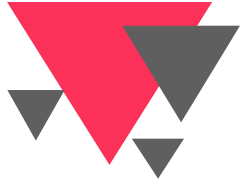


Codeforces183E Candy Shop

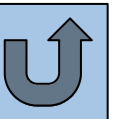
- 题目大意:
- 有N个小朋友，第i个小朋友手上有 $money_i$ 的钱，1块钱能买1颗糖果。
- 货架上按顺序摆着M个糖果袋子，第i个糖果袋子里装着i个糖果，价格为i。每袋糖果不一定要被买。
- 小朋友们要按顺序买若干轮（不确定），每轮每个人按顺序买一袋。第N个人买完后，如果不是最后一轮的话第1个人继续买。每轮必须从第1个人买到第N个人，一个都不能跳过。
- 除了第一个人第一次买，每个人只能买比上一次卖出的那袋糖果多的一袋糖果，而且不能超过他的支付限额。
- 给出N,M,以及每个小朋友手上的钱数，要求求出最多能卖出的糖果数。
- $N \leq 2 * 10^5$, $M \leq 5 * 10^6$, $N \leq M$, $money_i \leq \frac{M(M+1)}{2}$

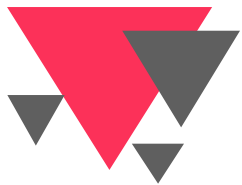




Codeforces183E Candy Shop Tutorial

- 题目分析：
- 首先枚举 P ，即购买的轮数。
- 因为一共有 M 件物品，所以一共要枚举 M/N 次。
- 现在我们假定 P 已经确定。那么如果第 i 个小朋友一共买了 X 颗糖果，那么第 $i+1$ 个小朋友至少买了 $X+P$ 颗糖果。既然这样，不妨假设 $money_i \leq money_{i+1} - P$ 。
- 并且一个小朋友先后拿的两个糖果袋中的糖果数至少相差 N ，否则中间的糖果袋就不够 $N-1$ 个人购买了。
- 假设有 P 轮，那么第 i 个人至少要花的钱就是 $P*i + (P*(P-1)/2*N)$ 。（然而这其实和后面讲的没有什么卵关系，只是先写出来，实际打代码的时候处理 $money$ 的时候方便点）



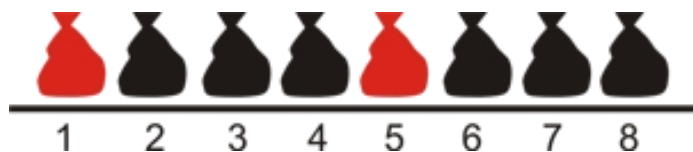


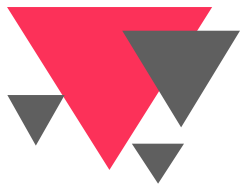
Codeforces183E Candy Shop Tutorial

- 为了使接下来的讨论更加形象，我们举个例子。
 - 假设 $N=3$ ， $M=8$ ，三个小朋友分别拥有7，10，14的钱，一共进行两轮购买。



- 假设我们已经知道了第一个小朋友每轮拿了哪些。（下面红色的两个）

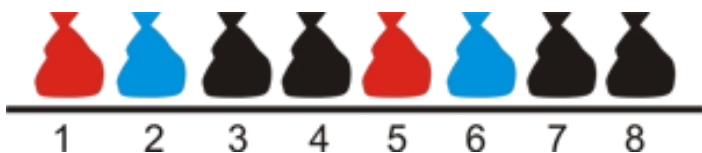




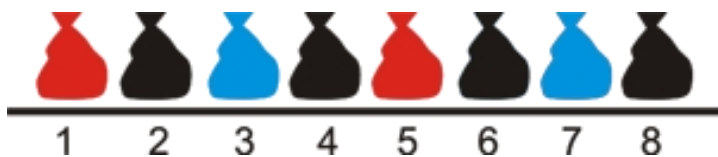
Codeforces183E Candy Shop Tutorial

- 贪心策略:

- 首先使下一个小朋友购买符合条件且数量最少的糖果袋。(如下图)



