次找到最大值

任务三

任务三： $N = W = 100$ ，比较两个点的大小

要求：次数 ≤ 14 ， ≤ 3 为满分

任务三

任务三： $N = W = 100$ ，比较两个点的大小

要求：次数 ≤ 14 ， ≤ 3 为满分

分析个两个点都分配 x 的结果，讨论：

- 都分配8个，那么最多只能选择一个，如果都不选则
都 $\leq 2 + 3 + 4 + 5 + 6 + 7 + 8 = 35$
- 下一步都分配6个，那么最多只能选择一个，如果都不能则
都 $\leq 2 + 3 + 4 + 5 + 6 = 20$
- 下一步都分配4个，那么最多只能选择一个，如果都不能则
都 $\leq 2 + 3 + 4 = 9$
- 下一步都分配3个，那么最多只能选择一个，如果都不能则
都 $\leq 1 + 2 = 3$
- 最后给这两个都分配1个，结束

调整顺序即可获得满分

子任务四、五

子任务四、五：还原整个序列， $N = 100, W = 200/100$
要求：子任务五要求次数 ≤ 100

子任务四、五

子任务四、五：还原整个序列， $N = 100, W = 200/100$

要求：子任务五要求次数 ≤ 100

注意到问题的关键在于将整个序列分成两段，假设现在对权值在 $[l, r]$ 内的进行分裂：

如果 $r - l + 1$ 比较大或者 r 比较小，考虑给所有的点都分配少量的石子

否则考虑给所有的点分配较多的石子（大约为 $\sqrt{2N}$ ），找出最大值

实际上可以枚举分配的石子数并用程序验证能否实现分裂

NOI2016 旷野大计算

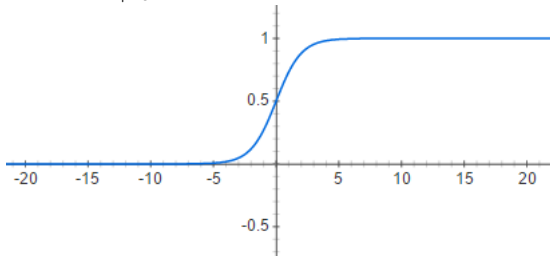
利用如下节点造一个能完成一定目的的计算机：

| 名称 | 操作符 (类型) | 操作数 | 计算结果 |
|-------|----------|--------|--|
| 输入节点 | I | 无 | 从终端读入一个实数作为 x_t |
| 输出节点 | O | i | $x_t = x_i$ ，并将 x_t 输出到终端 |
| 加法节点 | + | i, j | $x_t = x_i + x_j$ |
| 偏移节点 | C | i, c | $x_t = x_i + c$ |
| 取反节点 | - | i | $x_t = -x_i$ |
| 左移节点 | < | i, k | $x_t = x_i \cdot 2^k$ |
| 右移节点 | > | i, k | $x_t = x_i / 2^k$ |
| S型节点 | S | i | $x_t = s(x_i)$ |
| 比较节点 | P | i, j | $x_t = \begin{cases} -1 & x_i < x_j \\ 0 & x_i = x_j \\ 1 & x_i > x_j \end{cases}$ |
| Max节点 | M | i, j | $x_t = \begin{cases} x_i & x_i > x_j \\ x_j & x_i \leq x_j \end{cases}$ |
| 乘法节点 | * | i, j | $x_t = x_i \cdot x_j$ |

$$s(k) = \frac{1}{1 + e^{-x}}$$

NOI2016 旷野大计算

$$s(k) = \frac{1}{1 + e^{-x}}$$



注意到题目里面的数都是高精度小数，考虑卡精度来利用 $s(k)$ 的极端值

准备操作

对所有的节点类型写一个函数用来添加一个该类型的节点

任务一

任务一：计算 $-2a - 2b$

任务一

任务一：计算 $-2a - 2b$

$$-((a + b) \ll 1)$$

任务二

任务二：计算 $\frac{1}{1 + e^{17x}}$

任务二

任务二：计算 $\frac{1}{1 + e^{17x}}$

$$s(-(x + (x << 4)))$$

任务三

任务三：计算 $signal(x) = \frac{x}{|x|}$

任务三

任务三：计算 $signal(x) = \frac{x}{|x|}$

利用 $s(x)$ 的两个无穷大的值为 0 和 1, $s(0) = \frac{1}{2}$, 考虑计算 $p(x) \approx s(x \ll 500)$

