

# [题解]Div2 2019下半学期周赛(一)

## A 贪心

由于n个人的单独体重都不超过m。所以我们先进行排序。然后尽量让重的和轻的人匹配，如果不超过m，则这两人一艘船。否则，优先考虑先让重的人上船，然后轻的再和重的匹配。这样一直反复操作，直到所有人都上船。

时间复杂度： $O(n\log n)$

## B 模拟

主要体会一下pair的用处。使用 `map<pair<int, int>, int>` 把走过的路径坐标映射成对应的下标，我们只要走过的坐标点曾经出现过，就说明这是一段无效的路径，然后更新其下标，不断的更新，找出最小值。

map每次操作的开销都是 $O(\log n)$ 的，所以时间复杂度为： $O(n\log n)$

(下面简单介绍下 `std STL pair`)

就是个二元结构体。

用法: `pair<类型1, 类型2> var_name` 例 `pair<string, int> p1`

```
struct pair {
    typeName1 first;
    typeName2 second;
};
```

好处: 内部重载了运算符，可以进行逻辑比较

比较操作 `==, !=, <, <=, >, >=`

比较的时候，先比较first，first相等才比较second

生成新的 pair 可以用 `{T1, T2}` 或者 `make_pair(T1, T2)`

## C GCD

其实简单来说就是p刀+q刀-重复切的次数。

那么问题就转化成如何计算重复切的次数了。

比如4和6 那么我们会发现2刀是重合的

因为我们的目标是减少刀数，所以要尽量重复利用，那么重复切的最大次数就是 $\gcd(p, q)$  答案即为： $p + q - \gcd(p, q)$

## D 打表找规律

首先我们会发现0和它自身是一定会有的答案。

后面我们手动算几个就会发现 $n=1,2$ 时,为2,  $n=3,4$ 时为3,  $n=5,6$ 时为4

那么我们就可以观察出规律为 $2 + \frac{n-1}{2}$

时间复杂度:  $O(1)$

## E 签到

---

本场签到题。两个人分三盒糖果, 多希望尽可能的多, 但如果比对方多了, 就会丢去多的部分。

考虑假设 $a \leq b \leq c$ , 两人分别取 $a$ 和 $b$ , 则有 $b \leq a + c$

所以最终的答案会是两人平分糖果即 $\lceil \frac{a+b+c}{2} \rceil$ 。注意数据范围即可。

## F 思维递推

---

最坏的情况下每道题肯定要试满 $a[i]$ 次, 每次错误后又会回到第一题, 所以试错需要的次数是 $(i-1) * (a[i] - 1)$ 次, 即 前面的题数\*自己的试错次数。

那么总的递推即:  $\sum_{i=1}^{n-1} a[i] + (i-1) * (a[i] - 1)$

## G 01背包/dfs/贪心

---

这里只讲 $dp$ 的解法, 视频题解里会讲三种解法(贪心,  $dp$ ,  $dfs$ )。

$dp$ 解法就是01背包, 每个食物只能选一次, 食物的热量代表它的体积和价值, 那么我们就可以把背包容量设成前面的热量和 $s$ , 接下来只需要跑一遍01背包的板子得出在当前体积的最大价值多少即可, 01背包这里不讲...之前给了视频让大家看过(视频题解会再讲一下)。

那么答案就是 $dp[s] - dp[s - m]$ , 热量和的最大价值-热量和减去目标热量的最大价值, 就是目标 $m$ 的最小价值。