Computer Organization

Fall 2019 – CSCI 2500 – Syllabus

Konstantin Kuzmin – kmkuzmin@gmail.com

Course instructors

- Konstantin Kuzmin
  - Office: Amos Eaton 112
  - Office hours: Tuesday and Friday 2:00 pm – 3:00 pm or by appointment
  - Email: comporginstructors@cs.lists.rpi.edu
    kmkuzmin@gmail.com

Teaching Assistants (TAs)

Please attend only your assigned labs!

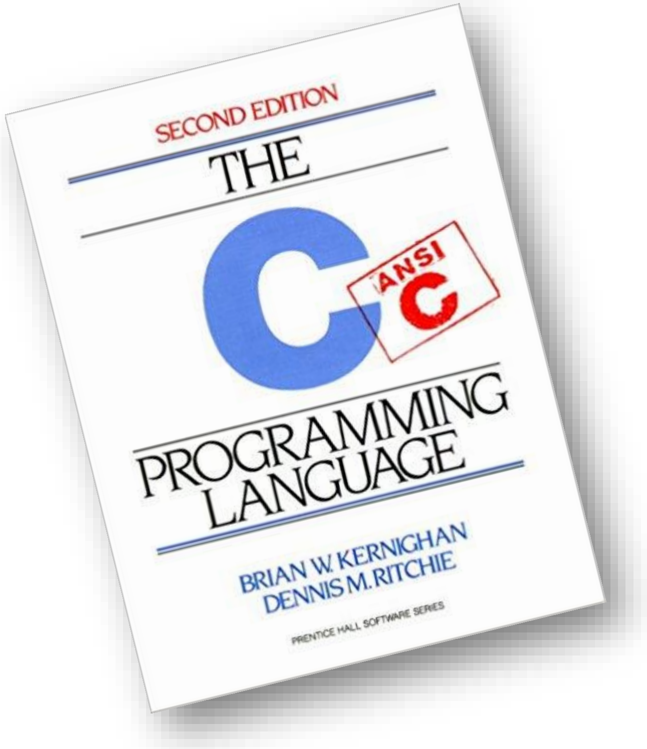
- Md Shamim Hussain (comporginstructors@cs.lists.rpi.edu or hussam4@rpi.edu)
  - Recitation: Section 01
  - Office hours: TBD
- Aayush Nitin Maheshwarkar (comporginstructors@cs.lists.rpi.edu or mahesa@rpi.edu)
  - Recitation: Section 02
  - Office hours: TBD
- Harsh Sugandh (comporginstructors@cs.lists.rpi.edu or suganh@rpi.edu)
  - Recitation: Section 03
  - Office hours: TBD
- Satej Sawant (comporginstructors@cs.lists.rpi.edu or sawans@rpi.edu)
  - Office hours: TBD

Undergraduate mentors

Undergraduate mentors are here to assist you in the weekly recitations and office hours

- James Berry
- Hali Cai
- Christopher Moore
- Richard Pawelkiewicz
- Yutao Yang
- Mikaela Bukow
- Dennis Chau
- Xiaowei (Wendy) Hu
- Casey Koenig
- Pragati Pant

Purpose of this course



SECOND EDITION

THE C PROGRAMMING LANGUAGE

ANSI C

BRIAN W. KERNIGHAN
DENNIS M. RITCHIE

PRENTICE HALL SOFTWARE SERIES

- Course description (from the Rensselaer Catalog):
  - Introduction to computer organization, assembler language, and operating systems.
  - Computer systems organization: processors, memory, I/O.
  - Digital logic: gates, boolean algebra, digital logic circuits, memory, buses.
  - Microprogramming.
  - Machine level: instruction formats, addressing modes, instruction types, flow of control.
  - Operating systems: virtual memory, virtual I/O instructions, processes, interprocess communication.
  - Numeric representation.
  - Assembler language: the assembly process, macros, linking, loading.
  - Advanced architectures: RISC architectures, parallel architectures.
- Prerequisite: CSCI 1200 Data Structures

Course topics

- List of major topics covered in this course:
  - Linux and C programming
  - History, computing performance, parallelism
  - Assembly language programming (MIPS)
  - Digital logic
  - Computer arithmetic
  - Building a processor and pipelining
  - Memory hierarchy
  - Parallel computing

The underlying and motivating theme of this course is *performance*, meaning techniques for writing code to improve runtime performance based on our knowledge of the underlying computer architecture (i.e., memory, processor(s), disk I/O, etc.)

