

Петрозаводский Государственный Университет  
Институт математики и информационных технологий  
Кафедра Информатики и математического обеспечения

## Учебный процесс ИМИТ. Кафедра ИМО

Заведующий кафедрой Информатики и Математического  
обеспечения, к. т. н., доцент,

Ю. А. Богоявленский  
[ybgv@cs.karelia.ru](mailto:ybgv@cs.karelia.ru)  
<http://cs.karelia.ru>



# Содержание

- Введение.
- Профессиональные задачи.
- Математика и информатика.
- Математики — лауреаты премии Тьюринга.
- Почему математическая культура полезна профессионалам ИКТ.
- Определяющая роль математической культуры в сфере ИКТ.
- Как организовать учебную работу.
- Современные учебные планы.
- Информационно-вычислительная инфраструктура ИМИТ.
- ОС UNIX → Linux.
- Программирование “в малом”. Язык C на все времена.
- Командный интерфейс против графического.
- От учебы к профессиональной карьере.

# Введение

Новый важный этап интеллектуального развития.

- **Цель студента** — сформировать культуру и навыки решения профессиональных **задач**.
  - ▶ **Мотивация.** Очень хотеть решить задачу.
  - ▶ **Знания.** Владеть объектами, их свойствами и связями предметной области.
  - ▶ **Анализ.** Уметь выполнять исчерпывающий анализ условий и требований постановки задачи.
  - ▶ **Синтез.** Строить путь от исходных данных к решению задачи.
- **Математика** — наилучший инструмент для формирования культуры решения задач, как профессиональных, так и жизненных.
- Черная овца в Австралии.
- Очень полезная книга. Д. Пойа Как решать задачу

# Введение. Профессиональные задачи.

## ■ Алгоритмы

- ▶ Знать основные классы. Владеть основами оценки сложности.
- ▶ Выбирать, комбинировать, модифицировать, разрабатывать.

## ■ Структуры данных

- ▶ Знать базовые структуры и их свойства.
- ▶ Понимать алгоритмы работы с элементами и их временную сложность.
- ▶ Выбирать, комбинировать, применять.

## ■ Компьютерные сети

- ▶ Знать уровни абстракции протоколов.
- ▶ Разрабатывать сетевые системы.

## ■ Программные проекты

- ▶ Владеть технологиями и инструментами командной разработки.
- ▶ Выбирать архитектуру. Оценивать стоимость. Тестировать.

## ■ Информационные системы

- ▶ Выбирать, проектировать архитектуру.
- ▶ Эксплуатировать. Сопровождать.

# Математика и информатика

## Центральное понятие — Алгоритм

В 1936 г. **Алонзо Чёрч (Alonzo Church)**, **Эмиль Пост (Post Emil)** и **Алан Тьюринг (Alan Turing)** начали разработку **теории алгоритмов** (и их сложности), в т.ч. на основе абстрактных вычислительных **машин Тьюринга, Поста**.

**Алан Тьюринг** в 1946 г. разработал проект ЭВМ ACE ( Automatic Computing Engine, Автоматическая вычислительная машина) с хранимой в памяти программой. Проект был реализован в 1950 г.

**Джон фон Нейман (John von Neumann)** в 1946 г. возглавил проект по созданию ЭВМ IAS (Institute of Advanced Studies). Он — основной автор монографии, описывающей логическую структура ЭВМ, названную «**архитектура фон Неймана**» и релизованную в десятках ЭВМ по всему миру, в том числе в советской БЭСМ.

