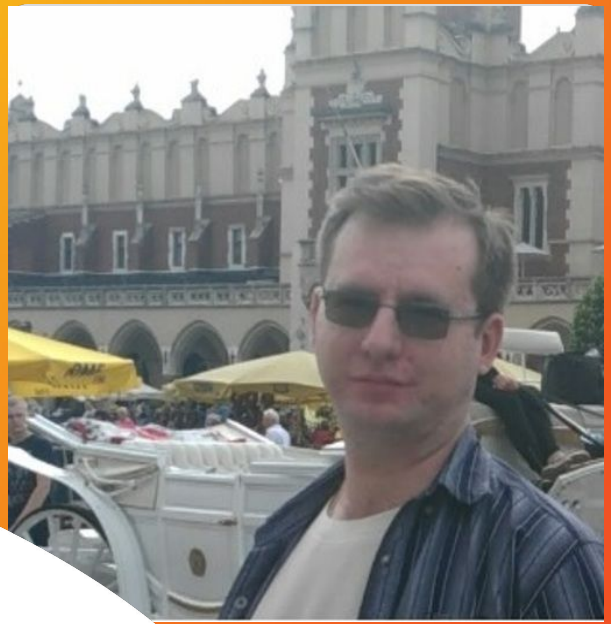
Algorithms 2

TEL-RAN
by Starta Institute

Big-O обозначение

ПРЕПОДАВАТЕЛЬ



TEL-RAN
by Starta Institute

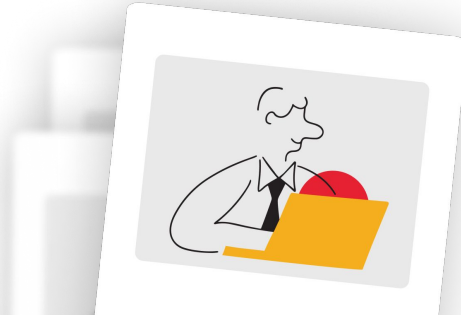
Артем Гордийчук

Full-stack software engineer

- Более 8 лет опыта работы
- Java, Spring, Hibernate, AWS, Oracle, PostgreSQL
- Проекты связанные с банковской, финансовой деятельность, e-commerce

artemsgor@gmail.com

www.linkedin.com/in/artem-g-48071a61



ВАЖНО:

- Камера должна быть включена на протяжении всего занятия.
- Если у Вас возник вопрос в процессе занятия, пожалуйста, поднимите руку и дождитесь, пока преподаватель закончит мысль и спросит Вас, также можно задать вопрос в чате или когда преподаватель скажет, что начался блок вопросов.
- Организационные вопросы по обучению решаются с кураторами, а не на тематических занятиях.
- Вести себя уважительно и этично по отношению к остальным участникам занятия.
- Во время занятия будут интерактивные задания, будьте готовы включить камеру или демонстрацию экрана по просьбе преподавателя.

TEL-RAN
by Starta Institute

ПЛАН ЗАНЯТИЯ

1. Повторение изученного
2. Вопросы по повторению
3. Разбор домашнего задания
4. Основной блок
5. Вопросы по основному блоку
6. Задание для закрепления
7. Практическая работа
8. Задание для закрепления
9. Оставшиеся вопросы

TEL-RAN
by Starta Institute

1 ПОВТОРЕНИЕ ИЗУЧЕННОГО

Повторение

TEL-RAN
by Starta Institute

- What is algorithm, introduction
- Characteristics of an algorithm
- Properties of an algorithm
- Type of an algorithm
- How to design an algorithm

TEL-RAN
by Starta Institute

2 ВОПРОСЫ ПО ПОВТОРЕНИЮ

TEL-RAN
by Starta Institute

3 РАЗБОР ДОМАШНЕГО ЗАДАНИЯ

Алгоритм сложения 3 чисел

TEL-RAN
by Starta Institute

1. Написать псевдокод для алгоритма: сложить три числа и вывести сумму

2. Реализовать алгоритм в коде

Алгоритм сложения 3 чисел

TEL-RAN
by Starta Institute

Шаг 1: Выполнение предварительных условий

Шаг 2: Разработка алгоритма

- Алгоритм сложения 3 чисел и вывода их суммы:
- Получить от пользователя 3 целочисленные переменные num1, num2 и num3.
- Возьмите три добавляемых числа в качестве входных данных для переменных num1, num2 и num3 соответственно.
- Объявите целочисленную переменную sum для хранения результирующей суммы трех чисел.
- Добавьте 3 числа и сохраните результат в переменной sum.
- Вывести значение переменной sum

Шаг 3: Проверка алгоритма путем его реализации.

