Вопросы: <https://docs.google.com/document/d/10CwDCTxfKa6020fCPzoeJIpeLRzcyecUk1ZVyeh5JPY/mobilebasic>

Видеолекции ШАД:

<https://yadi.sk/d/sW5pwqgG3Nndfr>

### **Лекция 1 (12.09.16)**

Модель вычислений во внешней памяти. Сортировка во внешней памяти.

### **Лекция 2 (19.09.16)**

Задача List Ranking. Рандимизированное и детерминированное решение. Задача о вершинной раскраске графа степени k в k+1 цвет.

### **Лекция 3 (28.09.16)**

B+ деревья. Буфферизированные деревья. Кучи во внешней памяти.

### **Лекция 4 (5.10.16)**

Представление графов во внешней памяти. Обход в ширину неориентированного графа. Поиск связных компонент методом Борувки.

### **Лекция 5 (12.10.16)**

Cache oblivious lookahead array (COLA). Fractional cascading.

### **Лекция 6 (17.10.16)**

Кеширование. OPT- и LRU-стратегии вытеснение, соотношение качества их работы. Сache oblivious алгоритмы. Cache oblivious транспонирование матриц. Cache oblivious бинарный поиск. van Emde Boas layout.

### **Лекция 7 (24.10.16)**

Алгоритм Munro-Paterson выбора порядковой статистики за ограниченное число проходов.

### **Лекция 9 (31.10.16)**

Выбор подмножеств из потоков. Резервуарное семплирование.

Основные понятия потоковых алгоритмов: модели cash-register и turnstile.

Алгоритм Misra-Gries.

### **Лекция 9 (7.11.16)**

Count Sketch и Count-Min Sketch.

### **Лекция 10 (14.11.16)**

AMS и BJKST.

### **Семинар 1 (19.09.16)**

Поверхностный рассказ про архитектура компьютера. Внутреннее устройство HDD и SSD. Особенности Linux при работе с диском. Алгоритм Join, структуры Stack, Queue во внешней памяти.

Полезные ссылки:

* [Википедия про архитектуру, north, south bridges](https://en.wikipedia.org/wiki/Motherboard#Integrated_peripherals)
* [Алгоритмы упорядочивания запросов к HDD](http://www.ijcsi.org/papers/IJCSI-11-1-1-74-79.pdf)
* [Хороший цикл статей про SSD](http://codecapsule.com/2014/02/12/coding-for-ssds-part-1-introduction-and-table-of-contents/)
* [Статья про эксперименты с SSD и их выходы из строя](https://www.usenix.org/system/files/conference/fast13/fast13-final80.pdf)
* [Опции ext4](https://www.kernel.org/doc/Documentation/filesystems/ext4.txt), про [data=ordered, data=journal](http://unix.stackexchange.com/questions/127235/is-data-journal-safer-for-ext4-as-opposed-to-data-ordered).

### **Семинар 2 (26.09.16)**

Окончание про SSD и HDD. Алгоритм Distribution Sort (аналог QuickSort во внешней памяти). Оценка снизу на количество IO-операций, необходимых для сортировки. Задача построения эйлерова обхода дерева и её приложения.

Полезные ссылки:

* [Про устройство чтения/записи с HDD platter](http://hddscan.com/doc/HDD_Tracks_and_Zones.html)
* Программирование SSD: [Willow](https://cseweb.ucsd.edu/~swanson/papers/OSDI2014-Willow.pdf), [Biscuit](http://isca2016.eecs.umich.edu/wp-content/uploads/2016/07/3A-1.pdf).
* [Статья про NVMe протокол](http://blogs.cisco.com/datacenter/nvme-for-absolute-beginners).
* [Нижняя оценка на сортировку, про Distribution Sort и другие алгоритмы](https://www.cs.cmu.edu/~guyb/realworld/slidesF10/AV88.pdf)

### **Семинар 3 (03.10.16)**

Обсуждение второго практического домашнего задания, Radix Sort. Построение хэш-таблицы во внешней памяти. Геометрические задачи во внешней памяти.

Полезные ссылки:

* [Оригинальная статья про Linear Hashing](http://www.cs.cmu.edu/~christos/courses/826-resources/PAPERS+BOOK/linear-hashing.PDF), [Linear hashing with expansions](https://www.researchgate.net/publication/221310070_Linear_Hashing_with_Partial_Expansions).
* [Статья про 1 + 1/sqrt(B) приближение в задаче хэширования](http://citeseer.ist.psu.edu/viewdoc/citations;jsessionid=FBFF99911C5C0B6F0AAEC3A0A622C351?doi=10.1.1.112.9063).
* [Статья про решение геометрических задач во внешней памяти](http://www.ittc.ku.edu/~jsv/Papers/GTV93.ecg.pdf)

### **Семинар 4 (10.10.16)**

Краткий обзор того, как решаются задачи BFS, DFS и SSSP на ненаправленных графах во внешней памяти. Модель Parallel Sliding Window, которая реализуется библиотекой GraphChi и позволяет на практике решать задачи на больших графах. Пример решения задачи MST в модели PSW.