$$2s(x \ll 500) - 1$$

任务四

任务四：计算 $|x|$

任务四

任务四：计算 $|x|$

考虑能否利用 $s(x)$ 得到一个关于 x ，且与 x 正负值相关的函数

利用 $s(x + p(x) \ll 500) = [x < 0]s(x) + [x \geq 0]$

注意到当 $|x|$ 比较小时， $\frac{s(x) - s(0)}{x - 0} \approx s'(0) = \frac{1}{4}$

于是，当 $|x|$ 比较小且 $x < 0$ 时， $s(x + p(x) \ll 500) = \frac{1}{2} + \frac{x}{4}$ 也就是

$$t(x) = s((x \gg 150) + p(x) \ll 500) = \frac{1}{2} + (x \gg 152)$$

最终我们只要求

$$|x| = x - ((\frac{1}{2} - t(x)) \ll 153) + p(x + eps) \ll 152$$

任务五

任务五：二进制转十进制

任务五

任务五：二进制转十进制

直接按顺序利用左移和加法即可

任务六

任务六：将 $[0, 2^{32})$ 的数 x 转化为二进制

任务六

任务六：将 $[0, 2^{32})$ 的数 x 转化为二进制

从高位到低位枚举 k ，将 x 与 2^k 比较，作差后转化为任务三
注意最后一位可以不需要比较直接输出

任务七

任务七，求 $x \text{ xor } y$

任务七

任务七，求 $x \text{ xor } y$

转化为二进制后可以只考虑 $x, y \in \{0, 1\}$ 的情况

猜测解的形式为 $x + y - f(x + y)$ ，那么需要满足 $f(0) = f(1) = 0, f(2) = 2$ ，联想到 $p(x)$ ，可以得到 $f(x) = p(x - 1.5) \ll 1$

任务七

任务七，求 $x \text{ xor } y$

转化为二进制后可以只考虑 $x, y \in \{0, 1\}$ 的情况

猜测解的形式为 $x + y - f(x + y)$ ，那么需要满足 $f(0) = f(1) = 0, f(2) = 2$ ，联想到 $p(x)$ ，可以得到 $f(x) = p(x - 1.5) \ll 1$