Алгоритм сложения 3 чисел

TEL-RAN
by Starta Institute

Реализация на псевдокоде

```
START
READ number input: num1,num2,num3
  declare sum
  sum = num1+num2+num3
  print sum
END
```

Алгоритм сложения 3 чисел

TEL-RAN
by Starta Institute

Реализация на Java

```java
public static void main(String[] args) {
    int sum;
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the 1-st number: ");
    int num1 = sc.nextInt();
    System.out.println("Enter the 2-nd number: ");
    int num2 = sc.nextInt();
    System.out.println("Enter the 3-rd number: ");
    int num3 = sc.nextInt();
    sum = num1 + num2 + num3;
    System.out.println("Sum of the 3 numbers is = " + sum);
}
```

Реализация на Java Script

```javascript
function threeNumbersSum() {
    let sum = 0;
    let num1 = parseInt(prompt("Enter the 1st number: "));
    console.log(` ${num1}`);
    let num2 = parseInt(prompt("Enter the 2nd number: "));
    console.log(` ${num2}`);
    let num3 = parseInt(prompt("Enter the 3rd number: "));
    console.log(` ${num3}`);
    sum = num1 + num2 + num3;
    console.log(`Sum of the 3 numbers is = ${sum}`);
}
```

Введение

TEL-RAN
by Starta Institute

Big-O обозначение

- Асимптотический анализ
- Порядок роста
- Константный - O(1)
- Логарифмический- O(log n)
- Линейный - O(n)
- Линейно-логарифмический - O(n log n)
- Квадратичный - O(n 2)
- Факториальный - O(н!)
- Лучший, средний и худший случай
- Что мы измеряем?

TEL-RAN
by Starta Institute

4 ОСНОВНОЙ БЛОК

Асимптотический анализ

TEL-RAN
by Starta Institute

- В асимптотическом анализе мы оцениваем производительность алгоритма с точки зрения размера входных данных, другими словами мы подразумеваем анализ времени, которое потребуется для обработки очень большого набора данных.
- Имея два алгоритма для задачи, как мы узнаем, какой из них лучше?
- Мы рассчитываем, как время (time complexity) или пространство (space complexity), занимаемое алгоритмом, увеличивается с размером входных данных.

Асимптотический анализ

TEL-RAN
by Starta Institute

- Рассмотрим задачу поиска (поиск заданного элемента) в отсортированном массиве.
  - линейный поиск (порядок роста — линейный O(n))
  - бинарный поиск (порядок роста — логарифмический O(log n))

Компьютер A – константное время 0.2 сек

Компьютер B – константное время 1000 сек

Асимптотический анализ

TEL-RAN
by Starta Institute

Время выполнения линейного поиска в секундах для A : 0,2 * n

Время выполнения двоичного поиска в секундах для B : 1000*log(n)

n	time on A	time on B
10	2 sec	~ 1 h
100	20 sec	~ 1.8 h
10^6	~ 55.5 h	~ 5.5 h
10^9	~ 6.3 years	~ 8.3 h

Причина в том, что порядок роста бинарного поиска по размеру входных данных является логарифмическим, а порядок роста линейного поиска — линейным.

Порядок роста

TEL-RAN
by Starta Institute

Порядок роста описывает то, как сложность алгоритма растет с увеличением размера входных данных. Порядок роста представляется в виде O-нотации:

O(f(x)), где f(x) — формула, выражающая сложность алгоритма.

O(1) – **Константный**

Порядок роста O(1) означает, что вычислительная сложность алгоритма не зависит от размера входных данных.

```
public int getSize(int[] arr) {
    return arr.length;
}
```

Экспресс-опрос

TEL-RAN
by Starta Institute

- **Вопрос 1.**

  Как узнать, какой алгоритм из двух лучше?

