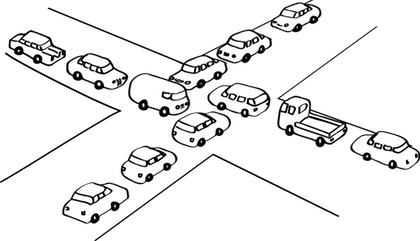
**Concurrency**:

теория и практика



Страница для студентов

Тема курса в одном предложении: теория и практика **конкурентных (concurrent)** **вычислений** в **разделяемой памяти**.

**План курса:**

* Введение, потоки и операционная система
* Взаимное исключение: протоколы и свойства
* Синхронизация с помощью условных переменных и семафоров
* Мелкогранулярные блокировки, хэш-таблицы, списки и скип-списки
* Когерентность кэшей и спинлоки
* Модель памяти sequential consistency for data race free programs (SC-DRF)
* Неблокирующая синхронизация (гарантии lock-freedom, wait-freedom, obstruction-freedom), lock-free контейнеры (стек, очередь, список, хэш-таблица), сборка мусора
* Линеаризуемость и другие модели консистентности для высокоуровневых объектов
* Сила атомарных операций: консенсус и wait-free иерархия
* Универсальная lock-free и wait-free конструкция
* Дополнительные темы: concurrent GC, изоляция транзакций и транзакционная память, применение идей консенсуса и универсальной конструкции в распределенных системах на примере RSM с помощью RAFT.

**Диаграмма зависимостей тем**



**Материалы по темам:**

[Слайды лекций](https://yadi.sk/d/TbkZMpkC3EELaB)

[Статьи по темам лекций](https://yadi.sk/d/ho2k-Hxv38WVso)

**Про операционные системы**:

<http://duartes.org/gustavo/blog/post/anatomy-of-a-program-in-memory/>

<http://duartes.org/gustavo/blog/post/when-does-your-os-run/>

<http://duartes.org/gustavo/blog/post/system-calls/>

<http://duartes.org/gustavo/blog/post/what-does-an-idle-cpu-do/>

**Взаимное исключение**:

* Art of Multiprocessor Programming, глава 2
* Notes on Theory of Distributed Computing, глава 17

**Про семафоры**:

Книга Little Book of Semaphores: <http://greenteapress.com/wp/semaphores/>

Статья про применения семафоров: <http://preshing.com/20150316/semaphores-are-surprisingly-versatile/>

**Про фьютексы**:

Статья [Futexes Are Tricky](https://www.akkadia.org/drepper/futex.pdf)

Про реализацию условных переменных и мьютексов с помощью фьютексов:

<https://locklessinc.com/articles/mutex_cv_futex/>

Про локи на фьютексах:

<https://bartoszmilewski.com/2008/09/01/thin-lock-vs-futex/>

**Про списки**:

Список на оптимистичных блокировках:

[A Lazy Concurrent List-Based Set Algorithm](http://people.csail.mit.edu/shanir/publications/Lazy_Concurrent.pdf)

Скип-лист на оптимистичных блокировках:

[A Simple Optimistic Skip-List Algorithm](http://people.csail.mit.edu/shanir/publications/LazySkipList.pdf)

[Реализация скип-листа от инженеров Facebook](https://github.com/facebook/folly/blob/master/folly/ConcurrentSkipList.h)

Практические оптимизации при реализации скип-листов:

[Skip Lists: Done Righ](http://ticki.github.io/blog/skip-lists-done-right/)

Про применение скип-листов в движках баз данных:

<http://blog.memsql.com/the-story-behind-memsqls-skiplist-indexes/>

**Про когерентность кэшей и барьеры памяти:**

[Memory Barriers: Hardware View for Software Hackers](http://www.rdrop.com/users/paulmck/scalability/paper/whymb.2010.07.23a.pdf)

[Симулятор протоколов когерентности](https://www.scss.tcd.ie/~jones/vivio/caches/ALL%20protocols.htm)