Полезные ссылки:

* [Лекции по графовым алгоритмам, где разбираются BFS, DFS и SSSP](http://cs.au.dk/~large/ioS05/Znotes.pdf)
* [GraphChi: Large-Scale Graph Computation on Just a PC](https://www.cs.cmu.edu/~pavlo/courses/fall2013/static/papers/osdi2012-graphchi.pdf)
* [Minimum spanning tree using PSW model](https://people.eecs.berkeley.edu/~jshun/psw_sea2014.pdf)

### **Семинар 5 (24.10.16)**

Техническое устройство оперативной памяти: SRAM, DRAM, 3D XPoint, HBM. Мультипроцессорные архитектуры и особенности устройства кэшей и памяти в них. Эффективные методы реализации бинарного поиска.

Полезные ссылки:

* [Древняя статья про внутреннее устройство RAM](http://www.cse.scu.edu/~tschwarz/coen180/LN/DRAM.html). Найти что-то более современное мне не удалось.
* [Хороший цикл статей про shared memory architecture](http://frankdenneman.nl/2016/07/06/introduction-2016-numa-deep-dive-series/)
* [Статья про гарантии работы с памятью и алгоритм Деккера](https://bartoszmilewski.com/2008/11/05/who-ordered-memory-fences-on-an-x86/)
* [Статья про Van Emde Boas layout](http://www.cs.au.dk/~gerth/papers/soda02.pdf)

### **Семинар 6 (31.10.16)**

Струтуры данных для приближенное проверки наличия ключа: Bloom Filter, Quotient Filter. Теорема об оптимальности алгоритмов кэширования и её доказательство.

Полезные ссылки:

* [Don’t Thrash: How to Cache Your Hash on Flash](http://vldb.org/pvldb/vol5/p1627_michaelabender_vldb2012.pdf) – статья про Quotient Filter.
* [Оригинальная статья про оптимальное кэширование](https://www.cs.cmu.edu/~sleator/papers/amortized-efficiency.pdf)

### **Семинар 7 (07.11.16)**

Разбор первого домашнего задания. Обзор алгоритмов для поиска квантилий и построение гистограмм в потоке числе. Алгоритм Q-Digest.

Полезные ссылки:

* [Обзор алгоритмов для поиска квантилей и построения гистограмм](https://www.cs.ucsb.edu/~suri/psdir/ency.pdf)
* [Алгоритм Greenwald и Khanna](http://infolab.stanford.edu/~datar/courses/cs361a/papers/quantiles.pdf)
* [Q-Digest](https://www.inf.fu-berlin.de/lehre/WS11/Wireless/papers/AgrQdigest.pdf)
* [T-Digest – новый алгоритм с непонятными теоретическими гарантиями](https://github.com/tdunning/t-digest/blob/master/docs/t-digest-paper/histo.pdf)

### **Семинар 8 (14.11.16)**

Окончание про Q-Digest. Нижние оценки на время работы streaming-алгоритмов.

Полезные ссылки:

* [Лекции про streaming алгоритмы](https://courses.cs.washington.edu/courses/cse522/14sp/lectures/) – про нижние оценки рассказывается в лекциях 11 и 12.

### **Семинар 9 (21.11.16)**

Обзор алгоритмов для подсчета количества различных элементов в потоке: PCSA, LogLog, HyperLogLog, Linear Counting, Lumbroso Min Count et.c.

Разбор второго домашнего задания.

Полезные ссылки:

* Оригинальные статьи: [PCSA](http://algo.inria.fr/flajolet/Publications/FlMa85.pdf), [LogLog](http://algo.inria.fr/flajolet/Publications/DuFl03.pdf), [HyperLogLog](http://algo.inria.fr/flajolet/Publications/FlFuGaMe07.pdf), [Lumbroso Min Count](https://www.dmtcs.org/dmtcs-ojs/index.php/proceedings/article/viewFile/dmAM0134/3324.pdf), [Linear Counting](http://organ.kaist.ac.kr/Prof/pdf/Whang1990(linear).pdf), [Multiresolution Bitmap](http://pages.cs.wisc.edu/~estan/publications/countingbitmaps.pdf).
* Статья про практическую реализацию HyperLogLog: [HyperLogLog in practice](https://stefanheule.com/papers/edbt13-hyperloglog.pdf)

### **Семинар 10 (05.12.16)**

Потоковые алгоритмы в моделе Sliding Window. Задача подсчета суммы в такой моделе.

