#### **欢迎同学们在下方发帖补充常见错误，互相督促，赛场不要出锅**

### **致命错误——犯一条就零分**

* 未按题目要求添加文件读写
* 未按题目要求输出结果
* 忘写头文件
* C++万能头文件写错
* 主函数没有return 0

#### **编译问题**



#### **低级错误**

1. 文件读写错误或注释掉，或**"w"**和**"r"**写反，或英文拼写错误，或**".out"**写成**",out"**,**".int"**， 以及文件读写的过程中**"\*\*\*.in"**,**"\*\*\*.out"**,未写后缀名。 小tips:可以使用string类的加减操作(+".in", +".out")把freopen包装成一个函数使用，在主函数调用\_open("文件名前缀")即可。）
2. 输入数据中有n和m的，一定注意n和m的顺序（样例常常是相等的）
3. 看清楚给定的是字符串还是整数；
4. 写出正解，自己认为写错，没有调试直接交，编译错误。
5. 输出大小写要注意，**Yes**,**YES**区别开。
6. 在最后提交时，文件名粘贴复制时注意前后留存的空格。
7. 建图千万不要手抖，图建错了后面算法再写也不会对。
8. 调试信息、注释掉的语句正式提交前检查是否为最终版本。
9. 存数字的变量没有赋初始值，快读写挂。
10. 注意运算优先级问题，比如：!(a%b)和!a%b不等价，以及常错的位运算。

#### **运行超时**



1. 读取数据量很大时使用cin，导致超时。
2. bfs中多次入队，导致vector超过空间限制或者超时。
3. 一些大数组能用int就不必用long long，后者会拖慢效率。

#### **答案错误**

1. 存数字的变量没有赋初始值（windows会赋值为0，Linux随机赋值）；
2. 结构体内的函数定义的变量要初始化。
3. 快读写挂，建议不熟悉的人用scanf；
4. 对于常见的模，相加在范围，相乘在范围。（@所有人）
5. 一定要用大数据检验是否数组越界、乘法算术溢出；
6. 多组数据忘记清空数组；
7. 中间步骤计算量大，没取模，中间结果爆**int/long long**；
8. 计算结果为负数，直接取模将出错，考虑加一个模数的倍数，保证其为正后再取模。（@所有人）
9. ull与int或long long进行算术运算，结果出错；
10. 两个double不能直接用**==**比较，需要相减取绝对值与eps(根据题目精度要求设置具体值，比如1e-8)
11. 按位取反的时候先导0不会被考虑，用**（（1<<len）-1）-当前值**。

#### **运行错误**

1. 数组开小，RE；数组开大，超过空间限制；
2. 数组调用过程中越界，下标为负数或超过数组定义范围；
3. 大数组定义要放在主函数外面（map），大数组不能开在函数内部，取模要注意过程性（频繁）不能只给结果取模。容易爆；
4. 取模时注意模数可不可能为0，除零也会RE。

#### **算法逻辑**

1. **if()…if()… else…**结构不等价于**if()… else if()… else…**结构，逻辑错误；
2. 如果手写哈希难度不是太大，建议别开map，以上的数据容易被卡常；
3. 最短路SPFA，入队要做入队标记，出队时要做清空标记;
4. 使用某些数据结构时要check有没有在主函数内调用build()函数;
5. 开数组时**[maxn << 2]**注意不要写成**[maxn < 2]**(c++没有任何警告);
6. 按位取反的时候需要先导0不取反，用**（（1<<len）-1）-当前值**；
7. 写分块等算法时，注意区分每个变量代表当前块还是当前位置，最好避免在代码中将两种变量先后取同一变量名，如都用i变量进行遍历；
8. tarjan求scc时每次搜到一个点要入模拟栈；

#### **玄学错误**

1. 注意：要用你熟悉的语句去， fin >> f[i] >> g[i++] ; 在linux下是错误的，windows下也有类似问题： 如有函数 cal(i), 当**cout << cal (3） << cal (5) <<endl;**，其实是先调用 cal(5)。
2. **g[t++]=g[t-1]**多次使用t有时会出错，改写为：**t++; g[t]=g[t-1];**

#### **超过空间限制**

1. 数组一般不要定义在函数内部，定义数组时注意所占用空间，若使用vector注意不要一直添加元素
2. 数组占用空间计算方法，以int数组为例，一个int数据类型占用4个字节（Byte），则二维数组**int num[5000][5000]**占用空间 ,如果题目给的空间限制为128MB，则可以考虑定义这么大的数组。

* long long类型：8字节； float类型：4字节；double类型：8字节；char类型：1字节。

### **注意**

* 开O2对内存审查就更加严格，越界一个位置都要错，有同学**const int N=1e7+5**然后定义数组**s[N/100]**，由于整除，实际上就刚好开了1e5。
* 考试的时候,对比自己的输出和大样例输出时,用cmd的fc来判断,本身fc是不忽略行末空格的,用fc /W可以做到忽略行末空格（by 许星宇）