



# Woshiluo 的自我总结

Woshiluo<sup>1</sup>

My blog: <https://blog.woshiluo.com>

November 15, 2019

---

<sup>1</sup>背景来源：恋 × シンアイ彼女 CG

# Content

序

心态

考场心态

场外心态

读题

程序

代码以外的细节

代码中的奇妙细节

一点小东西

思路

场地

结



# 序

The background of the image is a vibrant, colorful illustration of a coastal city at sunset or sunrise. In the foreground, there's a lush green hillside covered in pink cherry blossom trees. A small, traditional-style building with a red roof sits atop the hill. The city below is densely packed with buildings, and a bridge spans a body of water. The sky is filled with soft, pastel-colored clouds.

老师们说要我们这群参加过 NOI 的写写总结  
虽然我参加过 NOI，但是打铁了...  
而且感觉过了一年感觉自己更菜了  
不过还是写一写吧，就当自己的总结

# 考场心态

我应该是整个机房里最没有资格讨论考场心态的人

# 考场心态

我应该是整个机房里最没有资格讨论考场心态的人

首先就是考场不要慌

NOI 系列比赛通常对你代码能力要求不大

根据经验，你考场上敲不出来的东西

绝大多数你的同学们也敲不出来

# 考场心态

我应该是整个机房里最没有资格讨论考场心态的人

首先就是考场不要慌

NOI 系列比赛通常对你代码能力要求不大

根据经验，你考场上敲不出来的东西

绝大多数你的同学们也敲不出来

不要想结果

比如 15ewzLy1 是不是已经 AK 了啊之类的

想结果往往分散注意力，不如好好想题目

虽然这个东西说来容易做来难

# 场外心态

众所周知，CSP-S2 是要考两天的

# 场外心态

众所周知，CSP-S2 是要考两天的

对于这种两天的考试，最好在 Day1 考完之后不要讨论题目一类的题目  
虽然 NOI 中间会强制社会活动

# 场外心态

众所周知，CSP-S2 是要考两天的

对于这种两天的考试，最好在 Day1 考完之后不要讨论题目一类的题目

虽然 NOI 中间会强制社会活动

然后就是不要自暴自弃

# 读题

如果有能力，一定一定要手推样例  
这个可以防止 90% 的理解问题

<sup>2</sup>参见……算了，不鞭尸了

# 读题

如果有能力，一定一定要手推样例  
这个可以防止 90% 的理解问题  
最后将思路在演草纸上确定个大概  
再上手键盘  
可以最大程度的减少不必要的动脑和代码

<sup>2</sup>参见……算了，不鞭尸了

# 读题

如果有能力，一定一定要手推样例

这个可以防止 90% 的理解问题

最后将思路在演草纸上确定个大概

再上手键盘

可以最大程度的减少不必要的动脑和代码

还有，虽然说 CSP/NOIP 中确确实实有原题的状况出现<sup>2</sup>

---

<sup>2</sup>参见……算了，不鞭尸了

# 读题

**如果有能力，一定一定要手推样例**

这个可以防止 90% 的理解问题

最后将思路在演草纸上确定个大概

再上手键盘

可以最大程度的减少不必要的动脑和代码

还有，虽然说 CSP/NOIP 中确确实实有原题的状况出现<sup>2</sup>

但是一定不要觉得自己做过就上手瞎写，容易白给

---

<sup>2</sup>参见……算了，不鞭尸了

# 代码以外的细节<sup>3</sup>

编译一定要加 `-Wall -Wextra -Wshadow`, 虽然会报一些无意义错误,  
总是有用的

---

<sup>3</sup>neta 「时间之外的往事」

# 代码以外的细节<sup>3</sup>

编译一定要加 `-Wall -Wextra -Wshadow`, 虽然会报一些无意义错误,  
总是有用的  
警惕 Dev-CPP 的默认头文件缓存, 它可能会让你少加头文件

