Домашнее задание

1. Определим $f(n)$ как количество выводов «Hello, World!» следующей функцией (на входе n). Оцените асимптотику роста $f(n)$.

```
Function HelloWorld(n):
    if n > 2020 then
        HelloWorld(⌊n/4⌋);
        print("Hello, World!");
        HelloWorld(⌊n/4⌋);
        for i = 1 to 2020 do
            print("Hello, World!");
        end
        HelloWorld(⌊n/4⌋);
    else
        for i = 1 to n do
            print("Hello, World!");
        end
    end
end
```

2. На доске написан массив положительных целых чисел. За шаг можно брать любые два различных числа и вычитать из большего меньшее. Процесс остановится, когда на доске все числа будут одинаковыми.

1. Докажите, что процесс всегда останавливается.

2. Чему будут равны оставшиеся числа?

3. Предположим, удалось установить, что любое число можно возвести в квадрат за $O(n)$, где n – длина числа в двоичной записи. Докажите, что тогда любые два числа можно перемножать за $O(n)$, где n – длина максимального из чисел в двоичной записи.

4[ДПВ 1.33]. Постройте эффективный алгоритм для вычисления НОК и оцените его сложность. В данной задаче используется модель вычислений с атомарными битовыми операциями (т. е. время выполнения арифметических действий пропорционально длине чисел).

5. Дан массив $a_1, \ldots, a_n$. Найдите $\sum\limits_{i \neq j} a_i \cdot a_j$ за линейное от n количество арифметических операций.

6. Найдите Θ-асимптотику рекуррент, считая что $T(n) = \Theta(1)$ при малых n:

а) $T(n) = 36T(\frac{n}{6}) + n^2$; **б)** $T(n) = 3T(\frac{n}{3}) + n^2$; **в)** $T(n) = 4T(\frac{n}{2}) + \frac{n}{\log n}$.

7. На вход подаётся числовой массив A из n элементов. Требуется найти число инверсий в массиве, т. е. пар индексов (i, j), таких что $i < j$ и $a[i] > a[j]$.

Указание. Модифицируйте алгоритм сортировки слиянием.

8. Докажите, что если $T_1(n) = aT_1(\frac{n}{b}) + f(n)$, $T_2(n) = aT_2(\frac{n}{b}) + g(n)$ и $f(n) = \Theta(g(n))$, то $T_1(n) = \Theta(T_2(n))$.

В случае, когда в рекуррентных соотношениях возникают округления, задачу можно решать считая, что их нет, т. е. считать, что на вход $T(n)$ могут подаваться дробные параметры. Это облегчит задачу. Однако для того, чтобы лучше разобраться в теме, рекомендуем решать задачи с учётом округлений.

9. Найдите Θ-асимптотику рекуррентной последовательности $T(n)$, считая что $T(n)$ ограничено константой при достаточно малых n:

а) $T(n) = 3T(\lfloor n/4 \rfloor) + T(\lceil n/6 \rceil) + n$;

б) $T(n) = T(\lfloor \alpha n \rfloor) + T(\lfloor (1-\alpha)n \rfloor) + \Theta(n) \quad (0 < \alpha < 1)$;

в) $T(n) = T(\lfloor n/2 \rfloor) + 2 \cdot T(\lfloor n/4 \rfloor) + \Theta(n)$;

г) $T(n) = 27T(\frac{n}{3}) + \frac{n^3}{\log^2 n}$.

10. На вход подаются натуральные числа n, p, $n < p, p$ – простое. Предложите алгоритм, который за $O(n + \log p)$ арифметических операций вычисляет массив длины n (i пробегает значение от 1 до n):

а) $\texttt{invfac[}i\texttt{]} = (i!)^{-1} \pmod p$;

б)* $\texttt{inv[}i\texttt{]} = (i)^{-1} \pmod p$.

11*. Оцените трудоемкость рекурсивного алгоритма, разбивающего исходную задачу размера n на n задач размеров $\lceil \frac{n}{2} \rceil$ каждая, используя для этого $\Theta(n)$ операций.

1. Можно считать n степенью двойки.

2. Решите для произвольного n.