**подавляющее большинство современных ЭВМ поддерживают эту архитектуру.**

# Математики — лауреаты премии Тьюринга

Премия Тьюринга (**Turing Award**) — самая престижная премия в информатике, вручаемая с 1966 г. Ассоциацией вычислительной техники (Association for Computing Machinery **ACM**)

- 1971.** Джон Маккарти — автор термина «искусственный интеллект» (1956), языка **Лисп**, парадигмы **функционального программирования**.
- 1977.** Джон Бэкус — руководитель разработки языка программирования **Фортран**, изобретатель формы **Бэкуса — Наура** для определения синтаксиса формальных языков.
- 1983.** Деннис Ритчи — один из авторов языка **C**, соавтор учебника **K&R**, участник разработки **ОС Unix**.
- 1986.** Роберт Тарьян — автор множества **алгоритмов на графах**, соавтор структур данных **“Фибоначчиева куча”** и **“Расширяющееся дерево”**.

## Математики — лауреаты премии Тьюринга

- 1999.** Фредерик Брукс (младший) — автор бестселлера “**Мифический человеко-месяц**” и “закона Брукса”: *Если проект не укладывается в сроки, то добавление рабочей силы задержит его еще больше..* Возглавлял разработки семейства мейнфреймов **IBM System/360** (теперь – IBM zSeries) и их операционной системы **OS/360**.
- 2002.** Рональд Ривест, Ади Шамир и Леонард Адлеман — авторы системы шифрования (**RSA Rivest — Shamir — Adleman**, 1977 год), которая широко применяется, в том числе в защищенном веб протоколе **https**.
- 2004.** Винтон Серф — вместе с Робертом Каном под руководством Леонарда Клейнрока в 1974 г. разработал семейство протоколов **TCP/IP**.
- 2013.** Лесли Лэмпорт — заложил основы теории распределённых систем (**автоматный подход, логические часы, подпись Лэмпорта**). Автор настольной издательской системы **L<sup>A</sup>T<sub>E</sub>X**, лауреат еще шести премий.

# Математики — лауреаты премии Тьюринга

**2020.** Джеффри Ульман — его учебники по **компиляторам, теории вычислений и базам данных** считаются стандартом.

## Не лауреат премии Тьюринга

**1988.** Ван Якобсон — разработал 4 алгоритма работы протокола TCP (в том числе **AIMD**), позволившие предотвращать коллапсы перегрузки Интернет в 1980 г.г. и перейти к ее широкому внедрению.

См. [M. Allman, V. Paxson, E. Blanton Request for Comments: 5681 TCP Congestion Control](#)

## Выводы

- **ЭВМ по сути** есть математические машины (**МСРПУ и ЭВМ!**). Они разрабатывались для ускорения вычислений по алгоритмам на идейной основе машины Тьюринга, под руководством математиков и согласно разработанной ими архитектуре.
- Основные достижения по развитию современных ИКТ получены математиками или учеными, владеющими математической культурой.

# Почему математическая культура полезна профессионалам ИКТ

- Математика — язык описания и исследования в ИКТ.
  - ▶ Формирует и развивает абстрактное логическое мышление.
  - ▶ Выбатывает навык регулярного подхода к решению задач.
  - ▶ Приучает не игнорировать “мелочи”.
- Абстракция — математика — информатика.
- Стремительное расширение сферы ИКТ — **инвестиции. Новые парадигмы, концепции, стандарты, инструменты, прикладные системы** появляются и внедряются очень быстро, дополняя и заменяя друг друга.
- Базовые математические идеи появляются **гораздо реже**. Пример — распараллеливание и не Неймановские архитектуры.
- Математическая культура позволяет профессионалу **быстро осваивать** любые современные и будущие концепции, методы и технологии ИКТ, обеспечивая ему **долговременную** карьеру.

# Определяющая роль математической культуры в ИКТ

## Важно владение именно математической культурой.

... доказательство, контрпример, необходимое и достаточное условие ...

**Математический анализ** — база ВСЕХ направлений математики.

Владение его основами и формирует математическую культуру и позволяет легко осваивать и применять нужные разделы математики:

- Дискретная математика (в т.ч. комбинаторика) — большинство задач, в т.ч:
  - ▶ Формальные языки — гсс, компиляторы компиляторов.
  - ▶ Графы, деревья, очереди, стеки и др.
  - ▶ Поиск, сортировка, кратчайшие пути, оптимизация, маршрутизация в компьютерных сетях ... !
  - ▶ Криптография, оптимизация.
- Линейная алгебра — Игры, вирт реальность.
- Теория вероятностей — финансы, Data Mining.
- Геометрия, дифф. уравнения — Машинная графика, игры.
- Линейная алгебра, мат. статистика — Криптография.

