



NATIONAL RESEARCH  
UNIVERSITY

# АЛГОРИТМЫ И СТРУКТУРЫ ДАННЫХ

(сентябрь–декабрь, модули 1 и 2)

## ВВЕДЕНИЕ

Рамон Антонио Родригес Залепинос  
[arodriges@hse.ru](mailto:arodriges@hse.ru)

# Родригес Залепинос Рамон Антонио

Доцент: [Факультет компьютерных наук / Департамент программной инженерии](#)

*Начал работать в НИУ ВШЭ в 2015 году.*

*Научно-педагогический стаж: 10 лет.*

[Домашняя страница](#) **Преподавание** [Публикации и исследования](#) [Проекты](#) [Группа Гео](#) [В новостях](#)

## Рамон Антонио

**S914**

[arodrigues@hse.ru](mailto:arodrigues@hse.ru)

**Предварительно  
договаривайтесь о  
встрече**



# Курс состоит из двух частей

	Модули 1–2
Лекции	Р.А. Родригес Залепинос
Семинары	Н.А. Терлыч А.Д. Рахмановский А.М. Бузулуков
	Структуры данных

Модули 3–4
М.В. Ульянов (лекции)
?
Алгоритмы

# Оценивание

**HW 60%**

**CW 20%**

**EX 20%**

HW – средняя оценка за домашние работы

CW – оценка за контрольную работу

EX – оценка за экзамен

IT – инициативная тема

## Веса заданий

- домашние задания **имеют одинаковый вес** (каждое оценивается 0..10)

