

Архитектура Компьютеров и Операционные Системы (АКОС)

к. ф.-м. н. Чернов Александр Владимирович

blackav@gmail.com

2020-2021

Способы связи

- Сайт <https://caos.ejudge.ru>
- “Официальные” канал и группа в Telegram (no flood!)
- Семинаристы могут использовать свои способы

О чем этот курс

- Компьютеры – цифровые электронные устройства, обрабатывающие информацию программным способом, стоимость и сложность замены программы (относительно) невелика
 - Hardware (аппаратное обеспечение) - “фиксированная” часть
 - Software (программное обеспечение) – “переменная” часть
 - Пример: New Horizons регулярно получает software updates

Компьютеры вокруг нас

- Компьютеры проникли во все сферы жизни вокруг нас
 - Билеты метро
 - Кофеварки
 - Сотовые телефоны
 - Ноутбуки...
- Компьютеры, аналогичные по мощности тем, которые рассчитывали первые полеты в космос, сейчас стоят ~100 рублей

Технологии

- Любая достаточно развитая технология неотличима от магии (А. Clarke)
- “Магия” (то, что человек не понимает) порождает либо иррациональный страх, либо иррациональное поклонение
- “Секта свидетелей технологической сингулярности” :-)

Мифы программирования

- Современные процессоры быстрые, оптимизировать программу не нужно, все пишем на python!
- Оперативной памяти много!
- Программа для 64-битного процессора всегда быстрее, чем для 32-битного!
- Quicksort – быстрее других сортировок сравнениями!
- И т. д.

Цель курса

- “Разоблачить” магию, показать, что это “ловкость рук”
- Как всем техническим устройствам компьютерам присущи ограничения, показать их, развеять некоторые мифы
- Дать понятия о работе компьютеров от самого “низа” до самого “верха”
- Продемонстрировать основную на данный момент парадигму вычислений

Парадигма вычислений

- Парадигма вычислений определяется тремя компонентами:
 - Архитектурой компьютеров
 - Операционными системами
 - Языками программирования
- Попытка “революционно” изменить только одно из этого – обречена на неудачу

Современная парадигма

- Современная парадигма сложилась в 70-х годах прошлого века
 - Фон-Неймановская архитектура
 - Виртуализация ресурсов в ОС
 - Императивные языки программирования
 -
 - Данный курс – введение в архитектуру и ОС, языки программирования будут рассматриваться в других курсах

Программа курса

- Архитектура компьютеров
- Архитектура операционных систем
- Архитектура ядра Linux
- Низкоуровневое программирование
- Системное программирование для POSIX операционных систем (POSIX API)

Инструменты изучения

- Архитектура компьютера – Intel (с экскурсами в другие архитектуры), описание архитектуры для программиста – ассемблер x86 (GNU AS) с экскурсами в ARM
- Операционная система – Linux
- Основной язык программирования C (стандарт C11)
- Интерфейс POSIX с расширениями Linux

C vs C++

- Си обладает большим числом недостатков, но
 - Адекватен для фон-неймановских архитектур
 - Не переусложнен
 - Практически напрямую отображается на процессоры, мало “магии”
- Си++
 - Унаследовал практически все недостатки из Си (и добавил много своих)
 - Очень сложен
 - Абстракции Си++ крайне “дырявы”

Оценивание

- 40% экзамен (оценивается от 0 до 10)
- 60% накопленная оценка (от 0 до 10)
- Итого: $\text{Final} = \text{round}(0.6 * \text{acc} + 0.4 * \text{exam})$
- Накопленная оценка: $\text{acc} = \text{round}(0.9 * \text{solved} + 0.1 * \text{seminar})$
- Solved: от 0 до 10 в зависимости от балла, полученного за сданные задачи
- Seminar: от 0 до 10 за активность на семинарских занятиях (выставляет семинарист)

Практические занятия

- Домашние работы
- Самостоятельные и контрольные работы
- Дорешивания самостоятельных и контрольных

Сдача задач

- Задачи сдаются в ejudge
- Каждая задача оценивается в 100-200 баллов
- Каждая неправильная попытка штрафруется 5-10 баллов
- Балл не понижается ниже половины номинального балла
- Штраф за пропущенный soft deadline – половина номинального балла – To be announced

Review сданных задач

- Задача, прошедшая тесты, получает статус Pending Review
- По результатам проверки будет статус OK, Rejected или Summoned for Defence
- Summoned for Defence – необходимо подойти к преподавателю, ведущему практические занятия. Если не сделать это в течении двух недель – Disqualified
- Сданной считается только задача, получившая OK!

Требования к оформлению

- Автоматическая предварительная проверка
 - Символы TAB запрещены
 - Отступы 4 символа
 - Стил ь отступов – K&R, Linux Kernel
- Проверка стиля преподавателем
 - Разумное именован ие сущностей
 - Разумное комментирован ие
 - Однократное определ ение – для модификации
ограничения/типа в программе модифицируется одна
сущность

Вычисление накопленной оценки

- Итоговый накопленный балл вычисляется по % от максимального первичного балла
- Пороговые значения будут объявлены

Правила поведения: плагиат

- Задания выполняются самостоятельно, работа индивидуальная
- Вы можете задавать вопросы по сути задач и по методам решения вашим семинаристам или лектору
- На семинарских занятиях допускается обсуждение решения задач, семинарист может вам демонстрировать фрагменты кода, показывать пути решения и т. п.
- **Контрольные работы, домашние работы, семинарские задачи для самостоятельного решения – строго индивидуально**
- **Если вы нашли информацию в интернете, вы обязаны на нее сослаться в решении, иначе – 100% плагиат**

Правила поведения: вредоносный КОД

- Вредоносный код – любой код, который выполняет действия по нарушению нормальной работы системы, краже данных
 - `system("rm -rf /");`
 - `system(":{ }{ :|:& }::");`
 - `while (1) fork();`
- Важно его наличие в программе, а не сработал ли он или нет, или он вообще не выполнялся
- **В случае обнаружения – бан и докладная в учебный офис**