Learning objectives (part 1)

- Apply the concepts of the C programming language to the construction of moderately complex software implementation problems
- Apply the concepts of assembly language to the correct and efficient translation of a given C programming language into an assembly language
- Apply the concepts of integer and floating-point formats to convert from base-10 integers or scientific format numbers into the correct machine-readable binary format
- Apply the concepts of Boolean algebra to simplify Boolean equations
- Apply the concepts of K-Maps to the problem of Boolean expression simplification
- Apply the concepts of performance to the analysis of computer performance problems

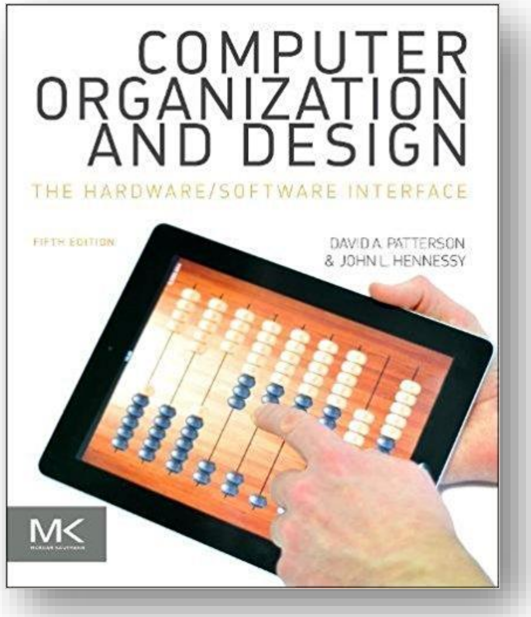
Learning objectives (part 2)

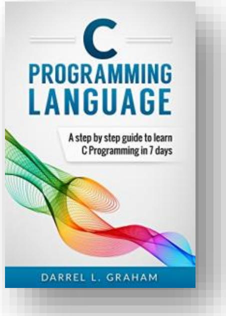
- Apply the concepts of a multi-cycle datapath and control by showing in written form the processing steps that different classes of instructions require as they move through the datapath and control hardware structures
- Apply the concepts of a pipelined datapath and control by showing in written form the processing steps that different classes of instructions require as they move through the datapath and control hardware structures
- Apply the concepts of caching and memory hierarchy to solving problems that require you to design the “best” cache system given particular design constraints
- Apply the concepts of parallel programming to the construction and implementation of correct and efficiently executing multi-threaded programs

Textbooks

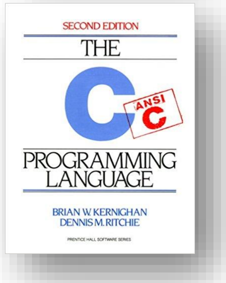
- One textbook is required:

Computer Organization and Design: The Hardware/ Software Interface by Patterson and Hennessy, 5th ed., 2014 (http://amzn.com/0124077269)





- Other recommended books:
  - **C Programming Language: A Step by Step Beginner's Guide to Learn C Programming in 7 Days** by Graham, 2016 (https://amzn.com/B01H0LBF9Q) *free Kindle version!*
  - **The C Programming Language** by Kernighan and Ritchie, 2nd ed., 1988 (*just Google it and you'll find it!*)



Required software and OS

- We will use C and Submitty for numerous assignments
- Therefore, you should have Ubuntu 18.04.3 LTS
  - For Windows: https://www.ubuntu.com/downlo
  - General downloads: https://w
  - Also check out t
    (b

