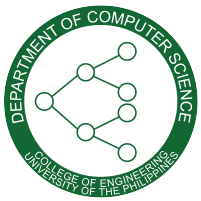


University of the Philippines, Diliman
College of Engineering
Department of Computer Science

Course Information

Course Number:	CS 21
Course Schedule:	**WFWX** W 1:00-4:00 lec, F 1:00-4:00 lab **WFUV** W 10:00 - 1:00 lec, F 10:00 - 1:00 lab
Course Title:	Computer Organization and Assembly Language Programming
Course Description:	Assembly language programming. Memory organization and architecture. Interfacing and communication. Digital Logic Design. Microcontrollers.
Credit:	4 units
Prerequisite:	CS 12 / ES 26
Methodology:	In-class lectures, programming exercises and projects
Instructor:	Kristofer E. delas Peñas
Consultation:	WF 1:00-6:00, Rm. 307, UPAECH
Email Address:	kedelaspenas@up.edu.ph

Course Goals

At the end of the course, the student should be able to:

- Understand the fundamentals and design of computers
- Identify the connection between hardware and software
- Create simple programs using assembly language
- Understand, design, and create simple digital circuits
- Program microcontrollers and design simple microcontroller-based circuits

Course Outline

1. Computer Organization

   Number systems and computer arithmetic
   IEEE 754: representing floating point numbers
   History of computers
   I/O devices
   Memory and Storage
   Assembly programming
   Registers and syscalls
   Data field, load-store
   Branches, conditionals, loops
   Bit masking and shifting, functions

2. Electronic Devices and Circuits

   Introduction
   Levels of abstraction: transistors to gates
   Boolean algebra
   Levels of abstraction: transistors to gates
   Basic electronics
   Building circuits from logic gates
   Logic implementation
   Using more complicated ICs

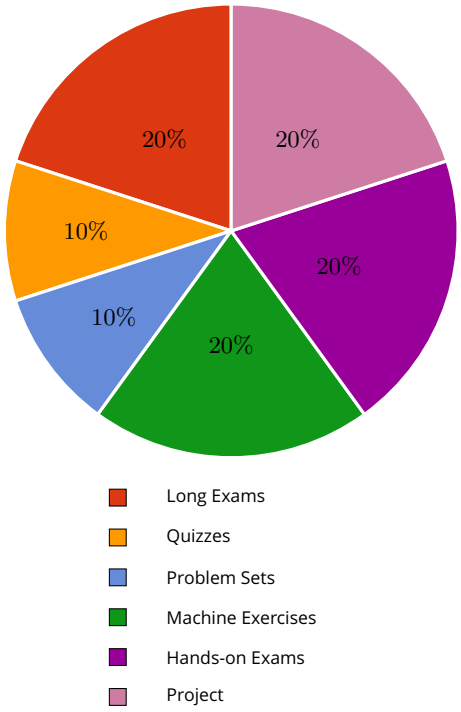
3. Processors

   Basic processor
   Measuring CPU Performance
   Pipelining
   Amdahl's law

   Microcontroller

4. Synthesis: Future of Computing

Course Requirements



Class Policies

- **Collaboration**
  Students are allowed to informally collaborate on their assignments, exercises and machine problems with other students who have taken the course previously or are currently taking the course. Submitting code copied verbatim or nearly verbatim even with proper citation is prohibited unless otherwise specified by the instructor.
- **Consultation**
  Consultation is encouraged. A student who wants to consult should inform the instructor at least a day before his/her preferred day of consultation.
- **Loss of Work**
  Students should make backup copies of all their work in this course. Loss of work due to hardware failure will not be considered as an acceptable excuse for late submission or non-submission of requirements.
- **Previous Work**
  Students are free to use programs they have written in the past provided they follow the required format and are authorized by the instructor.

- **Cheating**
  Any instance of copying the works and/or thoughts of others and passing it as one's own is considered as plagiarism. In using course materials, students should be careful not to claim words, ideas and algorithms as one's own.

Grading System

Students will be graded according to the following scale:

General Average	Final Grade
[92 - 100]	1.0
[88 - 92)	1.25
[84 - 88)	1.50
[80 - 84)	1.75
[76 - 80)	2.00
[72 - 76)	2.25
[68 - 72)	2.50
[64 - 68)	2.75
[60 - 64)	3.00
[0 - 60)	5.00

Tentative Course Schedule

Date	Day		Topic/Activity
5 August	Wednesday	Lecture	Course Introduction
7 August	Friday	Laboratory	Introduction to Assembly Language Programming
12 August	Wednesday	Lecture	Computer Organization
14 August	Friday	Laboratory	Registers
19 August	Wednesday		*NO CLASS (QUEZON CITY DAY)*
21 August	Friday		*NO CLASS (NINOY AQUINO DAY)*
26 August	Wednesday	Lecture	Computer Arithmetic
28 August	Friday	Laboratory	Load and Store Operations
2 September	Wednesday	Lecture	Representing Real Numbers
4 September	Friday	Laboratory	Bitwise Operators
9 September	Wednesday	Lecture	Memory
11 September	Friday	Laboratory	Conditionals and Loops
16 September	Wednesday	Lecture	Memory
18 September	Friday	Laboratory	Functions
23 September	Wednesday	Lecture	Interfaces and I/O Devices
25 September	Friday	Laboratory	Floating Point Arithmetic
30 September	Wednesday		*FIRST WRITTEN EXAM*
1 October	Thursday		*MID-SEMESTER*
2 October	Friday		*HANDS-ON CODING EXAM*
7 October	Wednesday	Lecture	Introduction to Electric Circuits: Laws of Electricity
9 October	Friday	Laboratory	Introduction to Electronics
14 October	Wednesday	Lecture	Network Analysis
16 October	Friday	Laboratory	Circuits and Networks
21 October	Wednesday	Lecture	Karnaugh Maps
23 October	Friday	Laboratory	Digital Logic Circuits
28 October	Wednesday	Lecture	Combinational and Sequential Logic
30 October	Friday	Laboratory	Complex ICs DEADLINE OF DROPPING
4 November	Wednesday	Lecture	Building a Processor
6 November	Friday	Laboratory	Sequential Logic
7 November	Saturday		*CIRCUITS EXAM*

11 November	Wednesday	Lecture	Pipelining
13 November	Friday	Laboratory	Introduction to Arduino
18 November	Wednesday		*NO CLASSES (APEC Meeting)*
20 November	Friday	Laboratory	Analog and Digital Input
25 November	Wednesday	Lecture	CPU Performance, Parallelism
27 November	Friday	Laboratory	Analog and Digital Output
1 December	Tuesday		*INTEGRATION PERIOD*
2-10 December			*FINALS WEEK* *(SECOND WRITTEN EXAM, ARDUINO PROJECT)*

References

Patterson, David, and John Hennessy. Computer Organization and Design: The Hardware/Software Interface. 3rd. Elsevier, 2005.
Hayes, John. Introduction to Digital Logic Design. Addison-Wesley. 1994.
Hayt, William et al. Engineering Circuit Analysis. 6th. McGrawhill, 2002.
Lecture slides and other resources at `http://kedelaspenas.github.io/CS21`.