## Округление оценок

- по стандартным правилам, в пользу студента: **5.5 -> 6**

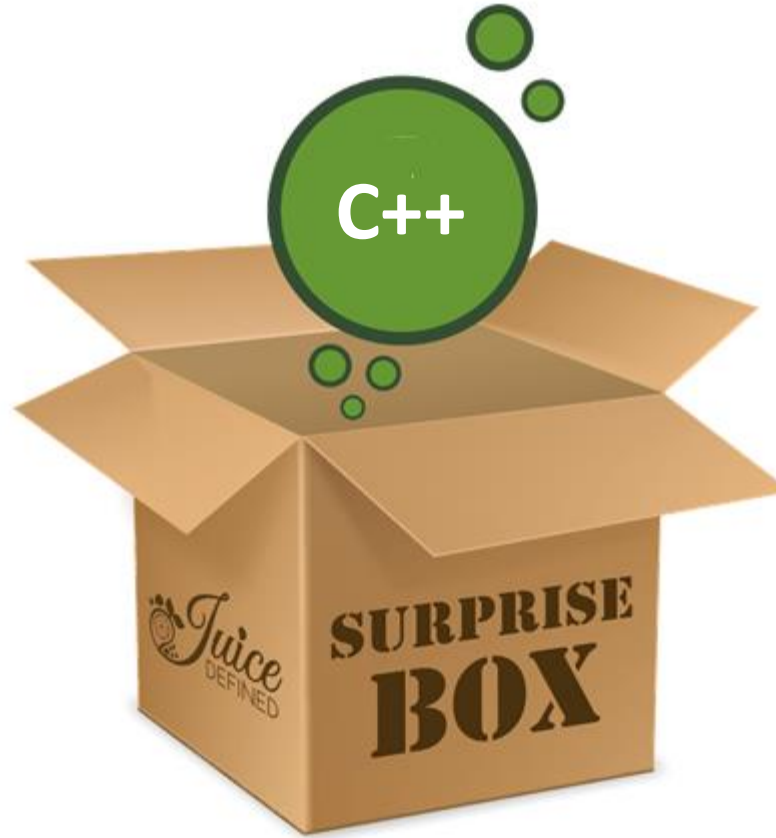
## Автомат

- можно получить, если  **$\text{округл}[(HW + CW)/80 * 10; 0] \geq 8$**

# Бонус

## Причины:

- Решение руководства департамента



## Это сложно, но

- возможно
- интересно
- полезно

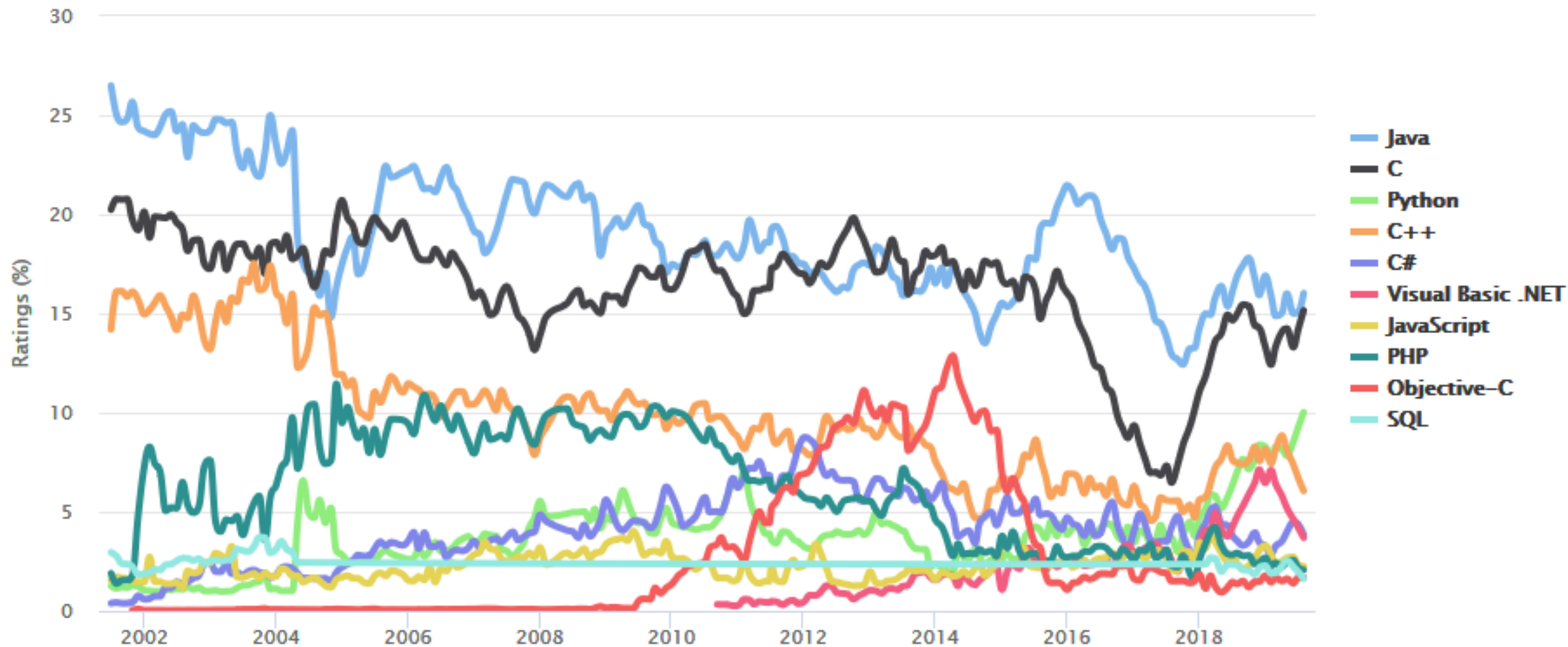
C#



C++

плавный переход

# Востребованность C++



# Источники информации ≈ литература

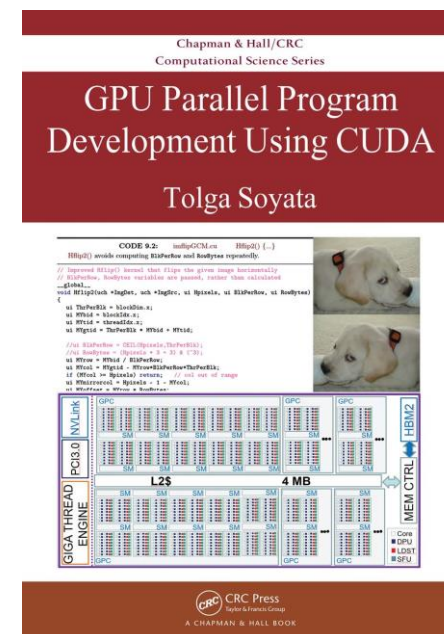
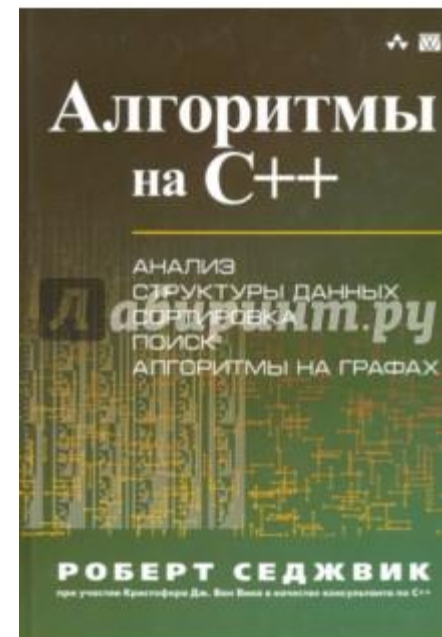
- Лекции
- Семинары
- Книги
- Статьи
- Конференции
- ...





# Литература

## C++, структуры данных, современные CPU




- Б. Страуструп: Язык программирования C++ (стандарт C++11). Краткий курс
- Кормен, Лейзерсон, Ривест: Алгоритмы. Построение и анализ
- Ахо, Ульман, Хопкрофт: Структуры данных и алгоритмы
- Роберт Седжвик: Алгоритмы на C++
- Tolga Soyata, GPU Parallel Program Development Using CUDA
- ☐ Печатные экз.: в НИУ ВШЭ есть хорошая библиотека
- ☐ Электронные – поисковые системы



21-22 сентября 2020 г.

Два исключительно наполненных суперкомпьютерными событиями дня: научные секции, выставка, семинары, мастер-классы и многое другое.

[О конгрессе](#) [О конференции »](#) [Важные даты](#) [Программа »](#) [Участникам](#) [Авторам »](#) [Регистрация](#) 

[Главная](#) » [Участие в конференции](#)

## Участие в конференции

### Время и место проведения конференции

Конференция "Суперкомпьютерные дни в России" пройдет 21-22 сентября 2020 г. в онлайн-формате.

### Участие в конференции

Для участия в конференции необходимо [зарегистрироваться и оплатить оргвзнос](#). Стоимость участия в конференции - 150 р.

### Докладчикам

Доклады на конференцию представляются в виде устного выступления или стендового доклада (плаката/постера). Материалы на конференцию принимаются в одном из трех видов: полная статья, короткая статья, аннотация постера. Все поступающие материалы проходят экспертный отбор программным комитетом. Все материалы, одобренные программным комитетом, будут опубликованы в сборнике трудов конференции.

### Участникам выставки

На выставке представляются новейшие российские и зарубежные программные и аппаратные решения и технологии для высокопроизводительных вычислений.

### Организаторам и участникам семинаров

В рамках конференции проходит множество семинаров и мастер-классов. Приглашаем посетить их, а также организовать свое мероприятие на конференции!



