

csp 赛前注意事项

代码方面：

- `#include <bits/stdc++.h>` 注意斜杠方向。
- `#include <vector>` 本地如果不写可能也会编译过，但评测时会 ce
- 线段树空间开四倍，动态开点线段树开 20 倍，图的邻接表开两倍。
- 最短路能用 dijkstra 就不用 spfa。
- `priority_queue` 默认大根堆，定义结构体重载运算符时将小于定义成实际的大于，才能将堆变成小根堆。
- `mt19937 rnd(chrono::steady_clock::now().time_since_epoch().count())` 生成随机数。调试时要将随机种子改成固定值，否则程序每次运行过程都不同，调试完记得改回随机种子。
- 对拍时 `gen` 的随机数要写随机种子！
- 构造极端数据测试边界情况，比如一条链的树，一个环的图，全为 0 的序列等。
- 注意逻辑与或非的优先级 (`! > && > ||`)，实在不确定加括号。异或运算优先级很低！不确定的都加括号强制运算顺序
- 多测注意初始化。不要 `memset`！
- 避免使用关键字作变量名 `x0 x1 y0 y1 time hash` 等。可以使用缩写或大小写或下划线区分，比如 `_time, Ha, nxt`
- 循环变量 `i j k` 别写混了！注意 `k` 是否在全局变量中使用过。
- 分清楚 `1` 和 `l`
- 注意每个变量可能的取值范围，是否要开 `long long`？使用 `-ftrapv` 编译选项检查。
- 关于取模：有些数不能取模，会到 $1e15$ 量级，统计答案时把他们和其他数乘起来时要先自身取一次模。比如 $1e5$ 个点，边权 $1e9$ 的图的最短路可能到达 $1e14$ 量级，统计答案和其他数乘起来时要先将最短路数值自身取模。
- 默写的模版不一定是！很多时候错误反而出现在输入，输出，初始化，模版。比如 `int gcd(int a,int b){return b?a:gcd(b,a%b);}` 是错的。错哪了？
- 数组下标是变量时注意变量范围，防止数组越界。使用 `-fsanitize=address` 编译选项检查。
- 多测判断无解结束 也要将这一组测试的输入数据读完！会影响之后的输入。
- `scanf("%d%d",&n,&m);` 不要将 `n,m` 读反
- 卡常数，不能让常数影响正确算法的得分。使用快读，用邻接表取代 `vector` 等。
- 计算程序使用空间时不能只算数组内存总和，要加上递归使用栈空间，还要加上 `stl` 使用的空间，比如 `deque,set,map,vector` 等。
- 编译命令：`g++ (-g) a.cpp -o a --std=c++11 -O2 (-Wall) (-ftrapv) (-fsanitize=address)`
- 可以单独写一个程序脚本来自动编译，运行样例。节省输长串指令时间。

思路方面：

- 反复读题：各个变量的取值范围？值域很小？边数很少？特殊性质？

- 思路方面：排序后贪心？二分处理 01 序列？费用提前计算 dp？枚举一些东西？容斥？离线？分块？
- 没思路从部分分做起，看特殊性质提示思路！或者人为增加限制条件思考算法，比如 $k = 0$ 可以怎么做？链可以怎么做？只有两种颜色可以怎么做？序列单调可以怎么做？ k 很小可以怎么做？
- 运行一个算法前先判断：图联通？图有重边和自环？序列有负数和零？

赛场策略方面：

- 试机时熟悉周围环境，知道自己座位的位置，卫生间的位置。熟悉电脑配置，键盘鼠标手感，测试电脑运行速度。
- 前一天晚上好好睡觉，不要熬夜水群。最好午睡，如果没有午睡习惯中午也最好休息一会。预留出足够的从家到考点的时间。没有必要在考点外等候的时候背板子。紧张是正常的。等候时保持冷静和清醒。
- 把手机和有通讯功能的手表放到背包里关机，把背包放到教室外面。可以将手机交由父母保管不要带进考点，出来的时候没有手机也能找得到家长。
- 开场将原下发样例原封不动复制一份到别处，防止误操作修改了样例输入输出。
- 开场不要着急写第一道题！通读四道题目，首先准确了解各个题题意，一些显然的可突破点（ k 很小？值域很小？），大体算法方向（数据结构？dp？字符串？），预估题目难度。
- 写代码三思而后行：这个做法是对的吗？需要用到哪些数据结构或算法？各个变量和函数如何定义？实现代码大概需要多长时间？
- 对于正解一定要写对拍！oi 赛场上没有任何挂分的余地。对拍是为数不多的检查出代码错误的方法。调不出代码也写对拍，面向小数据调试。
- 大胆猜结论和做法！可以使用电脑验证。场上只需要拿到结果和算法和标准的代码实现，并不需要严密的逻辑论证。
- 长时间没有思路就立即放弃正解！不要陷入沉没成本。把目前会的分数全部写出来，写部分分和暴力也能获得不错的成绩。
- 对最优化题目乱搞！不要放弃得分的机会，比如模拟退火，爬山，随机打乱序列贪心等。
- 将前一道题目处理妥当了再进行下一道题目，妥当到可以直接提交的程度。大多数情况下最后没有时间回头调试前面题目。
- 感到做法假了也不要大批量删代码，可以注释掉或者另存为一个文件修改。场上不知道自己的 hack 是不是对的，也许一会又发现之前的代码是对的了的。
- 随时 ctrl+s 保存代码，防止电脑死机，机房断电。
- 比赛时注意时间，最后一个半小时就要开始写暴力和部分分，不要再追求正解了。
- 思路和调试代码都没有进展的时候去上个厕所，放空一下大脑。可以洗个脸清醒一下。解题有时只需要一两下思维迸发的灵光。牺牲时间休息也不要让自己一直处于低效疲惫状态。
- 不要在考前对得分有估计和预期！每场比赛难易不同，没有估计分数线的价值。在场上也不要估计一等奖分数线和他人分数，全身心投入进做题中去。
- 不要受他人干扰，其他选手开场十分钟就开始敲代码大多只是在写板子和暴力，而不是切掉了第一题。任何情况下保持心态平和与自信。自己也不要故意影响其他选手。

- 考试最后十分钟不要对代码做大型改动！最后太紧张，大多数情况都会改错，最后仔细检查文件名，文件输入输出，考号。
- 不要注释 freopen！
- 非官方的民间数据全部没有任何可信度，同学的做法和自己的不一样大多情况下是他们错了。不要紧张焦虑难过，耐心等待官方数据和成绩即可。

祝各位选手都能在比赛中取得理想的成绩！