

ECE20027

시스템프로그래밍

Yunmin Go

School of CSEE



Introduction

- Instructor: 이강 교수

- Office: NTH 408
- Tel.: 054-260-1387
- E-mail: yk@handong.edu
- Office hours
 - 화요일: 5,6교시

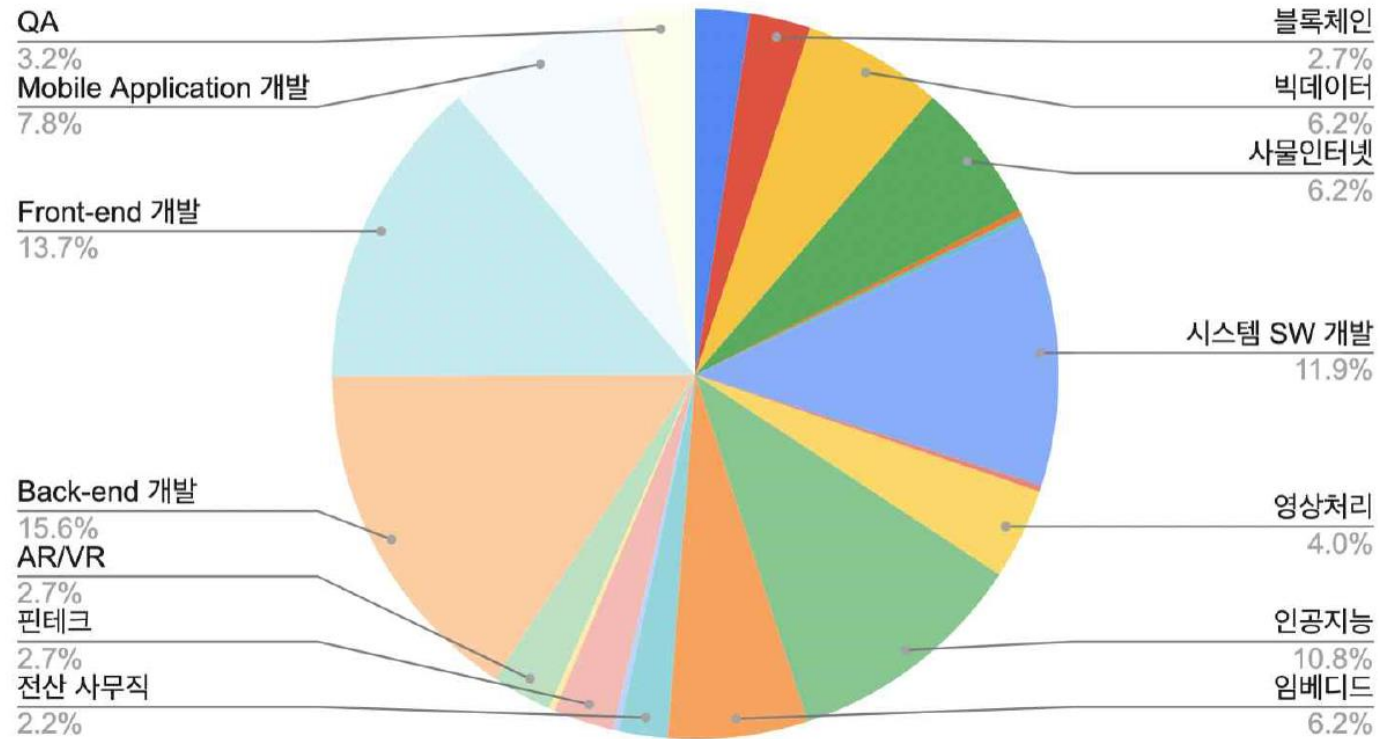
- Instructor: 고윤민 교수

- Office: NTH 306
- Tel.: 054-260-1384
- E-mail: yunmin@handong.ac.kr
- Office hours
 - 월요일/목요일: 6교시