Спонсоры

Платиновые



<http://russianscdays.org/attending>



Золотые



**МСА: обучение  
бесплатное,  
прием заявок до  
12.09.2020**

[Главная](#) » [Программа](#)

## Реализация глубоких нейросетей на высокопроизводительных кластерах

### Руководители научной школы:

Буряк Дмитрий Юрьевич, к.ф.-м.н., старший инженер-исследователь, Московская исследовательская лаборатория LG Electronics

### Аннотация:

В рамках научной школы будут рассмотрены базовые элементы теории глубоких нейронных сетей, практические приемы их построения и обучения, вопросы параллельной реализации нейросетевых алгоритмов. Слушатели научной школы получат возможность практической работы с нейросетевыми фреймворками TensorFlow и другими нейросетевыми пакетами, реализованными в рамках платформы PowerAI на высокопроизводительном кластере Polus, построенном на базе процессоров Power8 с графическими ускорителями P100.

[Главная](#) » [Программа](#)

## Применение платформы Python для высокопроизводительных вычислений

### Руководители научной школы:

Русол Андрей Владимирович, к.т.н., с.н.с. ГЕОХИ РАН

## Высокопроизводительные вычисления на кластерах с использованием графических ускорителей NVIDIA



### Руководители научной школы:

Афанасьев Илья, НИВЦ МГУ им. М.В.Ломоносова

Жуков Константин Андреевич, к.ф.м.н., факультет ВМК МГУ им. М.В.Ломоносова

<https://academy.hpc-russia.ru/>

# VLDB: впервые онлайн и бесплатное участие

**VLDB2020 TOKYO**

<https://vldb2020.org/>

Значимость VLDB, примеры:

- **Новая гибридная СУБД HadoopDB**
  - Abouzeid, A. et al., HadoopDB: An architectural hybrid of mapreduce and DBMS technologies for analytical workloads, 2009
- **Новая колоночная СУБД C-Store**
  - Stonebraker, M., et al., C-Store: A column-oriented DBMS, 2005
  - Коммерциализация в HP Vertica (E-Bay, кластер из 100 узлов, 7.5PB, 2007 г.)
- **Dremel, Hive, H-Store, ...**





# Значимость SIGMOD, примеры:

- Предложены В-деревья
  - Rudolf Bayer, Edward M. McCreight, Organization and maintenance of large ordered indices
- Предложены R-деревья
  - Guttman, A., R-trees: A dynamic index structure for spatial searching
- Предложена технология RAID
  - Patterson, D.A., Gibson, G., Katz, R.H., A case for Redundant Arrays of Inexpensive Disks (RAID)
- Новая растровая СУБД **SciDB**
  - Rogers, J., et al. Overview of SciDB: Large scale array storage, processing and analysis

**SIGMOD** –  
одна из самых  
значимых в мире  
конференций по

- Управлению  
Данными
- Бадам Данных

SIGMOD /  
AMSPXDS  
TER  
DAM 2X19



# Рамон Антонио – о себе



## VLDB 2018

Единственный устный доклад на VLDB от РФ за последние 10 лет (2018–2009) и первый от НИУ ВШЭ

## SIGMOD 2019

Седьмая статья на SIGMOD от РФ за все годы SIGMOD (1975–2019, 45 лет) и первая от НИУ ВШЭ

## VLDB 2020

Вторая по счету статья на VLDB от Российской Федерации за последние 10 лет (2010–2020) и вторая от НИУ ВШЭ



**VLDB & SIGMOD – самые значимые в мире конференции по**

- Базам данных
- Большим данным
- Распределенным системам
- Управлению данными



# Структура модуля 1

Модуль № 1	№	Дата	Тема лекции	№	Домашние задания
	1	08 сен	Введение	A1	Установить Visual Studio с поддержкой C# и C++ на все рабочие машины (стационарные, переносные, т.п.). Найти книги по структурам данных. Посетить конференции.
	2	15 сен	Асимптотика	A2	Тестовая задача (C#): знакомство с процессом
	3	22 сен	Базовые СД	1a	Задание 1a на C#
	4	29 сен	Базовые СД 2	1b	Задание 1b на C++, перевод 1a на C++
	5	06 окт	Bitmaps	2a	Задание 2a на C#
	6	13 окт	Хэш таблицы	2b	Задание 2b на C++, перевод 2a на C++

**КОНТРОЛЬНАЯ РАБОТА** (последняя неделя модуля № 1, **на семинаре**):  
 темы модуля 1; форма проведения зависит от карантинных мер  
 Возможна замена на КДЗ либо онлайн формат

**Итого 4+2 домашних заданий в модуле 1**  
 На каждое ДЗ – 1 неделя

**Возможны модификации, т.к. осталось всего 14 вторников**

# Требования, предъявляемые к студентам на лекции

## Если я хочу посетить лекцию

- Пожалуйста, посещайте лекции по расписанию, со своим потоком

## Если я опоздал(а)

- Пожалуйста, стараясь не привлекать к себе внимание, присоединитесь к Zoom

## Если я нахожусь на лекции

- Слушайте внимательно!

## Если я хочу задать вопрос/высказаться во время лекции

- Пожалуйста, поднимите руку и дождитесь момента, когда Вам дадут слово; не выкрикивайте

## Идентификация

Используйте свое имя, фамилию и группу:

ИМЯ ФАМИЛИЯ ГРУППА

напр.: Иван Иванов БПИ193

## Санкции

При неверной идентификации либо нарушении порядка – перевод в зал ожидания



# Сайт курса

<http://edu.gis.land/ds2020>

неотъемлемая часть ПУД для 1-2 модулей

# Типы и структуры данных

- **тип данных** переменной обозначает множество значений, которые может принимать эта переменная
- **абстрактный тип данных (АТД)** — это математическая модель плюс различные операторы, определенные в рамках этой модели
- для представления АТД используются **структуры данных**, которые представляют собой набор переменных, возможно, различных типов данных, объединенных определенным образом

## Notes

- АТД не имеют асимптотической сложности выполнения операций (операторов)
- АТД не зависят от языков программирования
- Структуры данных = реализация АТД, необязательно на конкретном языке (разная реализация – разная сложность выполнения операций)

# Алгоритм

**Алгоритм** (algorithm) —это формально описанная вычислительная процедура, получающая **исходные данные** (input), называемые также входом алгоритма или его аргументом, и выдающая **результат** вычислений на выход (output).