# Определяющая роль математической культуры в ИКТ

Из книги:

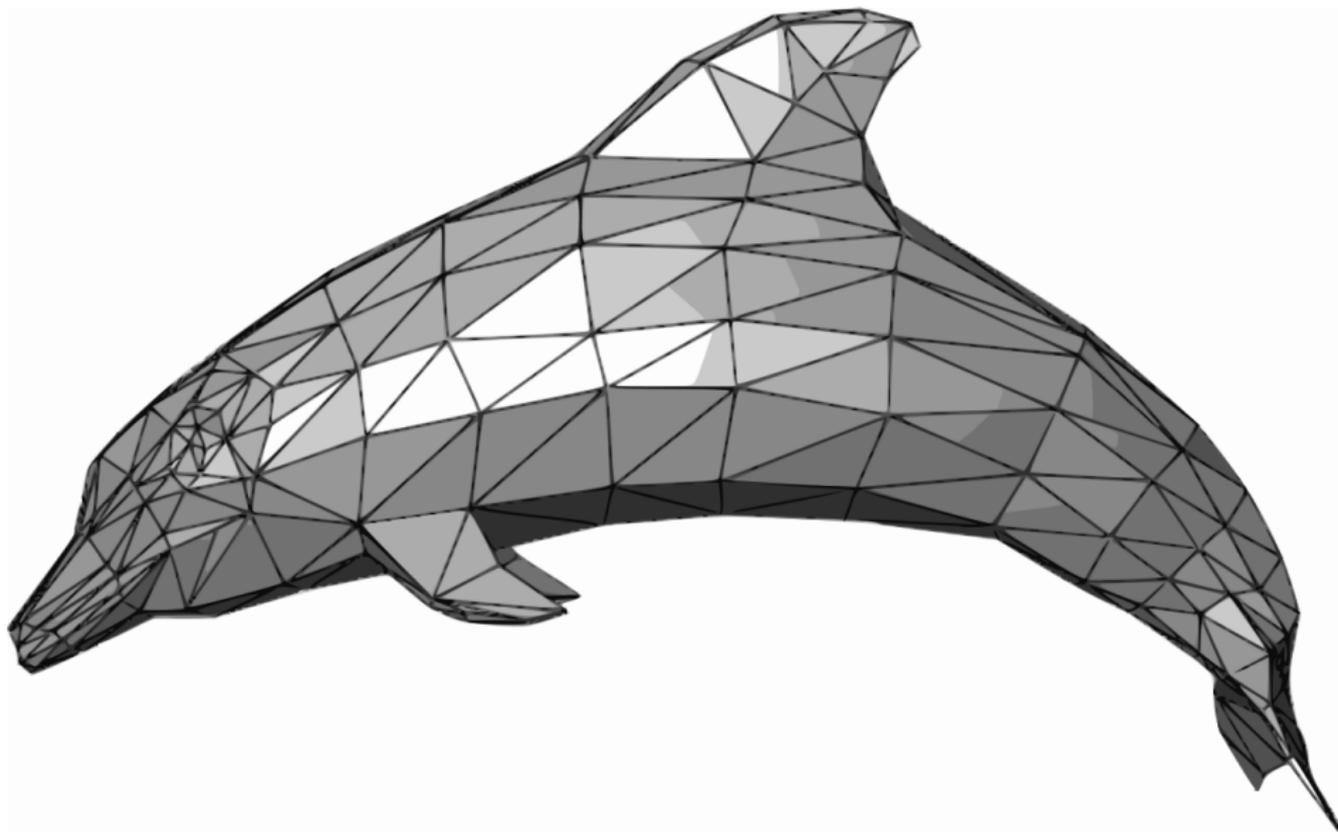
## Д. ПОЙА МАТЕМАТИКА И ПРАВДОПОДОБНЫЕ РАССУЖДЕНИЯ

### Математик, физик и инженер.

— Физик верит, — сказал математик, — что 60 делится на все числа. Он замечает, что 60 делится на 1, 2, 3, 4, 5 и 6. Он проверяет несколько других чисел, например 10, 20 и 30, взятых, как он говорит, наугад. Так как 60 делится и на них, то он считает экспериментальные данные достаточными.