시스템 프로그래밍 교과목 탄생의 배경

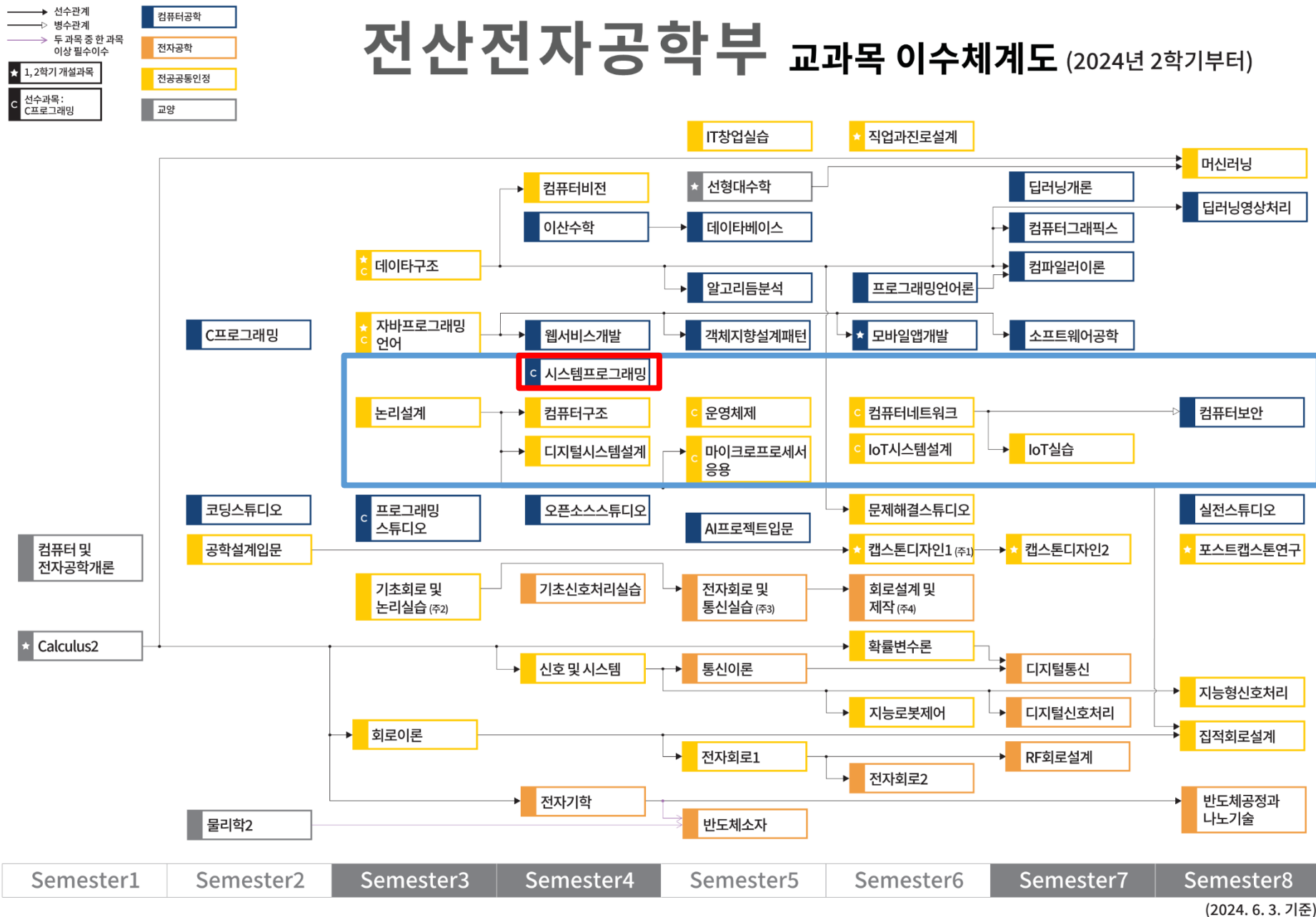
- 한동대 SW중심대학 사업 전략 계획 수립을 위한 설문 조사
 - 기간: 2022년 11월 1일 ~ 2022년 12월 10일
 - 대상: 한동대 가족기업, SW 중심대학 가족기업, 대경ICT기업 임직원

