

# 愉快的代码【8.12】

---

【7.27】我们特意加入了题解系统，也就是说，姐姐也会重新做一遍给你们的题目（包括选做），然后在第二天的题目前给出姐姐自己的代码和注释作为题解或参考

如果觉得自己的代码略为臃肿，可以参考对比一下姐姐的代码；

如果觉得姐姐的代码不如自己的优秀，也可以尽情地嘲讽姐姐～

【7.30】我们特意加入了团队系统，因为感觉到你们有点像是独立学习的样子，比如说姐姐和你们之间有交流，但是你们之间有没有交流呢姐姐就感受不到啦

所以正好在洛谷上发现了一个团队系统，我们可以在这上面布置作业呀（当然姐姐也会继续以 pdf 形式布置作业，你们也还是要以 pdf 形式交作业哈），然后你们就可以在上面看到其它妹妹们的代码呀（包括 AC 代码和还未 AC 的整个过程的代码和分数呀），觉得她们表现不够自己好的话，就可以在群里尽情地嘲笑她们呀～

然后那上面还有一个比赛功能哇，具体形式和我们平时的机考差不多，暑假差不多结束了我们也会有一次期末模拟机考的哈～

如果你们开心的话，你们也可以联合起来给姐姐布置一次平时的作业呀，或者给姐姐安排一次机考呀，你们都是团队的管理员了哈

【8.1】准备给你们留个有趣的团队大作业：给姐姐安排一次机考～

具体时间、题数、难度、知识点待定～

【8.3】经过了某些人性与道德的思考，得出了一个奇怪的想法：

“我今天把代码解决了，明天姐姐的代码还有兴趣看嘛”

那就当天放出来好啦～

同样地：如果觉得姐姐的代码不如自己的优秀，也可以尽情地嘲讽姐姐～

## 今天的题目：

知识点：数论

今天依然是快乐的数学问题～

- 1、<https://www.luogu.com.cn/problem/P1029>
- 2、<https://www.luogu.com.cn/problem/P2789>（略难）
- 3、<https://www.luogu.com.cn/problem/P6462>（快乐题）

（我们只是要求每天一道题而已哈，其它的都是选做哇）

## 今天的答案：

### 8.12问题1：

```
/*
    洛谷P1029：最大公约数和最小公倍数问题
    思想：如题所示，这题考的是gcd和lcm的知识点。设  $\text{gcd}(x,y) = k$ ，则  $x = a*k$ ， $y = b*k$ （且ab互质）。因此  $\text{lcm}(x,y) = a*b*k$ ，最终得到  $x*y = \text{gcd}(x,y)*\text{lcm}(x,y)$ 
    那么对于本题，即  $P*Q = x_0*y_0$ 。我们枚举P，即可求出Q，然后判断gcd(P,Q)是否为x0即可。
    （如果满足gcd(P,Q)=x0，自然满足lcm(P,Q)=y0，为什么？自己想一下~）
    时间复杂度：O(y0-x0)
*/

#include <iostream>
#include <algorithm>
#define ll long long
using namespace std;

ll gcd(ll x,ll y) { // gcd板子
    return y == 0 ? x : gcd(y,x%y);
}

int main() {
    ll i,j,x,y,ans = 0;
    cin >> x >> y;
    for (i=x; i<=y; i++) // 显然，P、Q不会小于x大于y
        if (x*y%i == 0 and gcd(i,x*y/i) == x) ans++; // x*y会爆int
    cout << ans << endl;
}
```

### 8.12问题2：

```
/*
    洛谷P2789：直线交点数
    思想：表面上是一个数学问题，实际上是一个递归。（思路略为难想）我们设这n条线中有i条互相平行，并且与其它的n-i条均不平行，那么则i条线会和n-i条线有i*(n-i)个交点。那剩下的n-i条线呢？递归一下吧
    时间复杂度：O(2^n)
*/

#include <iostream>
#include <algorithm>
#define ll long long
using namespace std;

int sum,a[301]; // a[i]：是否存在交点数为i的方式
void digui(int n,int num) { // 剩余边数、目前交点数
```

```

    if (n == 0 and a[num] == 0) { // 终止条件
        a[num] = 1;
        sum++;
    }
    for (int i=1; i<=n; i++) digui(n-i,num+i*(n-i)); // 枚举有多少条线平行的情况
}

int main() {
    int i,j,n;
    cin >> n;
    digui(n,0);
    cout << sum << endl;
}

```

## 8.12问题3:

```

/*
    洛谷P6462: 补刀
    思想: 如果你玩过或者看过别人玩LOL等某类游戏, 你就会明白补刀是多么的快乐~
    时间复杂度: O(t)
*/

#include <iostream>
#include <algorithm>
#define ll long long
using namespace std;

int main() {
    ll i,j,t,h,x,y,c,r;
    cin >> t;
    while (t--) {
        cin >> h >> x >> y; // 血量、塔攻击力、英雄攻击力
        if (y == 0) cout << "No" << endl; // 英雄攻击力为0
        else if (x == 0) cout << "Yes" << endl; // 英雄攻击力不为0, 塔攻击力为0
        else { // 均不为0
            c = h/x; // 塔能打多少下 (体现在这一下打不死, 下一下打死)
            r = h%x; // 小兵被塔打了c下剩余的血
            if (h%x == 0) { // 如果塔打了c下恰好打死, 这种情况下塔要少打一下
                c--;
                r += x;
            }
        }
    }
}
/*

```

最基础的想法是英雄打c+1下是否能打死小兵(剩余的血), 于是有  $r \leq y \cdot (c+1)$   
 但是由数据范围得知乘法会爆long long, 因此修改为除法  
 $(r+y-1)/y \leq c+1$ : 小兵剩余的血除以英雄攻击力是否小于等于英雄最多攻击次数  
 为什么是  $(r+y-1)/y$ ? 这是整数的向上取整除法  
 (另一种方法: 采用\_\_int128, c++128位整数)  
 (又另一种方法:  $r \leq y \cdot (c+1)$  or  $y \cdot (c+1) < 0$ , 自己思考一下为什么叭~)

```
        */  
        if ((r+y-1)/y <= c+1) cout << "Yes" << endl;  
        else cout << "No" << endl;  
    }  
}  
}
```

## Interesting thing:

又没啦