

# Алгоритмы

## Часть А

1. Чему равен порядок роста продолжительности работы следующего участка кода?

```
int sum = 0;
for (int i = 0; i < N; i++)
    for (int j = i+1; j < N; j++)
        for (int k = j+1; k < N; k++)
            sum++;
```

2. Расположите следующие функции в порядке увеличения скорости роста:

- |              |                   |                |
|--------------|-------------------|----------------|
| (a) $\log n$ | (e) $\log \log n$ | (i) $n \log n$ |
| (b) 1        | (f) $n\sqrt{n}$   | (j) $n^2$      |
| (c) $e^n$    | (g) $n!$          | (k) $2^n$      |
| (d) $n$      | (h) $n^n$         |                |

Отметьте все функции, чьё О-большое равно  $O(n^2)$

- |                     |                  |
|---------------------|------------------|
| • $1000n^2$         | • $n^3 - 100n^2$ |
| • $e^n$             | • $n \log n$     |
| • $4n^2 + 10n + 50$ | • $n^3/(1+n)$    |
| • $n^3 + 100n^2$    |                  |

3. Заполните таблицу:

Алгоритм	Сложность(в среднем)
Сортировка пузырьком	$O(n^2)$
Сортировка вставками	
Сортировка слиянием	
Быстрая сортировка	
Сортировка выбором	
Простейший алгоритм перемножения матриц nxn	
Простейший алгоритм сложения матриц nxn	

4. Заполните таблицу: Вычислительные сложности(в среднем) операций над структурами данных

	Доступ к элементу	Вставка	Удаление	Поиск
Массив				
Односвязный список(list)				
Двоичное дерево поиска				
Хеш-таблица				

5. Класс NP это:

- Класс задач, которые нельзя решить за полиномиальное время на детерминированной машине Тьюринга.
- Класс задач, которые можно решить за полиномиальное время на недетерминированной машине Тьюринга.
- Класс алгоритмов, которые нельзя выполнить за полиномиальное время на детерминированной машине Тьюринга.
- Класс алгоритмов, которые можно выполнить за полиномиальное время на недетерминированной машине Тьюринга.

## Часть В

1. Алиса и Боб любят игры и соревнования. И сейчас они готовы приступить к новой игре. Всего у них есть  $n$  плиток шоколада. По правилам игры они могут есть этот шоколад по очереди (первой начинает Алиса). Известно, что Алиса съедает  $a$  плиток шоколада за ход, а Боб —  $b$  плиток шоколада. Выйгрышает тот, кто съест последнюю плитку.

*Входные данные:*

Целые положительные  $n$ ,  $a$  и  $b$ .

*Выходные данные:*

"Alice" если победит Алиса, или "Bob" если победит Боб

*Пример:*

Вход	Выход
20 5 3	Alice
2 1 1	Bob

2. [Set1] Дано  $n$  чисел, причём известно, что среди этих чисел есть только одно уникальное. Все остальные числа повторяются парами. Найти это уникальное число.

*Входные данные:*

Целое положительное нечётное  $n \leq 2 * 10^9$

$n$  целых чисел.

*Выходные данные:*

Уникальное число

*Пример:*

Вход	Выход
3 1 8 1	8
11 5 4 7 4 8 78 15 5 78 8 15	7

3. **[Set2]** Дано  $n$  чисел и некоторое число  $A$ . Среди этих  $n$  чисел найти два таких, что их сумма равна  $A$ . Вернуть -1, если таких 2-х чисел нет.

*Входные данные:*

Целое положительное  $n \leq 2 * 10^9$  и целое положительное  $A$   
 $n$  целых чисел.

*Выходные данные:*

Индексы и значения искомой пары чисел или -1, если такой пары нет.

*Пример:*

Вход	Выход
3 1 8 1	8
11 5 4 7 4 8 78 15 5 78 8 15	7

4. Найти последнюю цифру  $n$ -го элемента последовательности Фибоначчи.

*Входные данные:*

Целое положительное  $n \leq 2 * 10^9$ .

*Выходные данные:*

Последняя цифра  $n$ -го элемента последовательности Фибоначчи

*Пример:*

Вход	Выход
20 5 3	Alice
2 1 1	Bob

5. Задача hexen в контексте NUMBER5.