---

<sup>3</sup>neta 「时间之外的往事」

# 代码以外的细节<sup>3</sup>



编译一定要加 `-Wall -Wextra -Wshadow`, 虽然会报一些无意义错误，总是有用的

警惕 Dev-CPP 的默认头文件缓存，它可能会让你少加头文件

注意不要错误的在 `freopen()` 中打开文件

特别是写入 `.cpp/.ans` 这种奇妙的问题

---

<sup>3</sup>neta 「时间之外的往事」

# 代码以外的细节<sup>3</sup>

编译一定要加 `-Wall -Wextra -Wshadow`, 虽然会报一些无意义错误,  
总是有用的

警惕 Dev-CPP 的默认头文件缓存, 它可能会让你少加头文件

注意不要错误的在 `freopen()` 中打开文件

特别是写入 `.cpp/.ans` 这种奇妙的问题

不要沉迷优化暴力

---

<sup>3</sup>neta 「时间之外的往事」

# 代码中的奇妙细节

小心变量初值问题，不合适/未定义的初值可能样例并不能检查出来这种问题

可以通过开 O2 来测试样例检测自己有没有 UB<sup>4</sup>

<sup>4</sup>未定义行为

<sup>5</sup>至少去年是 i7-8700k

<sup>6</sup>但是去年也有被卡 STL 常的人

# 代码中的奇妙细节

小心变量初值问题，不合适/未定义的初值可能样例并不能检查出来这种问题

可以通过开 O2 来测试样例检测自己有没有 UB<sup>4</sup>

内存也是一个自己测不出来的东西，一定一定要算内存 desu  
我已经见过两个人没算内存死亡了

<sup>4</sup>未定义行为

<sup>5</sup>至少去年是 i7-8700k

<sup>6</sup>但是去年也有被卡 STL 常的人

# 代码中的奇妙细节

小心变量初值问题，不合适/未定义的初值可能样例并不能检查出来这种问题

可以通过开 O2 来测试样例检测自己有没有 UB<sup>4</sup>

内存也是一个自己测不出来的东西，一定一定要算内存 desu

我已经见过两个人没算内存死亡了

虽然 STL 很快乐，而且 CCF 的机子也很快<sup>56</sup>

但是不开 O2 的 STL 因为封装过度，速度依然很感人

<sup>4</sup>未定义行为

<sup>5</sup>至少去年是 i7-8700k

<sup>6</sup>但是去年也有被卡 STL 常的人

# 代码中的奇妙细节

小心变量初值问题，不合适/未定义的初值可能样例并不能检查出来这种问题

可以通过开 O2 来测试样例检测自己有没有 UB<sup>4</sup>

内存也是一个自己测不出来的东西，一定一定要算内存 desu

我已经见过两个人没算内存死亡了

虽然 STL 很快乐，而且 CCF 的机子也很快<sup>56</sup>

但是不开 O2 的 STL 因为封装过度，速度依然很感人

特判写 namespace 有很大的快乐

不过注意分号有没有在特别的位置上

<sup>4</sup>未定义行为

<sup>5</sup>至少去年是 i7-8700k

<sup>6</sup>但是去年也有被卡 STL 常的人

# 思路

多上厕所，提神还能验证思路  
许多大佬们想题都是在上厕所时想起来的  
而且万一听到大佬在说正解呢？

<sup>7</sup>neta 了……不说了，大家都知道

<sup>8</sup>参见 CCF NOI WC 2017 挑战

# 思路

多上厕所，提神还能验证思路

许多大佬们想题都是在上厕所时想起来的  
而且万一听到大佬在说正解呢？

根据经验，第一直觉往往是错的，不要死想

<sup>7</sup>neta 了……不说了，大家都知道

<sup>8</sup>参见 CCF NOI WC 2017 挑战

# 思路

多上厕所，提神还能验证思路

许多大佬们想题都是在上厕所时想起来的  
而且万一听到大佬在说正解呢？

根据经验，第一直觉，往往是错的，不要死想  
面对数学题的最好方法，就是打表！<sup>7</sup>

顺带一提，遇字符串不决，就用哈希

<sup>7</sup>neta 了……不说了，大家都知道

<sup>8</sup>参见 CCF NOI WC 2017 挑战

# 思路

多上厕所，提神还能验证思路

许多大佬们想题都是在上厕所时想起来的  
而且万一听到大佬在说正解呢？

根据经验，第一直觉，往往是错的，不要死想  
面对数学题的最好方法，就是打表！<sup>7</sup>

顺带一提，遇字符串不决，就用哈希

$n^2$  过百万，暴力碾表算<sup>8</sup>

<sup>7</sup>neta 了……不说了，大家都知道

<sup>8</sup>参见 CCF NOI WC 2017 挑战

# 场地



这次还是在七十中  
小心特派员的嗓门，真的很大，不要试图和她杠  
不要管别人在干什么，特别是不要怕键盘声音，他可能只是在调 vimrc

结



嘛，但愿能帮助到大家吧  
总而言之，大家 RP ++  
「Fight だよ！」