Мы вспомнили алгоритмы сейчас, а не во 2ой части курса, т.к. операторы в структурах данных **реализуют алгоритмы**, разная реализация имеет разную асимптотическую **сложность**.

**Вход и выход:** лирическое отступление (курсовые работы)

Алгоритмы строятся для решения тех или иных **вычислительных задач** (computational problems).



# Традиционный вопрос

Какое «железо» (требование к аппаратуре) и какой язык выбрать для реализации системы?

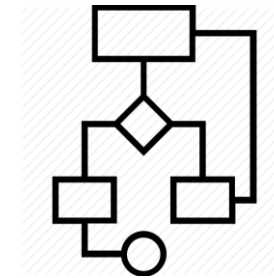
Цель – высокая производительность



Компьютер



Язык



Алгоритм

# Пример: задача сортировки

**Вход:** Последовательность  $n$  чисел  $\langle a_1, a_2, \dots, a_n \rangle$ .

**Выход:** Перестановка  $\langle a'_1, a'_2, \dots, a'_n \rangle$  исходной последовательности, для которой  $a'_1 \leq a'_2 \leq \dots \leq a'_n$ .

Алгоритм считают **правильным** (correct), если на любом допустимом (для данной задачи) входе он заканчивает работу и выдает результат, удовлетворяющий требованиям задачи. В этом случае говорят, что алгоритм **решает** (solves) данную вычислительную задачу.

# <https://www.top500.org/lists/top500>

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	<b>Supercomputer Fugaku</b> - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,299,072	415,530.0	513,854.7	28,335
2	<b>Summit</b> - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM DOE/SC/Oak Ridge National Laboratory United States	2,414,592	148,600.0	200,794.9	10,096
3	<b>Sierra</b> - IBM Power System AC922, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM / NVIDIA / Mellanox DOE/NNSA/LLNL United States	1,572,480	94,640.0	125,712.0	7,438
4	<b>Sunway TaihuLight</b> - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway, NRCPC National Supercomputing Center in Wuxi China	10,649,600	93,014.6	125,435.9	15,371
5	<b>Tianhe-2A</b> - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000, NUDT National Super Computer Center in Guangzhou China	4,981,760	61,444.5	100,678.7	18,482



# Summit's Home: The U.S. Department of Energy's Oak Ridge National Laboratory



**Oak Ridge National Laboratory  
is the largest US Department of  
Energy (DOE) open science  
laboratory**



# System Overview

## System Performance

- Peak performance of 200 petaflops for modeling & simulation
- Peak of 3.3 ExaOps for data analytics and artificial intelligence

## Each node has

- 2 IBM POWER9 processors
- 6 NVIDIA Tesla V100 GPUs
- 608 GB of fast memory
- 1.6 TB of NVMe memory

## The system includes

- 4608 nodes
- Dual-rail Mellanox EDR InfiniBand network
- 250 PB IBM Spectrum Scale file system transferring data at 2.5 TB/s





## О пользе быстрых алгоритмов

- Пусть сортировка вставками написана экономно, на низкоуровневом языке и требует  $2n^2$  операций
- Пусть сортировка слиянием написана неэкономно и на ??? и поэтому требует  $50n \log n$  операций
- Сортируем 1 терабайт (около 274,877,906,944 4-байтовых чисел)

## О пользе быстрых алгоритмов

- Пусть сортировка вставками написана экономно, на низкоуровневом языке и требует  $2n^2$  операций
- Пусть сортировка слиянием написана неэкономно и на ??? и поэтому требует  $50n \log n$  операций
- Сортируем 1 терабайт (около 274,877,906,944 4-байтовых чисел)

Для суперкомпьютера

$$\frac{2 \times (274,877,906,944^2) = 151,115,727,451,828,646,838,272}{200 P\text{flops} = 200 \times 10^{15}}$$

$\approx 755,578.64$  секунды  $\approx 9$  дней

## О пользе быстрых алгоритмов

- Пусть сортировка вставками написана экономно, на низкоуровневом языке и требует  $2n^2$  операций
- Пусть сортировка слиянием написана неэкономно и на ??? и поэтому требует  $50n \log n$  операций
- Сортируем 1 терабайт (около 274,877,906,944 4-байтовых чисел)

Для суперкомпьютера

$$\frac{2 \times (274,877,906,944^2)}{200 P\text{flops} = 200 \times 10^{15}}$$

$\approx 755,578.64$  секунды  $\approx 9$  дней

$$\log_2(274877906944) = 38$$
$$2^{38} = 274877906944$$

Для домашнего компьютера

$$\frac{50 \times (274,877,906,944 \times 38)}{3\text{GHz} = 3 \times 10^9}$$

## О пользе быстрых алгоритмов

- Пусть сортировка вставками написана экономно, на низкоуровневом языке и требует  $2n^2$  операций
- Пусть сортировка слиянием написана неэкономно и на ??? и поэтому требует  $50n \log n$  операций
- Сортируем 1 терабайт (около 274,877,906,944 4-байтовых чисел)