신입 채용 분야 (84개 기업체 임직원)





전산전자공학부 교과목 이수체계도 (2024년 2학기부터)



Course Objectives

- 시스템 프로그래밍을 통해 소프트웨어 개발을 위한 **컴퓨터의 핵심 작동 원리**를 이해한다.
- 시스템 소프트웨어 개발을 위한 **Assembly 및 C 등의 저수준 언어에 대한 이해도**를 높인다.
- **프로세스 관리, 메모리 관리, 파일시스템, 입출력 시스템** 등의 구성요소를 이해하고, 시스템콜을 사용하여 **시스템 소프트웨어를 구현**할 수 있다.
- 시스템 소프트웨어의 신뢰성과 성능을 높이기 위한 효과적인 **디버깅 및 시스템 최적화** 스킬을 사용할 수 있다.

Course Description

- 시스템 소프트웨어 및 사물인터넷 시스템을 개발하기 위해서는 **하드웨어 및 시스템 소프트웨어의 통합적인 구조를 이해**하고 양자간의 인터페이스를 설계 및 구현하는 능력이 필요하다. 본 과목에서는 시스템 프로그래밍을 통해 소프트웨어 개발을 위한 컴퓨터의 핵심 작동 원리를 보다 깊이 이해함으로써 고성능/대규모 소프트웨어를 개발할 수 있는 핵심 역량을 기른다. 또한 하드웨어/소프트웨어 통합 개발 및 시스템 소프트웨어 엔지니어로서 필요한 역량을 기르는 것을 목적으로 한다. 이 과목에서 개발되는 역량은 사물인터넷, 임베디드시스템, 고성능컴퓨팅, 분산컴퓨팅 등과 같은 신기술을 이해하고 활용하기 위한 기초 역량으로 활용될 수 있다. 구체적으로는, 다음 4가지를 배운다.
 - 1) C언어를 구사하는 역량을 심화하고, 기본적인 어셈블리어를 사용한 저수준 소프트웨어 개발 역량
 - 2) 소프트웨어 디버깅과 성능 최적화를 위한 소프트웨어 개발 역량
 - 3) 기본적인 운영체계의 기초 개념을 이해하고 주요 구성 요소인 프로세스 관리(스케줄링), 메모리 관리, 파일시스템, 입출력 시스템의 구성요소를 활용하는 프로그래밍 역량
 - 4) 컴퓨터구조와 하드웨어의 상호작용을 이해하고 하드웨어 자원을 직접 사용하는 소프트웨어의 작성

Course Operation Plan (1/6)

- Textbook

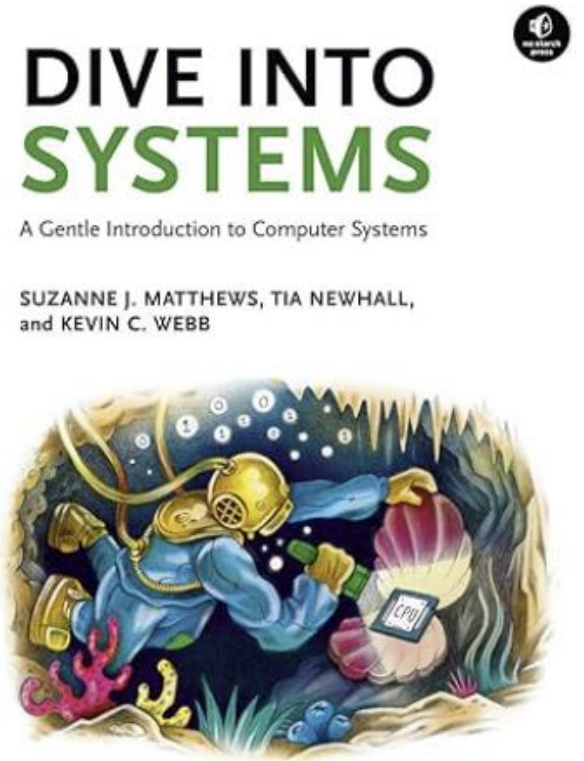
- Suzanne J. Matthews, Tia Newhall, Kevin C. Webb,
“**Dive into System**,” No Starch Press, 2022
 - <https://diveintosystems.org/>

