

АИСД | ДЗ-4 | 1

Потякин Арсений

TODO:

1. Составить рекуррентное соотношение, которое описывает время работы алгоритма `shuffle`
 2. Вычислить верхнюю границу временной сложности заданного алгоритма
-

shuffle(A, l, r)

```
1  if l < r
2      thrd = (r - l) / 3
3      shuffle(A, l, l + thrd - 1)
4      shuffle(A, r - thrd + 1, r)
5      for i = 0 to thrd - 1
6          swap(A[l + thrd + i],
                A[l + 2 * thrd + i])
```

Task 1: (Рекуррентное соотношение)

Формула: $T_{shuffle}(n) = 2 \cdot T(\frac{n}{3}) + c \cdot (\frac{n}{3})$

Обоснование: пока цикл работает (условие выполняется), мы рекурсивно вызываем функцию `shuffle` $\iff 2 \cdot T(n)$, но аргумент уменьшается $\implies 2 \cdot T(\frac{n}{3})$, далее выполняется другая функция (константа с времени) ровно столько раз, чему равен уменьшенный аргумент $(\frac{n}{3})$

Task 2: (Верхняя граница временной сложности)

Мастер-теорема: $T(n) = a \cdot T(\frac{n}{b}) + O(n^k \cdot f(n))$, так как $k > \log_b a \implies 1 > \log_3 2$

$a = \text{const} = 2$; $b = \text{const} = 3$; $k = \text{const} = 1$; $f(n) = \text{const} = 1 \implies O(n)$

Ответ: $O(n)$