



# РАЗБОР ЗАДАЧ

Финал



**Кубок  
Тамтэк**  
*по спортивному  
программированию*  
**2015**

# ЗАДАЧА А

«А ну ка... собери»

Time limit: 1 sec

Memory limit: 256 M



# Постановка задачи

- Даны мониторы трёх типов:  $V$ ,  $D$  и  $VD$  и системные блоки трёх типов:  $V$ ,  $D$  и  $VD$
- Мониторы  $V$  можно соединить с блоками  $V$  и  $VD$ , мониторы  $D$  с блоками  $D$  и  $VD$ , мониторы  $VD$  с любыми блоками

# Решение (100 баллов)

- Построим набор комплектов:
- Будем соединять между собой  $V$  мониторы и  $V$  блоки пока такие есть
- Аналогично для  $D$  мониторов и блоков.
- Соединим  $VD$  мониторы с оставшимися  $V$  и  $D$  блоками,  $VD$  блоки с оставшимися  $V$  и  $D$  мониторами
- Соединим оставшиеся  $VD$  мониторы и блоки

# Решение (100 баллов)

- Несложно проверить, что любой другой набор можно привести к этому действиями, не уменьшающими количество комплектов
- Значит этот построенный комплект – максимальный
- Итоговая сложность  $O(1)$



# ЗАДАЧА В «Бассейн»

Time limit: 1 sec

Memory limit: 256 M

# Постановка задачи

- Есть  $n$  сеансов,  $k$  шкафчиков
- После  $i$ -го сеанса  $a_i$  посетителей закрывает шкафчик и  $b_i$  оставляет открытым
- Распределяя шкафчики нужно минимизировать количество открытых шкафчиков после всех сеансов

# Решение (100 баллов)

- Будем жадно минимизировать количество открытых шкафчиков после каждого сеанса
- Если перед  $i$ -м сеансом открыто  $V$  шкафчиков, то
- Если  $b_i \geq V$ , то после сеанса будет открыто  $b_i$  шкафчиков
- Если  $b_i < V$ ,  $a_i \geq V - b_i$ , то  $b_i$  шкафчиков
- Если  $b_i < V$ ,  $a_i < V - b_i$ , то  $V - a_i$  шкафчиков
- Итоговая сложность  $O(n)$





# ЗАДАЧА С

## «Истинные братья»

Time limit: 1 sec

Memory limit: 256 M

# Постановка задачи

- Дан цикл из  $n$  человек
- Нужно выбрать как можно больше пар, находящихся на расстоянии  $k$  друг от друга

# Решение (50 баллов)

- Будем жадно выбирать пары  $(0, k)$ ,  $(2k \bmod n, 3k \bmod n)$ ,  $(4k \bmod n, 5k \bmod n)$  и так далее, пока не зациклимся
- После зацикливания найдём человека, которого ещё не выбирали, начнем новый цикл из него
- Итоговая сложность  $O(n)$

# Решение (100 баллов)

- Найдём наименьшее  $s$ , что  $ks \bmod n = 0$
- Это  $s = n / \text{НОД}(k, n) = \text{НОК}(k, n) / k$
- Тогда каждый цикл содержит  $s$  человек или  $\lfloor s / 2 \rfloor$  пар ( $\lfloor x \rfloor$  – целая часть  $x$ )
- Всего будет  $n / s$  циклов
- Ответ:  $(n / s) \cdot \lfloor s / 2 \rfloor$
- Итоговая сложность  $O(\log n)$



# ЗАДАЧА E

## «Князь – Жадина»

Time limit: 1 sec

Memory limit: 256 M

# Постановка задачи

- Посчитать  $S = \sum_{d=1}^n \left\lfloor \frac{n}{d} \right\rfloor$

# Решение (30 баллов)

- Т.к. в первой группе тестов  $n \leq 10^5$ , то эту сумму можно просто посчитать

# Решение (100 баллов)

- Заметим, что при  $c \leq \sqrt{n}$ ,  $n / (c + 1) < d \leq n / c$   
 $\lfloor n / d \rfloor = c$ . Это позволяет выписать формулу

$$S = \sum_{d=1}^{\lfloor \frac{n}{d} \rfloor \geq \sqrt{n}} \left\lfloor \frac{n}{d} \right\rfloor + \sum_{c=1}^{c \leq \sqrt{n}} c \left( \left\lfloor \frac{n}{c} \right\rfloor - \left\lfloor \frac{n}{c+1} \right\rfloor \right)$$

- Итоговая сложность  $O(\sqrt{n})$





# ЗАДАЧА D

## «Кусторезка»

Time limit: 1 sec

Memory limit: 256 M

# Постановка задачи

- Дан куст, состоящий из веток К, С, КК, СС, КС, СК (КС – красное ребро соединено с корнем, синее ребро соединено с красным)
- Два игрока удаляют ребра своего цвета, “висячие” ребра так же удаляются
- КК и СС ветки можно заменить двумя К и С ветками соответственно, поэтому далее не будем их рассматривать



# Решение (30 баллов)

- Напишем рекурсивную функцию `solve(n1, n2, n3, n4, first)`, где `n1`, `n2`, `n3`, `n4` – количество К, С, КС и СК веток, `first` – игрок, который ходит первым.
- функция делает рекурсивные вызовы в состояния, куда можно сделать ход, если в одном из них выигрывает `first`, возвращает `first`, иначе `!first`



# Решение (30 баллов)

- Возможные переходы при  $first = 0$  (Петя)
- $solve(a_1, a_2, a_3 - 1, a_4, 1)$
- $solve(a_1, a_2 + 1, a_3, a_4 - 1, 1)$
- $solve(a_1 - 1, a_2, a_3, a_4, 1)$
- Итоговая сложность  $O(3^n)$ , где  $n$  – сумма  $n_i$
- Если не удалять КК и СС рёбра или делать рекурсивные вызовы в другом порядке, решение может получать меньше баллов

# Решение (70 баллов)

- В предыдущем решении делалось слишком много дублирующих вызовов
- Будем записывать результаты вызовов и
- Всего  $2 \cdot (n_1 + n_3) \cdot (n_2 + n_4) n_3 n_4$  возможных вызовов
- Итоговая сложность  $O(n^4)$  операций и  $O(n^4)$  память

# Решение (100 баллов)

- Пусть есть  $x$  веток  $K$  и  $y$  веток  $CK$ .
- Тогда при  $2x \leq y$  выиграет Вася, иначе выиграет Петя
- То есть,  $CK$  ветвь «стоит» половину  $C$  ветви, аналогично  $KC$  ветвь «стоит» половину  $K$  ветви
- Тогда рассмотрим  $s = n_1 - n_2 + 0.5 \cdot n_3 - 0.5 \cdot n_4$
- Если  $s > 0$ , выиграет Петя, иначе Вася

# Решение (100 баллов)

- Итоговая сложность  $O(1)$
- *Рассуждение про «стоимость» не совсем строгое. Более строгое доказательство можно прочитать в книжке Berlekamp E.R., Conway J.H., Guy R.K. – *Winning Ways for your Mathematical Plays**



**СПАСИБО!**

[cup.thumbtack.ru](http://cup.thumbtack.ru)

[vk.com/thumbtackcup](https://vk.com/thumbtackcup)