Отличный лонг-рид про процессоры: <http://www.lighterra.com/papers/modernmicroprocessors/>

**Про Happens-Before и модели памяти:**

Про порядок событий в модели обмена сообщениями и про логические часы: [Capturing Causality in Distributed Systems](http://aqualab.cs.northwestern.edu/component/attachments/download/302)

Отличная вводная статья про модели памяти: [Memory Models: A Case for Rethinking Parallel Languages and Hardware](https://cacm.acm.org/magazines/2010/8/96610-memory-models-a-case-for-rethinking-parallel-languages-and-hardware/pdf)

Дизайн модели памяти для языка C++: [Foundations of C++ Memory Model](http://www.hpl.hp.com/techreports/2008/HPL-2008-56.pdf)

Строгое формальное изложение модели памяти Java: [The Java Memory Model: A Formal Explanation](http://www-sop.inria.fr/members/Gustavo.Petri/publis/jmm-vamp07.pdf)

**Lock-Free:**

[Data Structures in the Multicore Age](http://people.csail.mit.edu/shanir/publications/p76-shavit.pdf) - обязательная к прочтению обзорная статья про дизайн лок-фри структур данных

[Simple, Fast, and Practical Non-Blocking and Blocking Concurrent Queue Algorithms](https://www.research.ibm.com/people/m/michael/podc-1996.pdf) - классическая лок-фри очередь Майкла-Скотта

[Split-Ordered Lists - Lock-free Resizable Hash Tables](http://www.cs.tau.ac.il/~afek/SplitOrderListHashSS03.pdf) - расширяемая лок-фри хэш-таблица на основе сортированного списка

Больше статей - [здесь](https://yadi.sk/d/ho2k-Hxv38WVso)

**Домашние задания**

Разминочные задачи: <https://deadlockempire.github.io>

Тестовый контест: <https://contest.yandex.ru/contest/3954/problems/>

1. [Взаимное исключение](https://docs.google.com/document/d/11fLWngAO8N0Za5FpuVHz8xS6CqphNwX9Redz033f2bU/edit?usp=sharing)

**Дедлайн**: 6 марта, 6:00

<https://contest.yandex.ru/contest/4067/problems/>

2. [Механизмы синхронизации](https://docs.google.com/document/d/1ULHWelxkSkTKh1dKvdeCK3ErFFfA07g_5mKsBbbuS9E/edit?usp=sharing)

**Дедлайн**: 20 марта, 6:00

<https://contest.yandex.ru/contest/4118/problems/>

3. [Блокирующая очередь и пул потоков](https://docs.google.com/document/d/16K3FK7stDlALrwMe3lAEz3fjjkq9iHON7VwS9QsflVA/edit?usp=sharing)

**Дедлайн**: 27 марта, 6:00

<https://contest.yandex.ru/contest/4115/problems/>

4. [Хэш-таблицы и списки](https://docs.google.com/document/d/19pz2AOKR0NDJYSJtYeW6T9pE1AZ72-6OHh7RIfuamEE/edit?usp=sharing) - 15 баллов

**Дедлайн**: хэш-таблица - 6 апреля, 6:00, список - 12 апреля, 06:00

<https://contest.yandex.ru/contest/4278/problems/>

5. [Когерентность кэшей](https://docs.google.com/document/d/1md5KJMVvYbyeQHZORrEnN0s6BobofIZWfAD84BHocTQ/edit?usp=sharing) - 5 баллов

**Дедлайн**: 18 апреля, 6:00

<https://contest.yandex.ru/contest/4366/problems>

В помощь: [Заметки про когерентность кэшей и спинлоки](https://docs.google.com/document/d/14Od6AdGdT68mAbjdwqFYbN1mZGL4UzQamO6GIoLoa0E/edit?usp=sharing)

