Количество итераций внешнего цикла

- outer будет принимать значения: $1, 2, 4, 8, ..., 2^k$, где k - макс. значение, при котором $2^k \leq n \rightarrow k = \lfloor \log_2 n \rfloor + 1 =>$

=> количество итераций = $O(\log_2 n)$

Количество итераций внутреннего цикла

- inner будет принимать значения: $2, 3, ..., n-1 =>$

=> количество итераций цикла = $(n-2)$

- Будет тратиться 1 операция на сравнение и 1 операция на ариф. действие ($x > y$ (inner))

- Если условие истинно то 2 операции на ариф действие и 1 на присваение ($y = y + $ outer/inner)

- Если условие ложное то одна операция на инкримент

- Тогда функцию временной сложности можно представить следующим образом:

$T(n) = (\log_2 n \cdot (n-1) \cdot 5) + (\log_2 n \cdot (n-1) \cdot 3) = 8(n-1) \cdot \log_2 n$

Пункт 2

$f(n) = \Theta(g(n))$ тогда и только тогда, когда

$\exists\ c_1, c_2, n_0 > 0\ \forall n \geq n_0 : 0 \leq c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$

В нашем случае $g(n) = \log_2 n \cdot n$

Т.к. $T(n) \simeq 8 \cdot n \cdot \log_2 n$ для больших n, то мы можем выбрать $c_1 = 1, c_2 = 10, n_0 = 1$

$=> T_n = \Theta(\log_2 n \cdot n) => f(n) = \log_2 n \cdot n$