个人习惯

```cpp
/**
 * cf中不要轻易使用memset，尽量使用 for(int i = 0; i < n; ++i) 重置
 *      t = 1e5 n 永远等于 1 -> G
 *
 * n * m 等，如果n和m不相同等情况，多加考虑n和m，按n或m考虑
 *
 * https://www.cnblogs.com/lipoicyclic/p/12311394.html
 *
 * 对拍
 * https://blog.csdn.net/weixin_50624971/article/details/121185941
 * fast_OI 超级快读
*/
#include <iostream>
#include <vector>
#include <string>
#include <cstring>
#include <set>
#include <map>
#include <unordered_map>
#include <queue>
#include <ctime>
#include <random>
#include <bitset> // 是由int型拼接的， e.g. 1000位bitset 操作时间复杂度 O( 1000 / (大于等于 32))
#include <sstream>
#include <numeric>
#include <stdio.h>
#include <algorithm>
using namespace std;
    /**
     * 浮点数读入：
     *      double，少用float
     *      scanf("%.lf",&d);
     *      printf("%.f",d);
     *          --- 雨具说的 2023年8月4日
    */

// #define Multiple_groups_of_examples
#define dbgnb(a) std::cout << #a << " = " << a << '\n';
#define IOS std::cout.tie(0);std::cin.tie(0)->sync_with_stdio(false);
#define dbgtt cout<<" !!!test!!! "<<endl;
#define rep(i,x,n) for(int i = x; i <= n; i++)

#define all(x) (x).begin(),(x).end()
#define vf first
#define vs second

typedef long long LL;
// #define int long long  // 需要在LL定义之后
```

```
typedef pair<int,int> PII;

const int INF = 0x3f3f3f3f;
const int N = 2e5 + 21;
const int MOD = 1e9 + 7;
/**
 *  注意取模要注意，容易犯错，保险一点： (a + b % MOD) % MOD ==> ( (a + b % MOD) % MOD + MOD) % MOD
 *  这是因为 如果 -b > a 的话取模是负数，再次取模即可
 * e.g. ( -5 % 3 + 3) % 3 == 1
 *          -2 + 3 ==> 1 % 3 == 1
 */


// 当输入数据大于 1e6 时用快读
inline int fread() // 快读
{
    int x = 0, f = 1; char ch = getchar();
    while(ch < '0' || ch > '9') {if (ch == '-') f = -1; ch = getchar(); }
    while(ch >= '0' && ch <= '9') {
        x = x * 10 + (ch - '0');
        ch = getchar();
    }
    return x * f;
}

namespace direction {
namespace d8 {
vector<int> fx({0, 0, 1, 1, 1, -1, -1, -1}), fy({1, -1, 1, 0, -1, 1, 0, -1});
}
namespace d4 {
vector<int> fx({0,0,1,-1}), fy({1,-1,0,0});
}
}
namespace fast_IO {
    inline char read() {
        return getchar();
        static const int IN_LEN = 1000000;
        static char buf[IN_LEN], *s, *t;
        if (s == t) {
            t = (s = buf) + fread(buf, 1, IN_LEN, stdin);
            if (s == t) return -1;
        }
        return *s++;
    }
    template<class T>
    inline void read(T &x) {
        static bool iosig;
        static char c;
        for (iosig = false, c = read(); !isdigit(c); c = read()) {
            if (c == '-') iosig = true;
            if (c == -1) return;
        }
        for (x = 0; isdigit(c); c = read())
            x = ((x + (x << 2)) << 1) + (c ^ '0');
        if (iosig) x = -x;
    }
    const int OUT_LEN = 10000000;
    char obuf[OUT_LEN], *ooh = obuf;
    inline void print(char c) {
        if (ooh == obuf + OUT_LEN) fwrite(obuf, 1, OUT_LEN, stdout), ooh = obuf;
        *ooh++ = c;
    }
    template<class T>
    inline void print(T x) {
        static int buf[30], cnt;
        if (x == 0) {
            print('0');
        }
        else {
            if (x < 0) print('-'), x = -x;
            for (cnt = 0; x; x /= 10) buf[++cnt] = x % 10 + 48;
            while (cnt) print((char)buf[cnt--]);
        }
    }
    inline void flush() {
        fwrite(obuf, 1, ooh - obuf, stdout);
    }
}
void numeric() { // 数学
    LL k = 1e17;
    // cmath库里的数学尽量先乘个 1.0
    double d = sqrt(k * k); // × 会爆  原因：k*k 计算完转double，但是LL超了，因此是计算
错误
    double d = sqrt(1.0 * k * k); // √
    // 同理LL
    int ik = 3;
    LL l = (LL)ik * ik; // 麻烦
    LL l = 1LL * ik * ik; // 敲代码好敲些
}

void inpfile();
void solve() {

    map<int,int> mii;
    unordered_map<int,int> umii;
    set<int> si;

    int n; cin>>n;
    vector<int> vi(n);
    vector<PII> vpi(n);
    vector<vector<int>> f(n, vector<int>(2,0)); // c++14 auto dfs中用vector数组不
行，（或者我现在的编译器不支持

    // lambda表达式
    auto lam = [&](int a, int b) -> int {
        return a > b ? a : b;
    }; // 注意  逗号
}
int main() // signed main()
{
    #ifdef Multiple_groups_of_examples
    int T; cin>>T;
    while(T--)
    #endif
    solve();
    return 0;
}
void inpfile() {
    #define mytest
    #ifdef mytest
    freopen("ANSWER.txt", "w",stdout);
    #endif
}
```