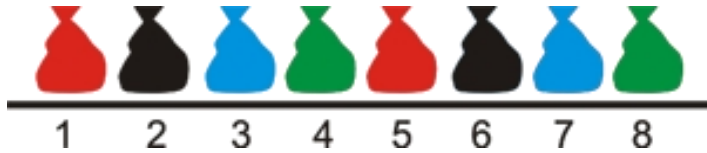
- 接下来, 如果他剩下的钱依然大于他购买的糖果总数, 并且他买的糖果中至少有一袋能够换成更多的一袋(向右移一格), 不影响接下来的人的购买, 那么就换成更多的。具体是哪儿袋换成更多的不要紧, 因为对答案是没有影响的。





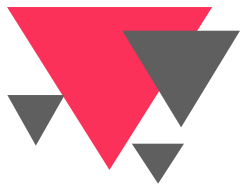
Codeforces183E Candy Shop Tutorial

- 而且因为 $money_i \leq money_{i+1} - P$,所以只要有空位, 第三个小朋友一定有购买的方案。然后像这样继续处理第三个小朋友。



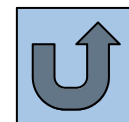
- 然后一直处理到最后一个小朋友就可以了。这样处理出的答案一定是最优解（在第一个确定的人的情况下），因为我们没有方法将任何一个人的物品向后移使得答案增大了。

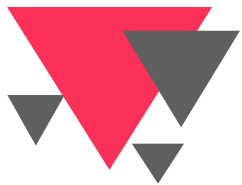




Codeforces183E Candy Shop Tutorial

- 如果我们知道了第一个小朋友购买了哪些，那么我们就能够在 $O(N)$ 的时间中得出答案。我们只需要算出按照价值最小的可能方案放完之后应该还有多少个空位（称之为FREEDOM，如下图中黄色框部分）。
- 处理每个人的时候，先加上这个人基本的贡献，然后把FREEDOM和他手上剩下的钱数取一个Min,这个值代表他把他的物品向右移了几位。然后把FREEDOM减去这个值，再把答案加上这个额外的值就可以了。



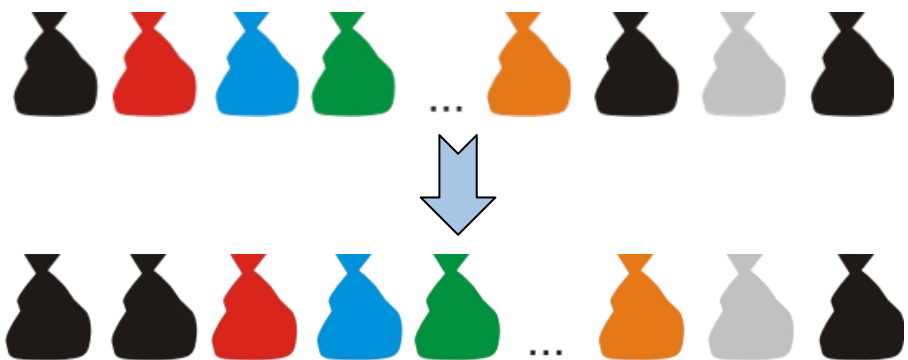


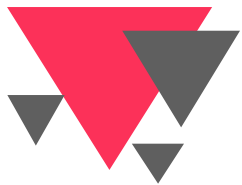
Codeforces183E Candy Shop Tutorial

- 然后我们现在假设我们并不知道第一个人选了哪些。
- 我们认为：第一个人应该买尽量多的糖果。不然既然他手上还有钱，那就可以把他手上的某个物品向右移（如下图），不会使答案变劣。



- 如果不能直接向右移，那么如果还有FREEDOM，还可以让在它后面的向右移，给他腾一个位置出来，也不会使答案变劣。





Codeforces183E Candy Shop Tutorial

- 然后容易知道：
 - FREEDOM只和第一个人第一次选哪个有关。
 - 假设第一个人选了第 i 个，那么 $FREEDOM = M - N * P - i + 1$ 。
- 因此为了让FREEDOM尽量大，第一个人第一次应该选得尽量小。
- 然后我们就要在第一个人买尽量多的情况下，让他第一次买的尽量少。
 - 这个可以 $O(1)$ 求出。（上面写的其实我不知道怎么求QAQ 请务必放过我）
- 总时间复杂度就是 $O(M/N)$ （枚举 P ）* $O(N)$ （单次计算答案）= $O(M)$