- **Вопрос 2.**

  Как вы поняли, что описывает порядок роста?

- **Вопрос 3.**

  Как обозначается сложность алгоритма?

Порядок роста

TEL-RAN
by Starta Institute

O(n) **– линейный**

Порядок роста O(n) означает, что сложность алгоритма линейно растет с увеличением входного массива.

Если линейный алгоритм обрабатывает один элемент 1 секунду, то сто элементов обработается за сто секунд.

```
public long getSum(int[] arr) {
      long sum = 0;
      for (int i = 0; i < arr.length; i++) {
          sum += i; }
     return sum;
 }
```

Порядок роста

TEL-RAN
by Starta Institute

O(log n) **– логарифмический**

Порядок роста O(log n) означает, что время выполнения алгоритма растет логарифмически с увеличением размера входного массива.

Большинство алгоритмов, работающих по принципу «деления пополам», имеют логарифмическую сложность.

Пример:

Алгоритм двоичного поиска

Порядок роста

TEL-RAN
by Starta Institute

O(n log n) **– линейно-логарифмический**

Некоторые алгоритмы типа «разделяй и властвуй» попадают в эту категорию.

Пример:

Сортировка слиянием и быстрая сортировка

Порядок роста

TEL-RAN
by Starta Institute

O(n2) **– квадратичный**

Время работы алгоритма O(n^2) зависит от квадрата размера входного массива.

Квадратичная сложность — повод задуматься и переписать алгоритм.

Массив из 100 элементов потребует 1 0000 операций

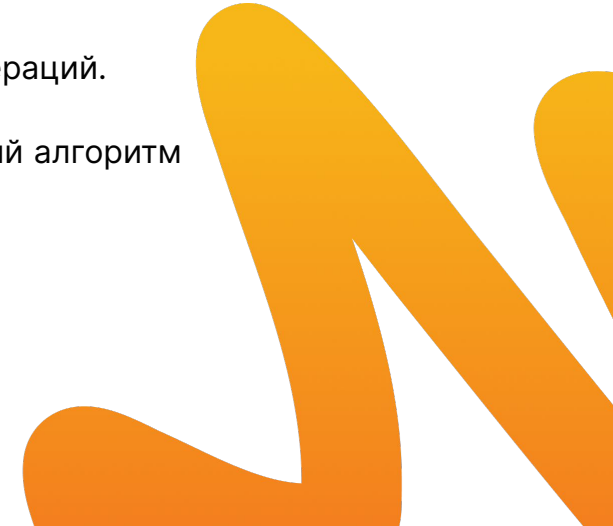
Массив из миллиона элементов потребует 1 000 000 000 000 (триллион) операций.

Если одна операция занимает миллисекунду для выполнения, квадратичный алгоритм будет обрабатывать миллион элементов 32 года.

Даже если он будет в сто раз быстрее, работа займет 84 дня.

Пример:

Алгоритм пузырьковая сортировка



Порядок роста

TEL-RAN
by Starta Institute

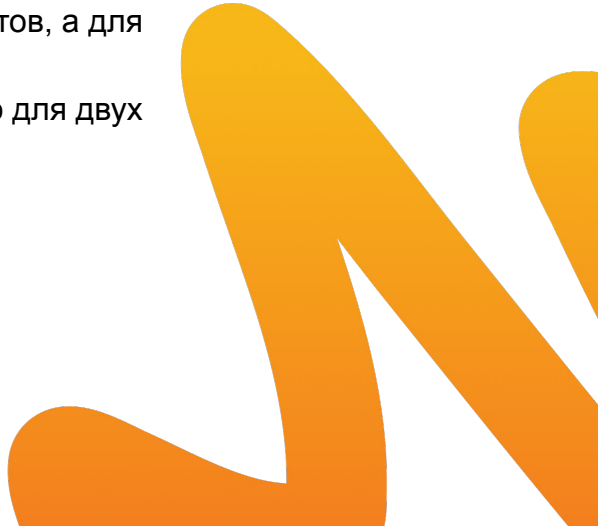
O(n!) **– факториальный**

Очень медленный алгоритм.

