

CISC 260 Machine Organization and Assembly Language

Spring 2019

Time & Place: 3:30PM-4:45PM, TR, Kirkbride Hall 004

Instructor: Li Liao (Smith 424, 831-3500, liliao@udel.edu), Office Hours: 2:00PM-3:00PM
Tuesdays and Thursdays or by appointment

TA: Md Mottalib (Smith 203, mmmdip@udel.edu), Office Hours: 12:00PM – 1:00PM,
Wednesdays and Thursdays.

Course Catalog Description:

Introduction to the basics of machine organization. Programming tools and techniques at the machine and assembly levels. Assembly language programming and computer arithmetic techniques.

Detailed Description:

In essence, this course serves to reveal the inner workings of modern digital computers: how computation is done inside the computer hardware, what are the main components of the hardware, how and why they are organized in certain ways, and what are the tradeoffs for different designs and their impact on programming and performance.

This course prepares you with necessary concepts and skills to learn more advanced topics in computer architecture, compiler, and operating systems. It helps you gain deeper understanding of issues arisen from programming in high level languages, such as security loopholes susceptible for virus attacks. It also touches on fundamental issues in computer science: what makes a computer much more powerful and versatile? Is there any problem that a computer cannot solve even if given sufficiently large memory and fast CPU?

Major topics covered include:

- Digital representation of information: decimal, hexadecimal, binary, ASCII
- Arithmetic in binary, two's complement
- Combinational and sequential logic, ALU
- Control and datapath
- Instructional Set Architecture (ISA)
- Machine language and assembly language
- Stacks and procedure calls
- Performance issues and optimization

As the specific goals of this course, the students should be able to

- explain the basic organization of a classical von Neumann machine and its major functional units
- explain how machine code is formatted/organized and executed via the corresponding functional units
- write simple assembly language program segments
- demonstrate how fundamental high-level programming constructs, such as loops, procedure calls and recursions, are implemented at the machine and assembly language level
- convert numerical data between different formats
- carry out basic logical and arithmetic operations
- understand memory management (cache) and basic I/O
- understand performance issues and optimization techniques

Required Text: Harris & Harris, *Digital Design and Computer Architecture*, **ARM Edition**.

Assignments and Grading:

There will be 6 homework assignments. All late assignments are subject to 10% penalty per 24 hours past the due time (Saturdays and Sundays do not count), and are not accepted one week past due time. Homework submission is handled electronically via Canvas.

There will be one midterm exam and one final exam. No makeup exams will be granted except when officially acceptable excuses are presented.

All questions about a grade must be presented in writing to the instructor/TA within 1 week since the graded assignment is returned to the class. Afterwards, all grades become final.

Homework 42%, midterm exam 18%, final exam 35%, and participation + quizzes 5%

Policy on Academic Dishonesty:

The assignments in this class should be performed individually. You are permitted to discuss with other students on any conceptual problems but not for concrete code and implementation on any programming assignments. The work handed in must be entirely your own. Any evidence of academic dishonesty will be handled as stated in the Official Student Handbook of the University of Delaware. If you are in doubt regarding the requirements, please consult with me before you complete any requirement of this course.