— Да, но взгляни на инженера, — возразил физик. — Инженер подозревает, что все нечётные числа простые. Во всяком случае, 1 можно рассматривать как простое число, доказывает он. Затем идут 3, 5 и 7, все, несомненно, простые. Затем идет 9 — досадный случай; 9, по-видимому, не является простым числом. Но 11 и 13, конечно, простые. Возвратимся к 9, — говорит он, — я заключаю, что 9 должно быть ошибкой эксперимента.

# Модель изображения дельфина: треугольная сетка



# Как организовать учебную работу

Я слышу и забываю,  
Я вижу и запоминаю,  
Я делаю и понимаю.

Конфуций  
Мыслитель и философ  
5 в до н.э.

Профессионализм в ИКТ — **только** за счет большого объема работы по решению учебных задач.

- Учебный процесс интенсивный, **много нового**.
- **Работоспособность**: режим дня, сон, питание, физкультура.  
Коты VK.
- **Меньше** прямого контроля, **больше** личной ответственности.

# Как организовать учебную работу

## ■ Лекции.

- ▶ Активное восприятия — залог успеха в сессии, следить за ходом лекции, **задавать вопросы**.
- ▶ Конспект — обязательно, записывать новое, главное, перечитать перед следующей лекцией.

## ■ Лабораторные и практических занятия.

- ▶ Сдавать в срок, иметь конспекты и учебники.
- ▶ Иметь план работы - схему программы, тестовые задачи и т.п.
- ▶ Настроить среду (предоставляется кафедрой ИМО) для разработки и отладки программ в месте проживания.
- ▶ **Прояснять все непонятное, не откладывая, на консультациях у преподавателей (будет расписание) или по эл.почте .**

