جامعة نيويورك أبوظبي
NYU | ABU DHABI

CS-UH-2010
Computer Systems Organization

Credits: **4**

Prerequisites: Data Structures (CS-UH 1050)

Corequisites: None

Faculty Details	Instructor	Teaching Assistant
Name	Azzam Mourad	Khalid Mengal
Email	azzam.mourad@nyu.edu	kqm1@nyu.edu
Telephone		
Workspace	A2-195	
Office Hours	Mon/Wed: 11:15AM-12:30PM Or By Appointment (In-Person and via Zoom) Zoom Link: https://nyu.zoom.us/j/5911288690	Wednesday 2:00-3:00PM

Course Details	Day/Time	Location
Lecture	Monday/Wednesday Section 1: 9:55AM-11:10AM Section 2: 3:35PM-4:50PM	Section 1: Campus Center (C2-E049) Section 2: Campus Center (C2-W005)
Mid Term Exam	October 12 (Week 7)	TBD
Final Exam	Final's Week (TBC, Dec. 15-20)	TBD

Course Description

This course counts toward the following NYUAD degree requirements:

- Majors > Computer Science
- Majors > Electrical Engineering
- Majors > General Engineering

The course focuses on understanding lower-level (closer to hardware) issues in computer design and programming. The course starts with the C programming language, down to assembly and machine-level code, to basic operating system concepts. Students learn to read assembly code and reverse-engineer programs in binary. Topics covered include: C programming language, data representation, machine-level code, memory organization and management, performance evaluation and optimization, and concurrency.

Course Learning Outcomes (CLOs) and Link to Program Learning Outcomes (PLOs)

Students who successfully complete this course will be able to:	Linked to CS Major PLOs
1. Write programs in C	CS 1
2. Analyze performance and optimize programs	CS 3, CS 4
3. Read and analyze the assembly of Intel x86 processor family	CS 1, CS 2
4. Demonstrate ability to reverse engineer Assembly code	CS 2
5. State how data types are represented and stored	CS 1, CS 2
6. Explain memory architecture and hierarchy	CS 1, CS 2
7. Describe virtual memory and the fundamentals of linking	CS 1, CS 2, CS 3, CS 4
8. Explain exceptional control flow	CS 1

Teaching Methodologies

1. **Lectures**: In class lectures will cover the fundamental concepts. Most lectures have interactive in-class exercises and discussions and you are expected to participate.
2. **Readings**: The course schedule lists sections of the course textbook that students should read prior to class. By reading the textbook before the lecture, you can better use the lecture: you can clarify concepts you found difficult to understand and you can better participate in class discussions and exercises.

3. **Individual Programming.** Lab assignments will be an opportunity for students to master concepts by completing a series of programming assignments.

Graded Activities

Activity Detail	Grade Percentage	Week	Linked to Course Learning Outcome(s)
Quizzes	10%	-	
Assignment 1	5%	Sep. 21	CLO 1
Assignment 2	5%	Oct. 5	CLO 5
Assignment 3	10%	Nov. 14	CLO 3, CLO 4
Assignment 4	10%	Dec. 5	CLO 3, CLO 4
Participation	5%	-	-
Mid-Term Exam	25%	Oct. 12	-
Final exam	30%	Final's week	-

Assignments and Grades:

Homework and Programming Assignments

There will be programming assignments every 3-4 weeks. The programs must run without compiler or runtime errors and are submitted in the designated format and submission procedure. Students will have 2 grace days that they can consume in any one of the assignments. Each grace day will protect the student from being penalized if they are late by one day. The grace days can be consumed in separate assignments or in the same assignment. If the submission is late, and if the student has consumed all of their grace days, 10% will be deducted from the assignment grade per each late day. No submission will be accepted 3 days after the due date.

You must work alone on the programming assignments. You are encouraged to answer each other's curiosities, but refrain from explicitly giving away solutions. You may use online resources for help, but verbatim copying code is considered cheating.
Import: Measurement Of Software Similarity (MOSS) system will be used in the course to check plagiarism, copy, and cheating.

Quizzes: There will be 2 quizzes given during classes.

Assignments: There will be 4 programming assignments and many reading tasks.
Exams: There will be one midterm exam and one final exam. Questions about grading must be discussed within 2 days after any grades are returned.

Grading policy
2 Quizzes, 4 Assignments, Mid-term Exam, Participation, Final Exam.

Required Bookstore Texts

- Computer Systems – A Programmer's Perspective, by *Randal Bryant and David O'Hallaron.* 3rd Edition, Pearson, 2015. *(ISBN: 978-0131103627)*. *An electronic edition of the book will be available.*
- The C programming language, by *Brian W. Kernighan, Dennis M. Ritchie.* 2nd Edition, Pearson, 1988. *(ISBN: 978-0134092669). An electronic edition of the book will be available.*

Academic Policies

Attendance: Attendance for this course is mandatory. Every absence needs to be agreed with the professor prior to class.