Для суперкомпьютера

$$\frac{2 \times (274,877,906,944^2)}{200 P\text{flops} = 200 \times 10^{15}}$$

$\approx 755,578.64$  секунды  $\approx 9$  дней

$$\log_2(274877906944) = 38$$
$$2^{38} = 274877906944$$

Для домашнего компьютера

$$\frac{50 \times (274,877,906,944 \times 38)}{3\text{GHz} = 3 \times 10^9}$$

$\approx 174,089.34$  секунды  $\approx 48$  часов

# Одни из самых быстрых систем в своих областях



distributed general-purpose  
cluster-computing framework



high-performance protocol  
servers & clients



<https://kafka.apache.org/>

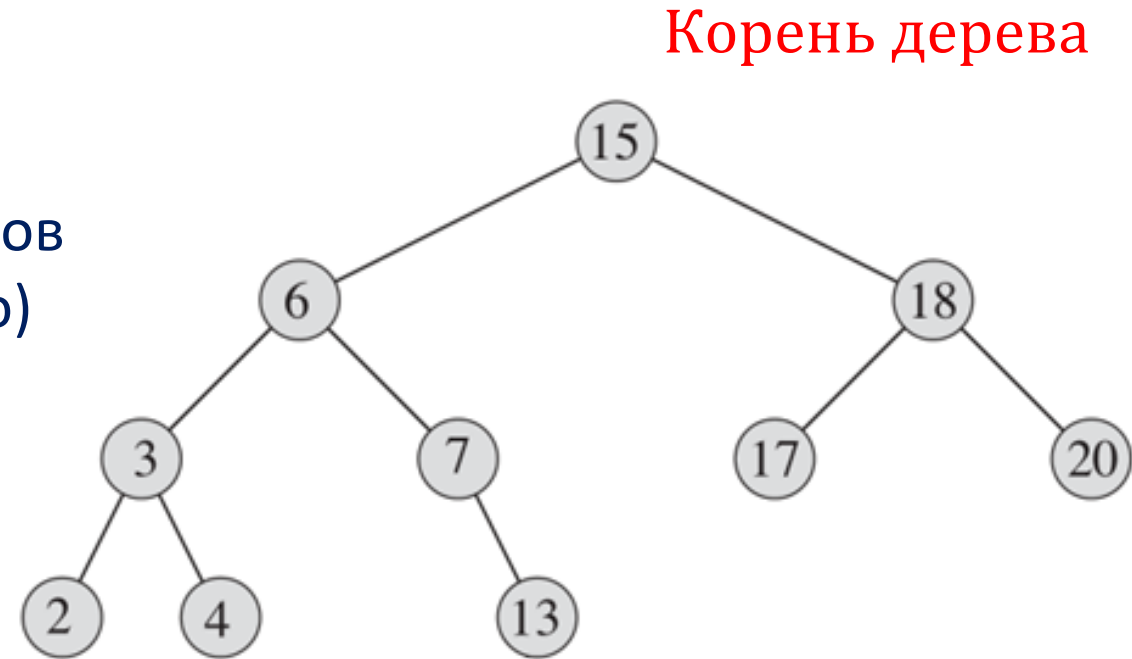
<https://spark.apache.org/>

<https://netty.io/>



# Двоичное дерево поиска

- **Дерево** состоит из узлов и ребер
- **Корень дерева** – один из узлов
- **Ребра** соединяют родителя и потомков
- У родителя может быть не более двух потомков
- В каждом узле – находится ключ (напр., число)
- Ключ левого потомка  $<$  ключа родителя
- Ключ правого потомка  $>$  ключа родителя



```
struct node
{
    int data;           ключ
    struct node *left;  указатель на левого потомка
    struct node *right; указатель на правого потомка
};
```

```
ITERATIVE-TREE-SEARCH( $x, k$ )
1  while  $x \neq \text{NIL}$  and  $k \neq x.\text{key}$ 
2      if  $k < x.\text{key}$ 
3           $x = x.\text{left}$ 
4      else  $x = x.\text{right}$ 
5  return  $x$ 
```

Подробнее о двоичном дереве поиска мы узнаем чуть позже







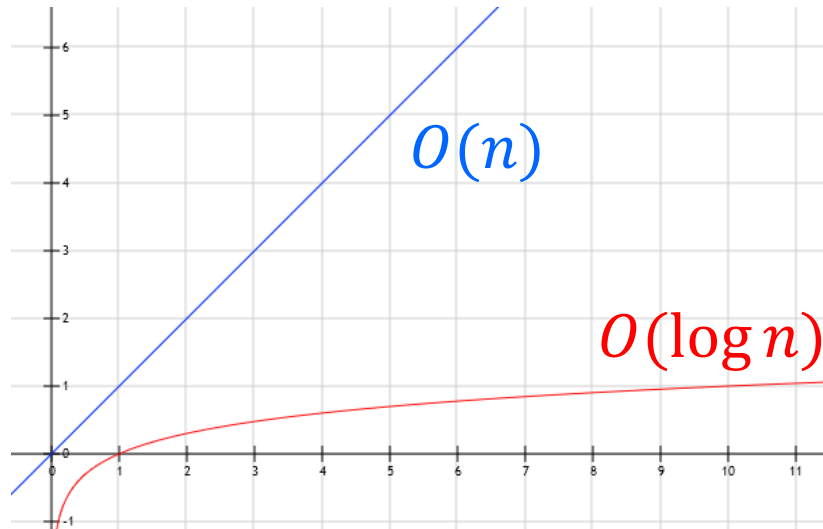
# Поиск элемента: что быстрее?