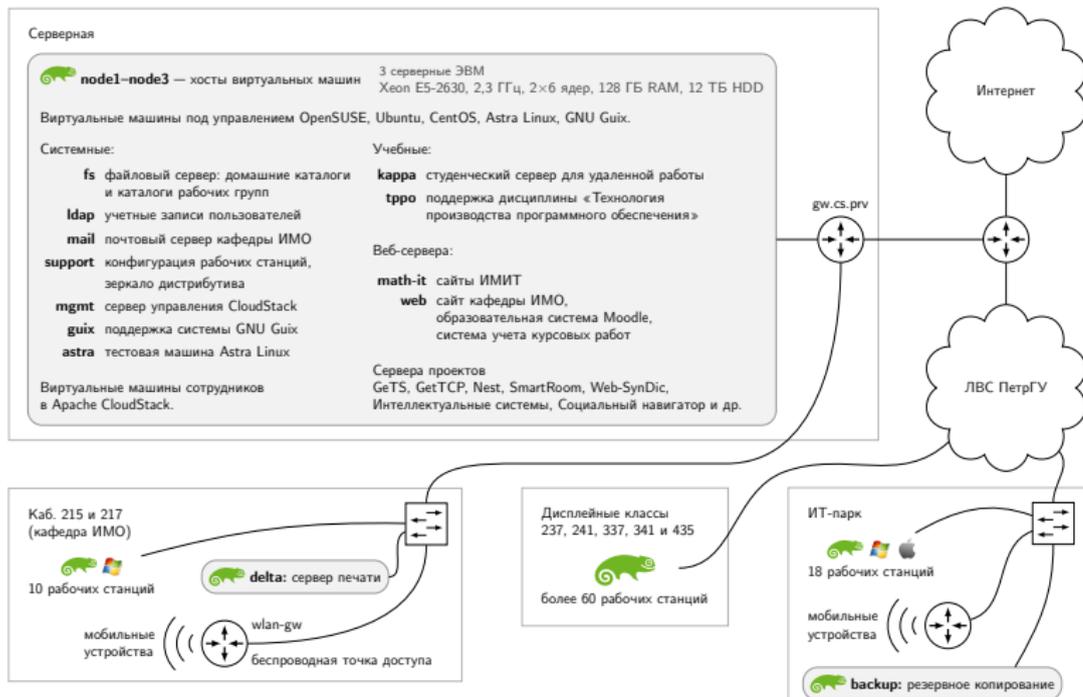
# Современные учебные планы

- Бакалавриат с 1993 г..
- Магистратура с 1997 г..
- Аспирантура — давно.
- ФГОС + Computing Curricula, регулярно публикуемые Association for Computing Machinery — ACM и Institute of Electrical and Electronics Engineers Computer Society — IEEE CS.
- Компетенции ИКТ = Математическая + Инженерная (технология разработки ПО).
- ИМИТ поддерживает УП на современном уровне.
- История развития учебных планов. Anatoly Voronin, Iurii Bogoiavlenskii, Vladimir Kuznetsov "Perspectives on the emergence of computing programs propelled by local industry in Russia", ACM Inroads, December 2015, Volume 6, No. 4.

# Информационно-вычислительная инфраструктура ИМИТ (ИВИ)

<http://cs.petrstu.ru/facilities/system.pdf>

Информационно-вычислительная инфраструктура Института математики и информационных технологий  
Поддерживается полный набор инструментов для учебного процесса и НИР/ОКР студентов и сотрудников (около 600 пользователей).



# Информационно-вычислительная инфраструктура ИМИТ (ИВИ)

- Развитая, современная.
  - ▶ ОС Linux для студентов — готовый образ — запускается в виртуальной машине (работает как приложение в вашей ОС).
  - ▶ Краткое руководство по Emacs (применяется и как IDE).
  - ▶ Использование отладчика.
  - ▶ Система видео-конференц-связи Jitsi Meet.
  - ▶ Система мгновенного обмена сообщениями Matrix.
- Поддерживается сотрудниками кафедры ИМО — В. А. Пономарев, М. А. Крышень, В. М. Димитров, Д. Б. Чистяков.
- Предоставляет современные программные продукты и технологии для учебного процесса, исследований и разработок.
- <http://cs.petrSU.ru/facilities/index.php.ru> — Описание ИВИ

# ОС UNIX → Linux

- Разработана “Программистами для программистов”.
- Переносимая, многозадачная, многопользовательская.
- Первая ОС с реализацией набора протоколов TCP/IP.
- Первая ОС на ЯВУ. Более 95% кода реализовано на языке C.
- Для переноса на другую архитектуру достаточно разработать кросс компилятор.
- Перенесена на много архитектур - Интернет вещей, котлы ТЭЦ.
- Надежная - сервера эл.почты работают без перезагрузки месяцами.
- UNIX → MS DOS → Windows - Subsystem for Linux → Linux от Microsoft
- Linux — лидер на рынке:
  - ▶ серверных ЭВМ;
  - ▶ смартфонов, планшетов;
  - ▶ высокопроизводительных вычислений (доля 99.4%.);
  - ▶ ОС сетевого оборудования (SOHO-роутеры, энтерпрайз-решения (Cisco));
  - ▶ ОС Аврора, бывшая Sailfish Mobile OS RUS;

