

# Учебная смена по информатике 2024/2025

30 октября 2024

## 0.1 А. Удача

### Решение на 70 баллов

Переберем все жетоны на отрезке от  $a$  до  $b$  и проверим каждый из них со всеми жетонами от  $c$  до  $d$ .

### Решение на 100 баллов

Переберем все жетоны на отрезке от  $a$  до  $b$  и проверим входит ли  $n - x$  ( $x$  - это текущий жетон) в отрезок от  $c$  до  $d$ .

## 0.2 В. Числа

**Решение на 30 баллов** Найдем количество различных пар 1 по формуле  $kol * (kol - 1) / 2$ .

**Решение на 100 баллов** Условие требуемое в задаче будет выполняться в двух случаях: при равных числах и если  $a[i] = 2, a[j] = 4$  Поэтому просто найдем количество различных пар каждого числа по формуле  $kol * (kol - 1) / 2$ . И просто добавим пары 2 и 4 по формуле  $kol2 * kol4$ .

## 0.3 С. Фестиваль

### Решение на 100 баллов

Можно найти ответ бинарным поиском. Чтобы проверить подходит ли нам время  $m$ , мы изначально отсортируем массив  $a$  и посмотрим сколько нам придется увеличить первый вид игрушки  $x = a[0] + m$ . На каждую игрушку мы тратим  $|a[i] - x|$ , если это больше чем  $m$ , можем сделать  $x = a[i] + m$ . Значит если мы поменяли  $x$  больше 2 раз, то  $m$  нам не подходит (так как у нас всего 3 резчика).

## 0.4 D. Опять AND

### Решение на 100 баллов

Мы можем заметить, что числа довольно маленькие, а значения AND будут до 63. Таким образом, мы можем подсчитать количество подпоследовательностей, у которых значение AND равно  $x$  для всех  $x$  от 0 до 63.

Мы можем сделать это с помощью динамического программирования. Обозначим  $dp[i][j]$  как количество подпоследовательностей, использующих первые  $i$  элементы, которые имеют общее AND-значение  $j$ . Переходы довольно простые. Мы можем перебрать все значения  $j$ , полученные ранее, и обновить значения соответственно: У нас есть три случая: Первый случай — когда мы не используем  $i$ -е значение. Здесь мы просто обновляем  $dp[i][j]$  следующим образом:  $dp[i][j] = dp[i][j] + dp[i-1][j]$ . Второй случай — когда мы используем  $i$ -е значение. Здесь мы обновляем  $dp[i][a[i] \& j]$  следующим образом:  $dp[i][a[i] \& j] = dp[i][a[i] \& j] + dp[i-1][j]$ . В третьем случае новая подпоследовательность начинается только с  $i$ -го элемента. Таким образом, мы обновляем  $dp[i][a[i]] = dp[i-1][a[i]] + 1$ .