*Array* – массив длины  $n$

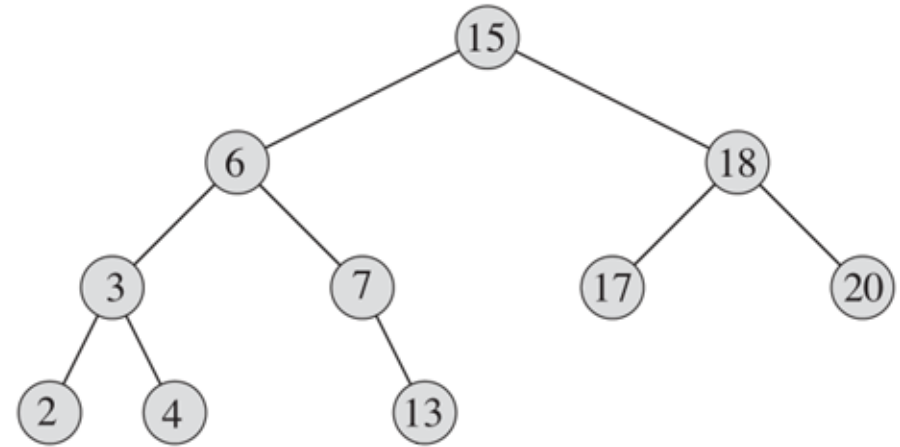


```
for(int i = 0; i < n; i++) {  
    if (Array[i] == n - 1) return i;  
}  
return -1;
```

сложность алгоритма?  $O(n)$



$x$  – двоичное дерево из  $n$  ключей высоты  $\log n$



сложность  
алгоритма?

$O(\log n)$

```
ITERATIVE-TREE-SEARCH( $x, k$ )  
1  while  $x \neq \text{NIL}$  and  $k \neq x.\text{key}$   
2      if  $k < x.\text{key}$   
3           $x = x.\text{left}$   
4      else  $x = x.\text{right}$   
5  return  $x$ 
```

# Поиск элемента: что быстрее?

*Array* – массив длины  $n$

0	1	2	3	4	5	6	7	...	
---	---	---	---	---	---	---	---	-----	--

```
for(int i = 0; i < n; i++) {  
    if (Array[i] == n - 1) return i;  
}  
return -1;
```

Система \_\_\_\_\_

Процессор: Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz 2.50 GHz

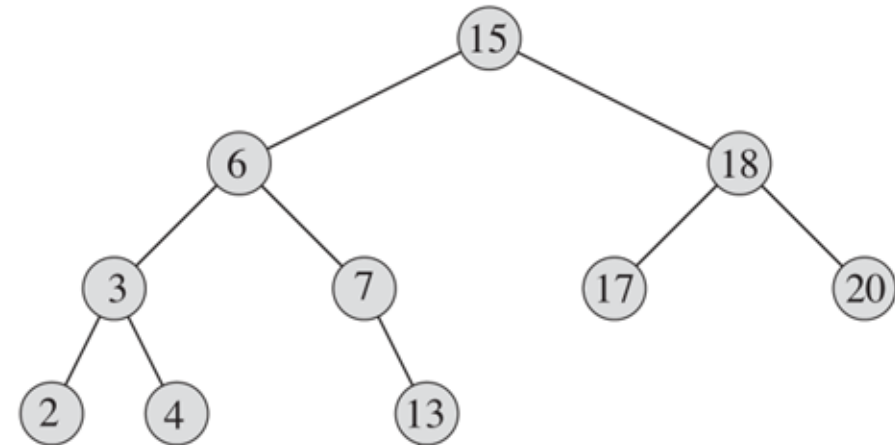
$n = 10$

Количество попыток поиска: 1000

Массив справился за 833 нс

Дерево справилось за 3045 нс

$x$  – двоичное дерево из  $n$  ключей:  
0, 1, 2, 3, ...



ITERATIVE-TREE-SEARCH( $x, k$ )

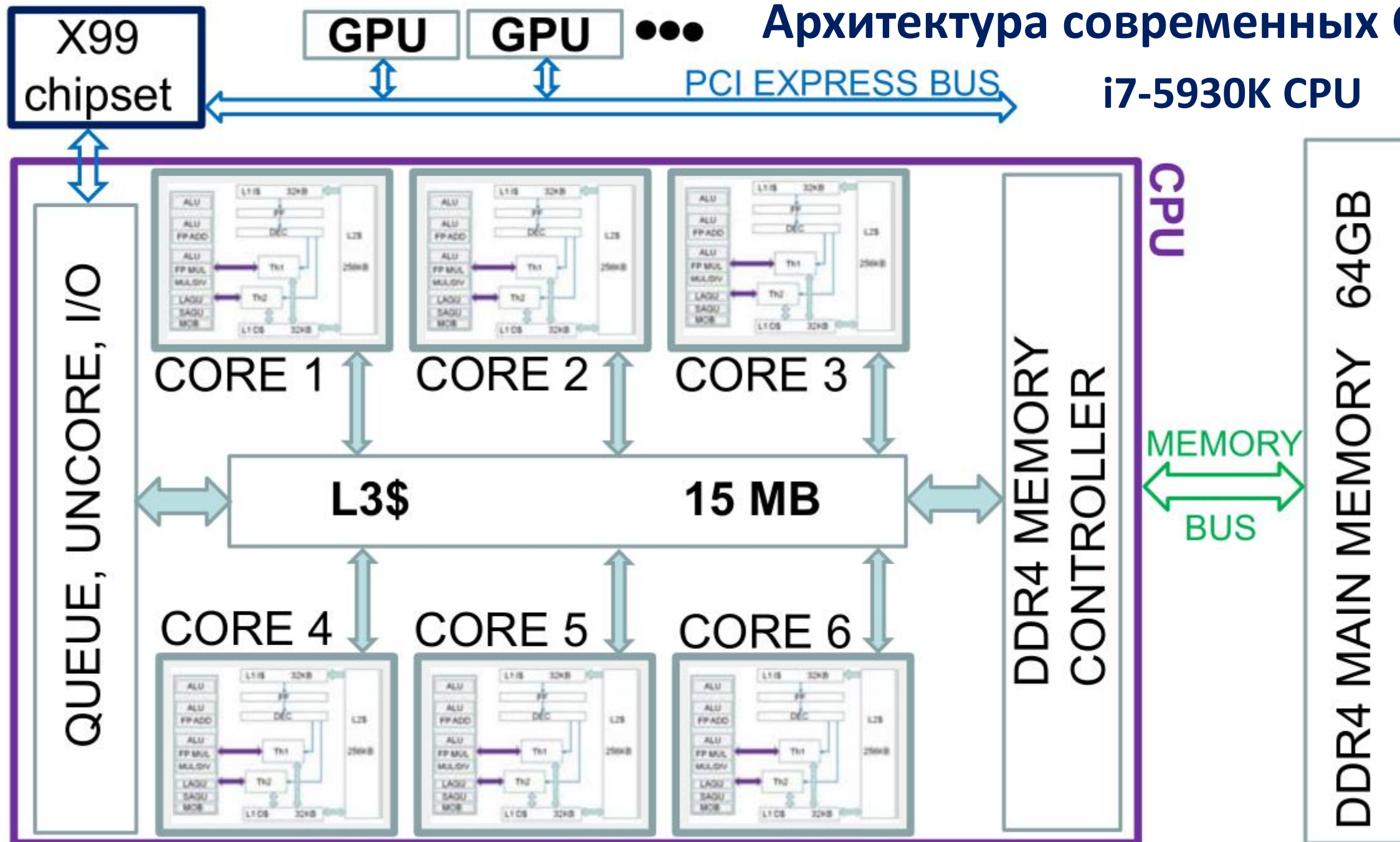
```
1  while  $x \neq \text{NIL}$  and  $k \neq x.\text{key}$   
2      if  $k < x.\text{key}$   
3           $x = x.\text{left}$   
4      else  $x = x.\text{right}$   
5  return  $x$ 
```