Пример:

Задача коммивояжёра
Найти оптимальный маршрут - для 15 городов существует 43 миллиарда маршрутов, а для 18 городов уже 177 триллионов.
Если бы существовало устройство, находящее решение для 30 городов за час, то для двух дополнительных городов требуется в тысячу раз больше времени;
то есть, более чем 40 суток!

Такая задача из класса NP - задача с ответом «да» или «нет»



Наилучший, средний и наихудший случаи

TEL-RAN
by Starta Institute

Обычно имеется в виду наихудший случай, за исключением тех случаев, когда наихудший и средний сильно отличаются. (оценка сверху)

Например: array.add()

В среднем имеет порядок роста O(1), но иногда может иметь O(n).

В этом случае мы будем указывать, что алгоритм работает в среднем за константное время, и объяснять случаи, когда сложность возрастает.

Самое важное здесь то, что O(n) означает, что алгоритм потребует не более n шагов!

Что мы в итоге измеряем и всегда ли это работает?

TEL-RAN
by Starta Institute

При измерении сложности алгоритмов и структур данных мы обычно говорим о двух вещах: количество операций, требуемых для завершения работы (вычислительная сложность), и объем ресурсов, в частности, памяти, который необходим алгоритму (пространственная сложность).

Алгоритм, который выполняется в десять раз быстрее, но использует в десять раз больше места, может вполне подходить для серверной машины с большим объемом памяти. Но на встроенных системах, где количество памяти ограничено, такой алгоритм использовать нельзя.

Что мы в итоге измеряем и всегда ли это работает?

TEL-RAN
by Starta Institute

Асимптотический анализ не идеален, но это лучший доступный способ анализа алгоритмов.

Два алгоритма сортировки, которые занимают на машине 1000 nLogn и 2 nLogn.

Мы не можем судить, какой из них лучше, поскольку мы игнорируем константы.

Таким образом, вы можете в конечном итоге выбрать алгоритм, который асимптотически медленнее, но быстрее для вашего программного обеспечения.

Важно:

TEL-RAN
by Starta Institute

- Скорость алгоритма измеряется не в секундах, а в приросте количества операций.
- Насколько быстро возрастает время работы алгоритма в зависимости от увеличения объема входящих данных.
- Время работы алгоритма выражается при помощи нотации большого «O».
- Алгоритм со скоростью O(log n) быстрее, чем со скоростью O(n), но он становится намного быстрее по мере увеличения списка элементов.

Экспресс-опрос

TEL-RAN
by Starta Institute

- **Вопрос 1.**

  Что обозначает Big O?

- **Вопрос 2.**

  Почему игнорируются константы?

Понимание сложности времени

TEL-RAN
by Starta Institute

Я кому то из вас дал ручку!

У меня есть несколько способов найти мою ручку.

Понимание сложности времени

TEL-RAN
by Starta Institute

Я кому то из вас дал ручку!

У меня есть несколько способов найти мою ручку.

1. Я иду и спрашиваю у первого человека в классе, есть ли у него ручка. А дальше мне самому лениво идти к следующему и я просто начинаю спрашивать этого человека о других людях в классе, есть ли у них эта ручка? и так далее.

Понимание сложности времени

TEL-RAN
by Starta Institute

Я кому то из вас дал ручку!

У меня есть несколько способов найти мою ручку.

1. ~~Я иду и спрашиваю у первого человека в классе, есть ли у него ручка. А дальше мне самому лениво идти к следующему и я просто начинаю спрашивать этого человека о других людях в классе, есть ли у них эта ручка? и так далее.~~
2. Я иду и спрашиваю каждого ученика по отдельности.