$$x + y - (p(x - 1.5) \ll 1)$$

任务八

任务八：计算 $\frac{x}{10}$

任务八

任务八：计算 $\frac{x}{10}$

联想任务四，我们找一个点 x_0 满足 $f'(x_0) = \frac{1}{10}$ 即可

任务九

任务九：对 a_1, \dots, a_{16} 排序

任务九

任务九：对 a_1, \dots, a_{16} 排序

由于没有判断语句，因此只能考虑冒泡或者选择排序，实现 $(x, y) \Rightarrow (\min(x, y), \max(x, y))$

注意到 $\max(x, y) = x - \min(x - y, 0)$ ，利用任务四可以实现

任务十

任务十：求 $a \times b \bmod m$

任务十

任务十：求 $a \times b \bmod m$

方法一：实现一个快速乘，需要求一个 $[0, 2m)$ 之间的数对 m 取模，至于一个比较函数即可

方法二：将 $s(x + \ln 3)$ 泰勒展开得到：

$$s(x + \ln 3) = \frac{1}{4} + \frac{3}{16}x + \frac{3}{64}x^2 + x^3 h(x)$$

通过 $(x + y)^2 - x^2 - y^2$ 得到 xy ，然后转 m 进制即可

复杂问题的特殊解类提答

这是最常见的一类提答题，一般都是一个复杂难解的问题，但是你只需要对特定的数据给出解法

分析数据

■ 通过压缩率分析数据的随机性

分析数据

- 通过压缩率分析数据的随机性
- 计算与某个数列的方差

分析数据

- 通过压缩率分析数据的随机性
- 计算与某个数列的方差
- 利用python画图

分析数据

- 通过压缩率分析数据的随机性
- 计算与某个数列的方差
- 利用python画图
- 使用程序验证猜想

分析数据

如果问题是在一张图上进行，一般有几个角度进行思考：

分析数据

如果问题是在一张图上进行，一般有几个角度进行思考：

- 分析点和边的关系：树、基环树、森林、完全图、完全二分图等

分析数据

如果问题是在一张图上进行，一般有几个角度进行思考：

- 分析点和边的关系：树、基环树、森林、完全图、完全二分图等
- 通过度为1,2,3的点的个数分析图的性质：链、若干条链、星型图等

分析数据

如果问题是在一张图上进行，一般有几个角度进行思考：

- 分析点和边的关系：树、基环树、森林、完全图、完全二分图等
- 通过度为1,2,3的点的个数分析图的性质：链、若干条链、星型图等
- 写个程序验证是否是二分图

分析数据

如果问题是在一张图上进行，一般有几个角度进行思考：

- 分析点和边的关系：树、基环树、森林、完全图、完全二分图等
- 通过度为1,2,3的点的个数分析图的性质：链、若干条链、星型图等
- 写个程序验证是否是二分图
- 树的高度，是否是二叉树等等

CTSC2016 科学考察队

给定一张图，图的边权有正有负，有若干个人从1号点走到 N 号点，每条边只有制定的若干个人可以走。一条边被经过多次值计算一次贡献。问最终经过的边权之和最大是多少，并给出方案

CTSC2016 科学考察队

给定一张图，图的边权有正有负，有若干个人从1号点走到 N 号点，每条边只有制定的若干个人可以走。一条边被经过多次值计算一次贡献。问最终经过的边权之和最大是多少，并给出方案

如果只有一个人，我们可以考虑直接用SPFA求最长路，如果有正权回路，考虑用 N 次bellman-ford来替代。

现在有多个人，我们考虑随机一个出发顺序，将一个人经过的路径的边都赋成零，然后再让下一个人出发。多次随机取最优解

这个做法易于实现而且对输入数据没有过多依赖，实现的好可以得到60-70的分数

搜索

搜索在提答题中是一种十分重要的手段，应用范围广且实现简单，而且提答题只需要最终结果的特点使得搜索速度慢的缺点被缩小，因此搜索常常能跑出最优解或者较优解

有些题目不要求输出最优解，这时可以考虑加大减枝的力度来加快速度

IDA*结合估价函数也是一个很强大的手段

UNR#2 解开尘封的仪式

给定一个DAG，每个点上有一个字符，求一条最长简单路，
满足路径上的字符构成一个回文串
可能会指定起点或终点

UNR#2 解开尘封的仪式

给定一个DAG，每个点上有一个字符，求一条最长简单路，满足路径上的字符构成一个回文串
可能会指定起点或终点

搜索，考虑从路径的中间向两端扩展；或者从两端向中间扩展

减枝：记录 $f_{i,j}$ 表示从 i,j 出发的最长度，假设当前拓展到 x,y ，答案为 z ，那么如果 $z + f_{x,y} \leq ans$ ，可以直接退出
可以在5h内跑出并获得100分

随机化算法

提交答案题允许程序运行较长的时间，因此随机化算法的随机次数可以很多，常常能取得不错的效果

随机化算法

提交答案题允许程序运行较长的时间，因此随机化算法的随机次数可以很多，常常能取得不错的效果

爬山算法和模拟退火算法是最常见、也是OI中较为实用的一类随机算法

爬山算法：每次从当前随机解向某个方向拓展，单次爬山往往能得到一个极大解；多次爬山后的结果往往非常接近最优解

模拟退火算法：爬山算法有时会陷入局部极值，因此模拟退火算法加入了一个“出错”概率，即拓展后即使不优也有一定转移的概率。“出错概率”一般随着时间逐渐减小

随机化算法

转移解的方法也很多，除了直接随机一个相近的解之外，还可以考虑通过随机一个局部并对这个局部求出最优解的方法进行转移

初始解也可以考虑用贪心的方式得到，这样往往能缩短与极大值的位置

总结

总得来说，做提答题有如下几个原则：

- 抓住数据的性质，算法往往具有极强的针对性
- 提答题的得分一般与花费的时间成正比，但需要注意得分的效率
- 遇到不会做的传统题往往可以由提答题打开局面，但不能被提答题打乱节奏。一般而言花费1h后，就需要根据其他题目的得分情况来决定是否继续做提答
- 做好的必要的准备工作，优化做题体验，有助于提高效率