# Архитектура современных CPU

i7-5930K CPU





### Процессор Intel Core i9-10900K OEM

**50 299 ₺**

[LGA 1200, 10 x 3700 МГц, L2 - 2.5 МБ, L3 - 20 МБ, 2xDDR4-2933 МГц, Intel UHD Graphics 630, TDP 95 Вт]

Код товара: 1645612



### Процессор Intel Core i9-10900K BOX

**51 499 ₺**

[LGA 1200, 10 x 3700 МГц, L2 - 2.5 МБ, L3 - 20 МБ, 2xDDR4-2933 МГц, Intel UHD Graphics 630, TDP 125 Вт]

Код товара: 1689781



### Процессор Intel Core i9-10900K BOX

**51 499 ₺**

[LGA 1200, 10 x 3700 МГц, L2 - 2.5 МБ, L3 - 20 МБ, 2xDDR4-2933 МГц, Intel UHD Graphics 630, TDP 125 Вт]

Код товара: 1645625



### Процессор Intel Core i9-10900X OEM

**52 199 ₺**

[LGA 2066, 10 x 3700 МГц, L2 - 10 МБ, L3 - 19.25 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605267



### Процессор Intel Core i9-10900X BOX

**52 999 ₺**

[LGA 2066, 10 x 3700 МГц, L2 - 10 МБ, L3 - 19.25 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605260



### Процессор Intel Core i9-10920X OEM

**58 999 ₺**

[LGA 2066, 12 x 3500 МГц, L2 - 12 МБ, L3 - 19.25 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605274



### Процессор Intel Core i9-10920X BOX

**59 999 ₺**

[LGA 2066, 12 x 3500 МГц, L2 - 12 МБ, L3 - 19.25 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605271



### Процессор Intel Core i9-10940X OEM

**66 999 ₺**

[LGA 2066, 14 x 3300 МГц, L2 - 14 МБ, L3 - 19.25 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605279



### Процессор Intel Core i9-10940X BOX

**67 999 ₺**

[LGA 2066, 14 x 3300 МГц, L2 - 14 МБ, L3 - 19.25 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605276



### Процессор Intel Core i9-10980XE OEM

**87 999 ₺**

[LGA 2066, 18 x 3000 МГц, L2 - 18 МБ, L3 - 24.75 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605284



### Процессор Intel Core i9-10980XE BOX

**88 999 ₺**

[LGA 2066, 18 x 3000 МГц, L2 - 18 МБ, L3 - 24.75 МБ, 4xDDR4-2933 МГц, TDP 165 Вт]