## 

## 

## **Рекомендуемая литература**

### **Архитектура компьютера**

[Computer Architecture, Fifth Edition: A Quantitative Approach](http://it.tdt.edu.vn/~tttin/giangday/KTMT/Computer-Architecture-Fifth-Edition-A-Quantitative-Approach.pdf) – подробная книжка про устройство процессора/памяти

### **Про все понемногу**

* Ulrich Drepper, "*What every programmer should know about memory*" ([link](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.91.957&rep=rep1&type=pdf)).
* Jeffery Scott Vitter, "*Algorithms and data structures for external memory*" ([link](http://www.ittc.ku.edu/~jsv/Papers/Vit.IO_book.pdf)).
* U. Meyer, P. Sanders, and J. Sibeyn (eds.), "*Algorithms for Memory Hierarchies*" (<http://www.win.tue.nl/~hermanh/teaching/2IL35/AMM/01-models-and-lower-bounds.pdf>), (<http://www.win.tue.nl/~hermanh/teaching/2IL35/AMM/02-basic-data-structures.pdf>), (<http://www.win.tue.nl/~hermanh/teaching/2IL35/AMM/03-techniques.pdf>), (<http://www.win.tue.nl/~hermanh/teaching/2IL35/AMM/04-elementary-graph-algorithms.pdf>) - очень неплохой обзор про алгоритмы по внешней памяти (основные примеры, оценки снизу, графы, структуры данных).
* Jeff Erickson, "*CS 473: Topics in Analysis of Algorithms (Fall 2003)*" (<http://compgeom.cs.uiuc.edu/~jeffe/teaching/473/>) - записки лекций

### **Подсчет числа различных элементов**

* Bar-Yossef, et. al, "*Counting distinct elements in a data stream*" ([link](http://citeseer.ist.psu.edu/viewdoc/download;jsessionid=33854A50C4CDB2BCFEC12F5E6E9FD21F?doi=10.1.1.12.6276&rep=rep1&type=pdf)).

### **Структуры данных**

* L. Arge, "*The Buffer Tree: a technique for designing batched external data structures*" (<http://www.win.tue.nl/~hermanh/teaching/2IL35/arge-buffer-tree.pdf>)
* Michael A. Bender, Martin Farach-Colton, Jeremy T. Fineman, Yonatan R. Fogel, Bradley C. Kuszmaul, Jelani Nelson, "*Cache-Oblivious Streaming B-Trees*" (<http://supertech.csail.mit.edu/papers/sbtree.pdf>) - статья про COLA

### **Алгоритмы на графах**

* N. Zeh, "*I/O-Efficient Graph Algorithms* (<http://users.cs.dal.ca/~nzeh/Teaching/6104/Notes/zeh02.pdf>)
* J. Abello, A. Buchsbaum, J. Westbrook, "*A Functional Approach to External Graph Algorithms*" <http://citeseer.ist.psu.edu/viewdoc/download;jsessionid=CE9E1F1207053A2A612446D71F7D9AA3?doi=10.1.1.30.1027&rep=rep1&type=pdf>

### **Cache-oblivious (введение в модель + базовые алгоритмы)**

* Aggarwal, Vitter, "*The input/output complexity of sorting and related problems*" <http://portal.acm.org/citation.cfm?id=48529.48535> - первоначальная статья, в которой была введена двухуровневая модель памяти.
* Frigo, Leiserson, Prokop, "*Cache-oblivious algorithms*" <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.4601&rep=rep1&type=pdf> - первоначальная статья, в которой была введена модель идеального кеша.
* Demaine, "*Cache-oblivious algorithms and data structures*" <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.1230&rep=rep1&type=pdf> - более подробное и внятное описание базовых алгоритмов: перемножение матриц, сортировка, бинарный поиск. Также есть введение в динамические структуры.

### **Cache-oblivious (экспериментальные результаты)**

* Brodal, Fagerberg, Vinther, "*Engineering a cache-oblivious sorting algorithm*" <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.88.9738&rep=rep1&type=pdf> - производительность различных сортировочных алгоритмов на разных платформах.
* Brodal, Fagerberg, Jacob, "*Cache-oblivious search trees via binary trees of small height*" <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.83.6795&rep=rep1&type=pdf> - производительность деревьев поиска, упакованных и индексированных в памяти различными способами.