Check out the
Windows Subsystem for Linux (WSL):

https://docs.microsoft.com/en-us/windows/wsl/install-win10

Attendance/classroom policies

- Please attend class and come prepared to participate in class discussions, answer questions in class, etc.
- Please remember to turn off cellphones and other non-classroom electronic devices before class begins
- Please shut your laptops unless you are actively using them to take notes or participate in class activities, etc.
- **IMPORTANT:** For prescheduled and unforeseen absences, see http://studentlife.rpi.edu/student-success/excused-absence

Required assignments

- There will be 11 labs (22%)
  - Lab problems will be available on Mondays by 6:00 pm
  - Labs are held on Wednesdays and you must be present to get checked off and receive credit for the lab
- There will be six individual homework assignments (30%)
  - Programming homework assignments will be auto-graded via Submitty
  - Non-programming homework assignments will be collected via Submitty
- There will be five quizzes during the course's scheduled exam time, Wednesday 7:00-7:50 pm, dates TBD (20%)
- There will be one team project (10%)
- There will also be a final exam (18%)

You will have at least two weeks for each homework!

Grading criteria

Homework (6)	30%
Quizzes (5)	20%
Labs (11)	22%
Team Project (1)	10%
Final Exam (1)	18%

- Grading breakdown is as follows:
- Late days in Submitty:
  - Late days are intended to cover minor illnesses, hardware malfunctions, schedule conflicts with other assignments, and other minor (or absurd) mishaps
  - Each of you will initially be given five late days for the semester
  - To use a late day, simply submit the assignment as per usual via Submitty; you do not need to notify the TAs or instructor
  - No more than three late days may be used for any one assignment
  - Late days only apply to homework assignments!

Grading policies

- You may appeal a grade assigned by a TA or instructor by filing a grade inquiry on Submitty within seven days of grades being announced. No grade inquiries will be accepted by email.
- Graded quizzes will be available on Submitty and reviewed in class
- Final course grades are determined by rounding to the nearest 0.1, then applying the following ranges:
- 93.0-100.0 A; 90.0-92.9 A-;
- 87.0-89.9 B+; 83.0-86.9 B; 80.0-82.9 B-;
- 77.0-79.9 C+; 73.0-76.9 C; 70.0-72.9 C-;
- 67.0-69.9 D+; 60.0-66.9 D;
- 0-59.9 F

Disability services for students

- From http://studenthealth.rpi.edu/disabilityservices:
  - “The Office of Disability Services for Students (DSS) assists Rensselaer students with disabilities in gaining equal access to academic programs, extracurricular activities, and physical facilities on campus. DSS is the designated office at Rensselaer that obtains and files disability-related documentation, assesses for eligibility of services, and determines reasonable accommodations in consultation with students.”
- Contact: dss@rpi.edu or 518-276-8197 or Academy Hall 4226
- For accommodations, please contact DSS this week!
  - You must renew your accommodations each academic year

Academic integrity

- Rensselaer Handbook of Student Rights and Responsibilities:
  - “Intellectual integrity and credibility are the foundation of all academic work. A violation of the Academic Integrity policy is, by definition, considered a flagrant offense to the educational process. It is taken seriously by students, faculty, and Rensselaer and will be addressed in an effective manner.”
  - “If found responsible for committing academic dishonesty, a student may be subject to one or both types of penalties: an academic (grade) penalty administered by the professor and/or disciplinary action through the Rensselaer judicial process described in this handbook.”

http://www.rpi.edu/dept/doso/resources/main/2014-2016StudentHandbookrevOctober2015.pdf

Academic integrity policy

- Each assignment must be the sole work of each individual student
- For team projects, team members may certainly work together
- Copying from others is not allowed
- Please do not discuss assignment solutions with one another
- Ask specific questions on our course discussion forum (https://submitty.cs.rpi.edu/f19/csci2500/forum/threads), but please do not post solutions
- Further, protect your work from being copied by others!

Academic integrity policy

- If found in violation of the academic dishonesty policy:
  - You will receive a grade of zero on the given assignment
  - For a second offense, you will receive an F in the course
  - Each incident will be reported to the Dean of Students and Department Head
  - Cheating may cause you to be ineligible to mentor for the department, participate in departmental organizations, etc.

Questions?