6. [Модель памяти](https://docs.google.com/document/d/1b-SDqFIHYvlWtjFK3fuhLoo53nSD94CHSCvB4RG3Bh8/edit?usp=sharing) - 10 баллов

**Дедлайн**: 25 апреля

7. Lock-Free структуры данных - 25 баллов

<https://contest.yandex.ru/contest/4489/problems>

**Дедлайн**: стек - 9 мая, очередь - 15 мая, список и хэш-таблица - 21 мая

В помощь: [Заметки про реализацию lock-free списка](https://docs.google.com/document/d/1KWdYLFhgNRqq1LF5cCFBBQcACJxXcPGmFdZwGslzn48/edit?usp=sharing)

8. [Линеаризуемость](https://docs.google.com/document/d/16Jpjq9rJJQ3wGDrk-WcYxTxE9U7xveS1LBHOGBkDGls/edit?usp=sharing) - 10 баллов

**Дедлайн**: 9 мая

9. [Консенсус](https://docs.google.com/document/d/15uaVpm6YHdfnTdtdK-f1pC_nSuyXw62AT_O_M8ikYeI/edit?usp=sharing) - 10 баллов

**Дедлайн**: 15 мая

[Репозиторий с исходным кодом для задач](https://bitbucket.org/misty_fungus/fivt/src)

[Регламент сдачи домашних заданий](https://docs.google.com/document/d/1QdsD2EcluQ0npFIn-Hxmjl4r9p6OTpuZ4H5iNmNhnzQ/edit?usp=sharing)

**Style Guide**

Google C++ Style Guide: <https://google.github.io/styleguide/cppguide.html>

Изучаем стайл гайд на примере кода с подробными комментариями:

<https://bitbucket.org/snippets/rlipovsky/qAR9M>

[Вики на bitbucket](https://bitbucket.org/misty_fungus/fivt/wiki/Home)

**Компилятор**

Рекомендуемый компилятор - clang.

Он поддерживает набор санитайзеров, которые умеют обнаруживать UB, утечки и проезды по памяти, многопоточные гонки.

Документация по санитайзерам:

<http://clang.llvm.org/docs/UndefinedBehaviorSanitizer.html>

<http://clang.llvm.org/docs/AddressSanitizer.html>

<http://clang.llvm.org/docs/ThreadSanitizer.html>

**Учебники**

Все [книги](https://yadi.sk/d/p6WR9Ts4oxZ7t) доступны в электронном виде.

**Главный учебник**: **The Art of Multiprocessor Programming** / Maurice Herlihy, Nir Shavit **[AMP]**

Если вы можете прочитать только одну книгу про многопоточное программирование, то этой книгой должна быть AMP.

Теория:

**Concurrent Programming: Algorithms, Principles, and Foundations** / Michel Raynal **[CP-APF]**

**Notes on Theory of Distributed Systems** / James Aspens **[NTDS]**

Глава II: Shared Memory

Практика:

**Is Parallel Programming Hard, And, If So, What Can You Do About It?** / Paul E. McKenney **[IPPH]**

Книга многопоточность от разработчика ядра Linux

**Java Concurrency in Practice** / Brian Goetz, Doug Lea, Joshua Bloch, ... **[JCIP]**

Паттерны/идиомы многопоточного программирования для практиков на Java.

**C++ Concurrency in Action** / Anthony Williams **[CPPCIA]**

Практика многопоточного программирования на C++11. Книга доступна на русском языке!

Про операционные системы:

**xv6 - A Simple, Unix-like Teaching Operating System**

Про архитектуру процессоров:

**Computer Architecture: A Quantitative Approach** / Hennessy, Patterson

Про С++:

**Effective Modern C++** / Scott Meyers

**Материалы курса Maurice Herlihy по учебнику Art of Multiprocessor Programming**

[Видеозаписи лекций](http://www.brown.edu/cis/sta/dev/herlihy_csci1760_fa12/)

[Слайды](http://booksite.elsevier.com/9780123705914/?ISBN=9780123705914)