Class Participation: Students are expected to participate in the class activities. These activities include: (1) participation in discussions during the class; (2) solving in-class exercises; (3) answering questions related to the readings. It is sufficient to participate in one category among the above categories (this depends on the student preferences).

Criteria/rubric for satisfactory participation are as follows: *(1) frequency of participation* (how many times does the student participate per session)*; (2) quality of participation* (are the comments constructive to the sessions and do they show a deep understanding of the lecture materials). Half of the participation grade is for the *frequency* while the other half is for the *quality* of participation. The frequency and quality of the student participation are evaluated in comparison to the average participation in the class. Showing correct solutions for in-class exercises is not required, instead the student needs to show an effort in solving the exercises and a draft of a solution.

Grade Distribution: Students need to obtain a grade of C or better to count the course towards their intended degree for required courses or electives. Course percentages will be translated into letter grades based on these intervals:

A	A-	B+	B	B-	C+	C	C-	D+	D	F
[95]	[90)	[87)	[83)	[80)	[77)	[73)	[70)	[67)	63	[0)

Grades are not curved.

Integrity

At NYU Abu Dhabi, a commitment to excellence, fairness, honesty, and respect within and outside the classroom is essential to maintaining the integrity of our community. By accepting membership in this community, students, faculty, and staff take responsibility for demonstrating these values in their own conduct and for recognizing and supporting these values in others. In turn, these values create a campus climate that encourages the free exchange of ideas, promotes scholarly excellence through active and creative thought, and allows community members to achieve and be recognized for achieving their highest potential.

Students should be aware that engaging in behaviors that violate the standards of academic integrity will be subject to review and may face the imposition of penalties in accordance with the procedures set out in the NYUAD policy:
https://students.nyuad.nyu.edu/campus-life/student-policies/community-standards-policies/academic-integrity/.

Mental Health Awareness

As a University student, you may experience a range of issues that can interfere with your ability to perform academically or impact your daily functioning, such as: heightened stress; anxiety; difficulty concentrating; sleep disturbance; strained relationships; grief and loss; personal struggles.

If you have any well-being or mental health concerns please visit the Counseling Center on the ground floor of the campus center from 9am-5pm Sunday - Thursday, or schedule an appointment to meet with a counselor by calling: 02-628-8100, or emailing:
nyuad.healthcenter@nyu.edu.

If you require mental health support outside of these hours, call NYU's Wellness Exchange hotline at 02-628-5555, which is available 24 hours a day, 7 days a week.

You can also utilize the Wellness Exchange mobile chat feature, details of which you can find on the student portal. If you need help connecting to these supports please contact me directly.

Course Schedule

Below is a provisional schedule of the topics to be covered in this course. Dates are subject to confirmation and may change. Please make sure to go over the readings before coming to class.

Week	Session	Date	Topic	Reading	Other
1	1	Aug. 29	Course Overview	/	
1	2	Aug. 31	C Programming (1)	[KR] 1,2	

2	3	Sep. 5	C Programming (2)	[KR] 3,4	
2	4	Sep. 7	C Programming (3)	[KR] 5,6	Assignment-1 Released
3	5	Sep. 12	Information Storage (Bits & Bytes)	[BO] Ch. 2.1	
3	6	Sep. 14	Boolean Algebra and Bit Manipulation		
4	7	Sep. 19	Integers	[BO] Ch. 2.2-2.3	
4	8	Sep. 21	Floating Point	[BO] Ch. 2.4	Assignment-1 Due Assignment-2 Released
5	9	Sep. 26	Machine Progr. Basics and Arithmetic	[BO] Ch. 3.1-3.5	Quiz
5	10	Sep. 28	Machine Progr. Basics and Arithmetic		
6	11	Oct. 3	Machine Progr. Control and Loops	[BO] Ch. 3.6	
6	12	Oct. 5	Machine Progr. Control and Loops		Assignment-2 Due
7	13	Oct. 10	Mid-term review		
7	14	Oct. 12	**Mid-term**		
9	15	Oct. 24	Machine Progr. Procedure		[BO] Ch. 3.7
10	16	Oct. 26	Machine Progr. Procedure		Assignment-3 Released
10	17	Oct. 31	Machine Progr. Data (Arrays & Struct)	[BO] Ch. 3.8, 3.9	
11	18	Nov. 2	Machine Progr. Data (Arrays & Struct)		
11	19	Nov. 7	Machine Progr. Advanced, Buffer Over-flow	[BO] Ch. 3.10	
12	20	Nov. 9	Machine Progr. Advanced, Buffer Over-flow		
12	21	Nov. 14	Machine Progr. Advanced, Buffer Over-flow		Assignment-3 Due Assignment-4 Released