- Prerequisites

- C Programming

- Recommend

- Data structures
- Computer Architecture and Organization (strongly recommend)



Course Operation Plan (2/6)

■ Grading Policies

Category	Policy
Attendance	10%
Homework	20%
Lab / Class Activities	10%
Quiz	10%
Midterm exam	25%
Final exam	25%

※ If you get 0 point on both the midterm and final exams, you will get F grade.

Course Operation Plan (3/6)

■ Attendance

- 3 tardiness == 1 absence
- If you are absent more than 1/4, you will get F grade.

■ Assignments

- Four programming assignments will be given
- All assignments must be handed in before the due date.
- If the assignment is late than due date, -1 pts penalty per day is given.
 - 1 day late: -1pts, 2 days late: -2pts, 3 days late: -3pts
 - Late submissions are allowed up to 3 days

Course Operation Plan (4/6)

■ Honor code

- You must strictly adhere to the Handong Honor Code.
- Any form of cheating, including code copying, will result in a score of zero.
- Utilizing AI tools like ChatGPT, Copilot, or similar to directly generate code or logical flow in any exams or assignments will be considered cheating.



Course Operation Plan (5/6)

- Basically, all students should participate offline lecture.
- All tests (quizzes, midterm exam, and final exam) will be offline only!
- Make-up lecture can be provided in online (ZOOM or LMS).
- Questions
 - E-mail
 - TA
 - Office hours

Course Operation Plan (6/6)

■ 참고 자료

- YouTube 소중대 동영상 자료
- ‘컴퓨터구조 및 설계’ 강좌 동영상
 - <https://www.youtube.com/playlist?list=PLoJdZ7VvEiROlYEltGhp-hg9jC47E3RDY>
- ‘처음 시작하는 Linux’ 강좌 동영상
 - <https://youtube.com/playlist?list=PLoJdZ7VvEiROLq5Fx8qKWcSL6AC1dAa3v&si=lx8BVkzljzjQg0Hly>

■ OSS 역량

- 도구 활용 능력
 - 효과적인 Linux 시스템 프로그래밍을 위한 Makefile과 Shell script를 작성할 수 있고, SW의 실행 및 최적화를 위해 gdb, valgrind 등과 같은 다양한 도구를 활용할 수 있다.
- API 검색 및 적용 역량
 - Linux 시스템 프로그래밍을 위한 적절한 API를 검색하고, 검색한 API를 활용할 수 있다.
- OSS 개발 역량
 - Linux OSS 프로젝트를 사용하여 코드를 이해하고 수정할 수 있다.

Weekly Schedule (Tentative) (1/2)

Week	Topic	Note
1	Low level C Programming: Command line arguments, Options	
2	Low level C Programming: Function pointer, Bit operators	
3	Low level C Programming: Compiler directive, Dynamic library, Make	
4	Low level C Programming: Debugger, Profiling	
5	File & I/O: File I/O	
6	File & I/O: File and directory management (access control)	
7	File & I/O: File system	
8	File & I/O: File system	

Weekly Schedule (Tentative) (2/2)

Week	Topic	Note
9	Midterm Exam	
10	Process & Memory: Process management	
11	Process & Memory: Threading	
12	Process & Memory: Process address space	
13	Process & Memory: Memory mapping and allocation	
14	Process & Memory: Optimization, Inline assembly	
15	Process & Memory: Signal, Interrupt, Timer	
16	Final Exam	