# Программирование "в малом". Язык С на все времена.

- Эффективный машинный код. Используется вместо ассемблера.
- Применяется при разработке как ОС (UNIX), так и прикладного ПО для платформ от суперЭВМ до встроенных.
- Синтаксис и структуры управления С **СТАЛИ КЛАССИКОЙ** и заимствуются языками C++, C#, Unix's C shell, D, Go, Java, JavaScript, Limbo, LPC, Objective-C, Perl, PHP, Python, Rust, Swift и др.
- Компиляторы и интерпретаторы языков программирования, СУБД и пр. реализуются, как правило, на С.
- Реализован для большинства аппаратных платформ и ОС.
- Стандартизован – ANSI + ISO. Обеспечивает кросс платформенную разработку.
- Регулярно занимает 1/2 места по популярности в [TIOBE Index](#)
- Библиотека `libc` — около 1500 базовых функций.
- **С – фундаментальная основа квалификации программиста.**

# Командный интерфейс против графического

- Письменность. Иероглифы против алфавита.
- Command Line Interface — CLI — интерфейс командной строки для взаимодействия с ОС.
- Популярный — `bash` — свободная версия Bourne shell.

Простые примеры:

- Цикл в командной строке `bash`:  

```
for d in *.txt ; do mv $d $d.utf8 ; done
```
- Простой скрипт в файле `k-m` — монтировать (присоединить) домашний каталог на удаленном сервере `kappa` в каталог `KAPPA` на локальной ЭВМ):  

```
#!/usr/bin/sh  
cd  
sshfs kappa.cs.karelia.ru: KAPPA
```

## Командный интерфейс — конвейер команд

Символ | — соединение в конвейер.

Конвейер (pipe) — однонаправленный канал взаимодействия команд

```
bash du -k -c -d 1|sort -n -r|less
```

- **du - disk usage**: **-k**: блоки по 1k,, **-c**: получить суммарный объем каталога, **-d 1**: для подкаталогов ниже на 1 уровень;
- **sort** - сортировать строки: **-n**: числовая сортировка, **-r**: в обратном порядке.

```
2890548 total
2890548
2765996  ./Lectures
84592   ./Intel-guides
9736    ./GCC-guide
5528    ./My-tests
4100    ./GDB
28      ./Coding-Style
12      ./Voprosy-k-zachetu
8       ./Profilers
lines 1-10/10 (END)
```

## Командный интерфейс: сложная работа с файлами

Дано: 150 УМКД дисциплин кафедры, каждый в 5 файлах lowriter.

Нужно: каждый из них перевести в pdf и объединить в один pdf.

```
#!/bin/bash
# M. A. Kryshen. Getting pdf files of UMKD
# Target directory
base="$(pwd)"
# For each directory
find "$source" -type f | xargs -d '\n' -n 1 dirname | sort -u | \
  while read name; do
    echo "$name"
    dest="$base${name:${#source}}"
    mkdir -p "$dest"

    libreoffice --headless --convert-to pdf --outdir "$dest" "$name"/*

    # Titele sheet to the begin
    mv "$dest"/*titul_list*.pdf "$dest"/000_titul_list.pdf
    # Adding empty page after title
    ln empty.pdf "$dest"/001_empty.pdf
    pdffute "$dest"/*.pdf "$dest".pdf
    rm -f "$dest"/*.pdf
    rmdir "$dest"
done
```

# От учебы к профессиональной карьере

- [https://imit.petrso.ru/files/2018/f\\_248\\_dubna.pdf](https://imit.petrso.ru/files/2018/f_248_dubna.pdf)  
А. В. Воронин Развитие ИТ - технологий в Петрозаводском государственном университете.
- <https://imit.petrso.ru/employers>  
Работа в сфере ИКТ и ПО. Компании партнеры ИМИТ.
- <https://imit.petrso.ru/page/vacancy>  
Предложения компаний-партнеров.

# Магистры выпуска 2011 г.



Спасибо за внимание.

*"... Важнейшим из всех искусств является для нас искусство программирования на языке Ассемблера."*

*Ю.А.Богоявленский  
"О грядущих катастрофах"*

**ДРАМА**

**НА**

**СИСТЕМНОЙ**

**МАГИСТРАЛИ.**

(с) зр.22302, 1991г.