13	22	Nov. 16	Memory Hierarchy & Cache Memories	[BO] Ch. 6.1-6.3, 6.4-6.7	Quiz
13	23	Nov. 21	Program Optimization	[BO] Ch. 5	
14	24	Nov. 23	Virtual Memory	[BO] Ch. 9.1-9.6.3	
14	25	Nov. 28	Memory Allocation	[BO] Ch. 9.9	
15	26	Dec. 5	Linking		Assignment-4 Due
15	27	Dec. 7	Exam review		

Appendix 1

Course Learning Outcomes and Link to Program Learning Outcomes (PLOs) Computer Science

Students who successfully complete this course will be able to:	CLO Level of Contribution, ie Low, Medium, High	Linked to CS Major PLOs
1. Write programs in C	**Medium**	**CS 1**
2. Analyze performance and optimize programs	**High**	**CS 3, CS 4**
3. Read and analyze the assembly of Intel x86 processor family	**High**	**CS 1, CS 2**
4. Demonstrate ability to reverse engineer Assembly code	**High**	**CS 2**
5. State how data types are represented and stored	**High**	**CS 1, CS 2**
6. Explain memory architecture and hierarchy	**High**	**CS 1, CS 2**
7. Describe virtual memory and the fundamentals of linking	**High**	**CS 1, CS 2, CS3, CS4**
8. Explain exceptional control flow	**Medium**	**CS 1**

Major Program Learning Outcomes (PLOs)

CS 1. Analyze a problem, and identify, define, and verify the appropriate computational tools required to solve it (Knowledge, Skill, Role in Context, Self-development).
CS 2. Apply up-to-date computational tools necessary in a variety of computing practices (Knowledge, Skill, Autonomy & Responsibility, Self-development).
CS 3. Implement algorithms as programs using modern computer languages (Knowledge, Skill).

CS 4. Apply their mathematical knowledge to solve computational problems (Knowledge, Skill, Autonomy & Responsibility, Self-development).
CS 5. Communicate computer science knowledge both orally and in writing (Skill, Autonomy & Responsibility, Role in Context).
CS 6. Collaborate in teams (Skill, Autonomy & Responsibility, Role in Context).

Course Learning Outcomes and Link to Program Learning Outcomes (PLOs) Computer Science

Students who successfully complete this course will be able to:	CLO Level of Contribution, ie Low, Medium, High	Linked to Engineering Major PLOs
1. Write programs in C	Medium	ENGR PLO 1 c
2. Analyze performance and optimize programs	High	ENGR PLO 6 b
3. Read and analyze the assembly of Intel x86 processor family	High	ENGR PLO 7 a, b
4. Demonstrate ability to reverse engineer Assembly code	High	ENGR PLO 7 a, b ENGR PLO 2 a
5. State how data types are represented and stored	High	ENGR PLO 7 a
6. Explain memory architecture and hierarchy	High	ENGR PLO 1 a
7. Describe virtual memory and the fundamentals of linking	High	ENGR PLO 1 a, ENGR PLO 7 a, b
8. Explain exceptional control flow	Medium	ENGR PLO 7 a, b

Engineering Major Program Learning Outcomes (PLOs):

1. an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics
    1. apply knowledge of mathematics in solving engineering problems;
    2. apply knowledge of natural sciences in solving engineering problems.
    3. apply knowledge and practices of engineering in solving problems.
2. an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors
    1. analyze and evaluate an engineering design against specific requirements.
    2. identify the non-technical constraints related to the engineering design.
    3. design an engineering system or component to meet desired needs within realistic constraints.
3. an ability to communicate effectively with a range of audiences
    1. demonstrate effective written communication of information, concepts and ideas.
    2. demonstrate effective oral or visual presentation of information, concepts and ideas.

4. an ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts
    1. demonstrate knowledge of engineering professional code of ethics
    2. evaluate the ethical implications of an engineering solution
    3. recognize the impact of engineering solutions in economic, environmental, and societal context
5. an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives
    1. fulfill individual duties.
    2. integrates input of team members.
6. an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions
    1. conduct or execute experiment in engineering.
    2. analyze data and draw conclusions from engineering experiments.
7. an ability to acquire and apply new knowledge as needed, using appropriate learning strategies.
    1. demonstrate the ability to learn independently.
    2. demonstrate the ability to find, evaluate and apply information from a variety of sources.