Код товара: 1605281

# Время доступа к данным

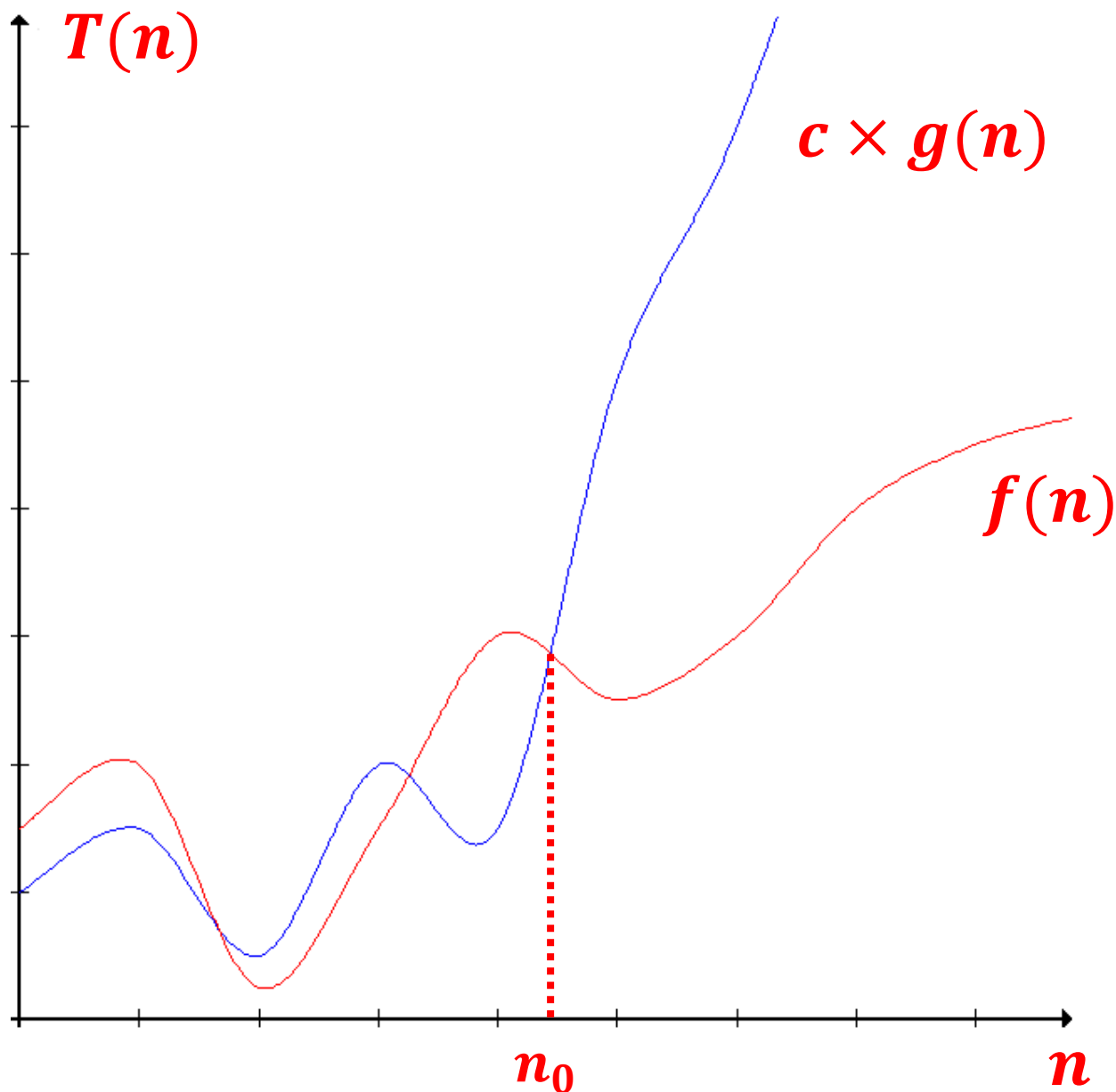
Core i7 Xeon 5500 Series Data Source Latency (approximate)

[Pg. 22]

local	L1 CACHE hit,	~4 cycles ( 2.1 - 1.2 ns )
local	L2 CACHE hit,	~10 cycles ( 5.3 - 3.0 ns )
local	L3 CACHE hit, line unshared	~40 cycles ( 21.4 - 12.0 ns )
local	L3 CACHE hit, shared line in another core	~65 cycles ( 34.8 - 19.5 ns )
local	L3 CACHE hit, modified in another core	~75 cycles ( 40.2 - 22.5 ns )
remote	L3 CACHE (Ref: Fig.1 [Pg. 5])	~100-300 cycles ( 160.7 - 30.0 ns )
local	DRAM	~60 ns
remote	DRAM	~100 ns

# В чем причина? или Скорость роста функций

Какова  
величина  
 $n$  ?



$$f(n) = O(g(n))$$



$$\begin{aligned} &\exists c, n_0 > 0: \\ &0 \leq f(n) \leq c \times g(n) \\ &\text{для } \forall n \geq n_0 \end{aligned}$$

т.е. при достаточно  
большом  $n$

$f(n)$  и  $g(n)$   
асимптотически  
положительны

# Поиск элемента: что быстрее?

*Array* – массив длины  $n$

0	1	2	3	4	5	6	7	...	
---	---	---	---	---	---	---	---	-----	--

```
for(int i = 0; i < n; i++) {  
    if (Array[i] == n - 1) return i;  
}  
return -1;
```

Система \_\_\_\_\_

Процессор: Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz 2.50 GHz

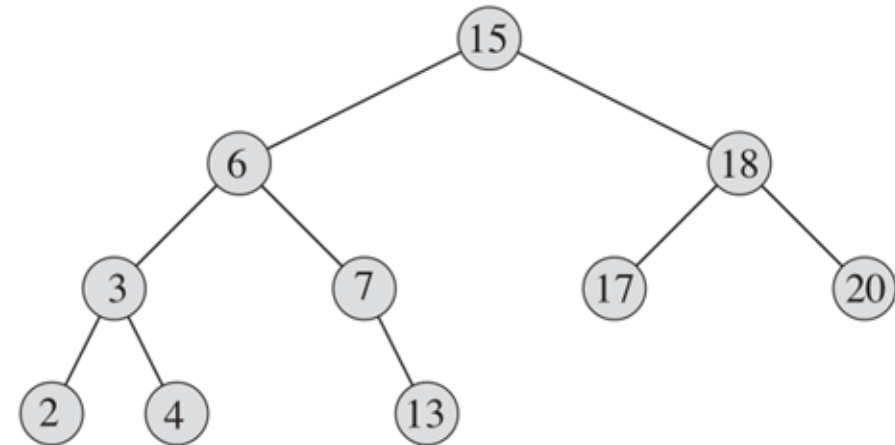
$n = 100$

Количество попыток поиска: 1000

Массив справился за 4793 нс

Дерево справилось за 1917 нс

$x$  – двоичное дерево из  $n$  ключей:  
0, 1, 2, 3, ...



ITERATIVE-TREE-SEARCH( $x, k$ )

```
1  while  $x \neq \text{NIL}$  and  $k \neq x.\text{key}$   
2      if  $k < x.\text{key}$   
3           $x = x.\text{left}$   
4      else  $x = x.\text{right}$   
5  return  $x$ 
```





NATIONAL RESEARCH  
UNIVERSITY

# Благодарю за внимание!

Рамон Антонио Родригес Залепинос  
[arodriges@hse.ru](mailto:arodriges@hse.ru)