

---

## 部分分练习

现行 NOI 系列赛事, 难度均有提升, 并且 day2 起飞的特点尤为明显

AK 离场已经不太现实, AC 一个题+一题高档部分分+一题基础暴力算法, 成为了选手们普遍追求的目标.

能顺利解决一个题, 需要大量的模型积累, 对知识性内容和技巧性内容烂熟于心, 考场临危不乱, 加之以不太差的运气.

不可不谓之“天时地利人和”, 作为一个在高水平比赛中没有现场 AC 过题目的垫底选手, 我实在不知道能帮各位什么

但拿好高档部分分和基础暴力算法, 往往只需要读清楚题意, 熟悉经典问题即可.

此外, 基础暴力算法还是打表找规律, 多项式插值的基础; 甚至在此之上再做优化有时能够得到满分做法.

常言到:“切个数据结构, 再切个数论, 剩下全都暴力打满, 没有你进不了的集训队.”, 足见其重要性.

所以, 这次课, 我会带领大家练习从看题面, 分析得分难度开始, 直到写暴力, 进行优化并对拍检验为止的“骗分流程”.

### tips

- 开场先读
  - - 编译参数 (有没有 c++11, 有没有扩大栈空间, 有没有优化)
  - - 提交方式 (用不用自己建立题目文件夹? 程序命名是什么格式?)
  - - 题目类型 (提交答案? 交互? 传统?)
  - - 每个题目的 TL, ML
- 读一个题
  - - 背景一般没用, 除非是非常陌生的问题
  - - 描述认真读, 尝试用程序语言或者数学语言**转述题意**
  - - 配合几个特点不一的小样例和其解释**检验题意把握**是否准确.
  - - 读输入输出格式 (是否有数据量过大使用玄妙方式生成, 输入中是否有表明 subtask 的参数, 是否有多组数据, 是否有强制在线)

信息足够了, 如果能秒就直接写. 如果不能就先写出基础暴力.

之后考虑换一个模型来解题或者优化暴力.

- subtask 怎么拿

- 
- - 先写出通用的输入, 以及判断 subtask 的代码.
  - - 善用 namespace, 分割不同子任务的代码.
  - - 从最简单的 subtask 开始. 分析特殊性质, 找到优化点, 写出代码, 与暴力进行检验.
  - - 确定没有能拿得更多的分数时, 检验不同 subtask 的输入输出, 调试信息, 是否互相影响, 确保分数拿到手.
  - 怎么暴力
    - - 一般而言模拟题意 + 搜索解空间就行了
    - - 比如计数题: 枚举可能的可行解, 模拟题意进行检验 + 可行性剪枝.
    - - 比如数据结构题: 直接模拟操作
    - - 最优化题目 (一般正解是图论经典问题或者 DP, 或者非常难...): 搜索 + 可行性剪枝 + 最优化剪枝
    - - 多写就会了
    - - 记得现场是有 python 可以用的,c++ 写起来麻烦不妨先试试用它,

## 题目列表

- [CSP-S 2019 树上的数](#)
- [CSP-S 2019 树的重心](#)
- [BJOI 2019 删数](#)
- APIO2019
  - - [1](#)
  - - [2](#)
  - - [3](#)
- [CTS\(C\) 2019 pearl](#)
- [NOIp 2018 填数游戏](#)
- [NOIp 2017 宝藏](#)
- [你在考场上遇到的那个不会做的题目](#)

---

对于如何拿高档次部分分,我说句实话,这和切掉一个题是一样难的,我帮不上你们.  
这是我这次培训中最后一次课了,我想说几句话.

- 扎实基础,勤能补拙.
- 积极主动,获取启发性思想.
- 心态调整.
  - - 降低期望 (并非概率论中的随机变量期望值)
  - - 拉高下限
  - - 力争平稳发挥
- 推荐写完一个题就去请求上厕所,顺便喝点水,回来吃点零食 (应该是允许拿的吧...).
- **除非 AK 了,不要提前离场,你一定会后悔的** (by LJF@HuiWen-naclyfish)

如果你在 OI 的训练中. 做足了准备,突破了自己的心理障碍,见识了广大的面向你的天地.  
那么即使你在比赛中失利,你也可以收起笔记本,取回你的文化课卷子,写下一句“尽吾志也而不能至者,可以无悔矣,其孰能讥之乎?”

---

上次课的  $n^2^n$  求和