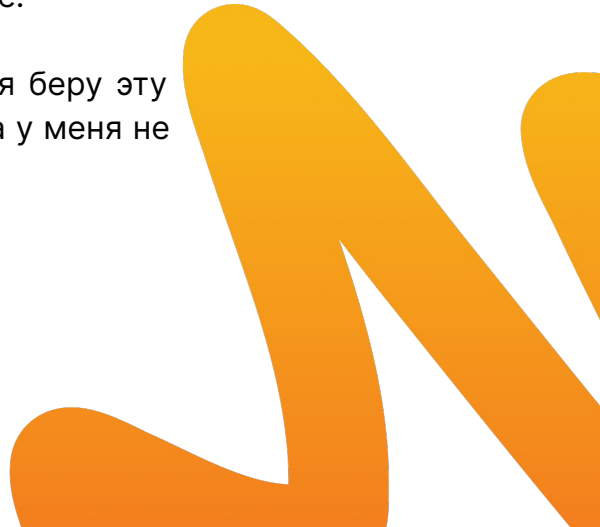
Понимание сложности времени

TEL-RAN
by Starta Institute

Я кому то из вас дал ручку!

У меня есть несколько способов найти мою ручку.

1. ~~Я иду и спрашиваю у первого человека в классе, есть ли у него ручка. А дальше мне самому лениво идти к следующему и я просто начинаю спрашивать этого человека о других людях в классе, есть ли у них эта ручка? и так далее.~~
2. ~~Я иду и спрашиваю каждого ученика по отдельности.~~
3. Я делю класс на две группы и спрашиваю: «Где моя ручка?» Затем я беру эту половину, делю ее на две части и спрашиваю снова, и так далее. Пока у меня не останется один ученик, у которого есть моя ручка.



Понимание сложности времени

TEL-RAN
by Starta Institute

Я кому то из вас дал ручку!

У меня есть несколько способов найти мою ручку.

1. ~~Я иду и спрашиваю у первого человека в классе, есть ли у него ручка. А дальше мне самому лениво идти к следующему и я просто начинаю спрашивать этого человека о других людях в классе, есть ли у них эта ручка? и так далее.~~
2. ~~Я иду и спрашиваю каждого ученика по отдельности.~~
3. ~~Я делю класс на две группы и спрашиваю: «Где моя ручка?» Затем я беру эту половину, делю ее на две части и спрашиваю снова, и так далее. Пока у меня не останется один ученик, у которого есть моя ручка.~~
4. Точно, я вспомнил кому я ее дал.

Пример 1:

TEL-RAN
by Starta Institute

Реализация на Java

```java
public static void main(String[] args){
    System.out.print("Hello World");
}
```

Реализация на Java Script

```javascript
function sayHello() {
    console.log("Hello World");
}
```

Сложность времени: в приведенном выше коде «Hello World» печатается на экране только один раз.
временная сложность постоянна: O(1)

Пример 2:

TEL-RAN
by Starta Institute

Реализация на Java

```java
public static void main(String[] args){
    int n = 8;
    for (int i = 1; i <= n; i++) {
        System.out.println("Hello World !!!");
    }
}
```

Реализация на Java Script

```javascript
function sayManyHello() {
    let n = 8;
    for (let i = 1; i <= n; i++) {
        console.log("Hello World !!!");
    }
}
```

Сложность времени: в приведенном выше коде «Hello World !!!» выводится на экран только n раз , временная сложность является линейной: O(n).

TEL-RAN
by Starta Institute

5 ВОПРОСЫ ПО ОСНОВНОМУ БЛОКУ

TEL-RAN
by Starta Institute

6 ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

ЗАДАНИЕ

TEL-RAN
by Starta Institute

Задание для закрепления:

Предложите свой пример для объяснения сложности времени (аналогично примеру с ручкой).

O(1)

O(n)

O(n^2)

O(log n)

TEL-RAN
by Starta Institute

7 ПРАКТИЧЕСКАЯ РАБОТА

Практическое задание 1

TEL-RAN
by Starta Institute

Задача: Найдите сумму двух чисел.

Научимся рассчитывать сложность алгоритма.

Определим константы:

- 1 единица времени на арифметические и логические операции
- 1 единица времени для операторов присваивания и возврата

Реализация задания 1

TEL-RAN
by Starta Institute

```java
public static int sum(int one, int two) {
    int result = one + two; // cost = 1 + 1
    return result; // cost = 1
}
```

```javascript
function sumTwoNumbers(one, two) {
    return one + two;
}
```

Реализация задания 1

TEL-RAN
by Starta Institute

```java
public static int sum(int one, int two) {
    int result = one + two; // cost = 1 + 1
    return result; // cost = 1
}
```

```
(Tsum) = 2 + 1 = 3.
O(3) -> 3 is constant
Time complexity = O(2) = O(1)
```

```javascript
function sumTwoNumbers(one, two) {
    return one + two;
}
```

?

Реализация задания 1

TEL-RAN
by Starta Institute

```java
public static int sum(int one, int two) {
    int result = one + two; // cost = 1 + 1
    return result; // cost = 1
}
```

(Tsum) = 2+1 = 3
O(3) -> 3 is const
Time comp O(3) = O(1)

```javascript
function sumTwoNumbers(one, two) {
    return one + two;
}
```

(Tsum) = 1+1 = 2
O(2) -> 2 is const
Time comp O(2) = O(1)

Пример игнорирования констант:

```
4n^3 = O(n^3)
n^2+n = O(n^2)
log2(n), log10(n) = O(log n)
```

Практическое задание 2

TEL-RAN
by Starta Institute

Задача:

1) Найти сумму всех элементов массива
    - Реализовать на Java или JS
2) Рассчитать сложность алгоритма.

Реализация задания 2

TEL-RAN
by Starta Institute

Реализация на Java

```java
public static int listSum(int[] array) {
    int result = 0;                              // cost = 1 times = 1
    for(int i = 0; i < array.lenght; i++) {      // cost = 3 times = n + 1
      result = result + array[i];                // cost = 2 times = n
    }
return result;                                   // cost = 1 times = 1
}
```

Реализация на Java Script

```javascript
function listSum(array) {
    let result = 0;                              // cost = 1 times = 1
    for(let i = 0; i < array.length; i++) {      // cost = 3 times = n + 1
        result = result + array[i];              // cost = 2 times = n
    }
    return result;                               // cost = 1 times = 1
}
```

TEL-RAN
by Starta Institute

8 ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

Задание 1

TEL-RAN
by Starta Institute

```java
public static void fork(int n) {
    if(n < 5) {
        System.out.println("number < 5");
    }
    else {
        for(int i = 0; i<n; i++) {
        System.out.println(i + " ");
        }
    }
}
```

JAVA

```javascript
function fork(n) {
    if(n < 5) {
        console.log("number < 5")
    } else {
        for(let i = 0; i<n; i++) {
            console.log(i + " ");
        }
    }
}
```

JAVAScript

Определить наилучший случай(best case) и наихудший (worst case) - объяснить: почему

Задание 2

TEL-RAN
by Starta Institute

```java
public static void bubleFor(int n) {
    for (int i = 0; i <= n / 3; i++) {
        for (int j = 1; j <= n; j = j + 4) {
            System.out.println("I study Big O notation");
        }
    }
}
```

JAVA

```javascript
function doubleFor(n) {
    for (let i = 0; i <= n / 3; i++){
        for (let j = 1; j <= n; j = j + 4){
            console.log("I study Big O notation");
        }
    }
}
```

JAVAScript

TEL-RAN
by Starta Institute

8 ОСТАВШИЕСЯ ВОПРОСЫ

Домашнее задание

TEL-RAN
by Starta Institute

1. Познакомиться с «Задачей коммивояжёра» (ссылка на следующем слайде)
2. Практические задачи: какова временная сложность?

task #1
```
START
READ number n
IF n == 1 THEN return
FOR i = 1, i <= n, i + 1
        FOR j = 1; j <= n, j + 1
                print "*"
                BREAK
END
```

task #2
```
START
READ number n
numbers i = 0, j = 0, a = 0
FOR i = n/2, i <= n; i + 1
        FOR j = 2, j <= n, j * 2
                a = a + n / 2
END
```

task #3
```
START
READ number n
number a = 0
FOR i = 0, i < n, i + 1
        FOR j = n, j > i, j - 1
                a = a + i + j
END
```

task #4
```
START
READ number n
numbers a = 0, i = n
WHILE i > 0
        a = a + i
        i = i / 2
END
```

Полезные ссылки

TEL-RAN
by Starta Institute

- Задача коммивояжёра — Википедия

ЗАКЛЮЧЕНИЕ

TEL-RAN
by Starta Institute