

# Computer Science 학습 가이드

# Contents

1. 개요
2. Computer Science 소개
3. Computer Science 과목체계
4. 자기주도학습 진행 순서
5. 맞춤 학습 가이드
6. 자기주도학습 진행 방법
7. 과목별 소개
8. CS스터디 그룹

## SSAFY 교육생 여러분들!

본 학습가이드는 컴퓨터 기초 학문 소양을 높이고 ,  
개발과정에서 발생한 문제를 해결하기 위해 Computer Science  
다양한 과목 중 어떤 부분을 찾아봐야 하는지,  
전체적인 로드맵을 통해 학습할 수 있도록 제시합니다.

학습에서 가장 중요 한 것은 스스로 계획하고 실천에 옮기는 것입니다.  
공개된 강의와 교재를 활용하여, 전략적으로 학습계획을 세워보고  
자신이 보완해야 하는 과목을 찾아보며 학습에 많은 도움이 되시길 바랍니다.

## 1. 학습 목적

- SSAFY 교육생의 컴퓨터 기초 학문 소양을 높이기 위함
- 이론적 지식 베이스를 갖추고 이후 개발 과정에서 발생한 문제를 해결하기 위해  
“어떤 과목의 어떤 부분을 찾아봐야 하는지 알 수 있도록” 전체적인 로드맵을 학습하기 위함

## 2. 학습 형태

- 공개된 강의와 교재를 활용하여 자기주도학습을 진행함
- 그룹 스터디를 통해 자기 공부의 최종 복습 점검 기회로 활용함

## 3. 학습 기간 및 방법

- 총 5개월  
\*하루 1~1.5시간 학습 기준
- 학습내용은 GitLab에 정리

## 2. Computer Science 소개

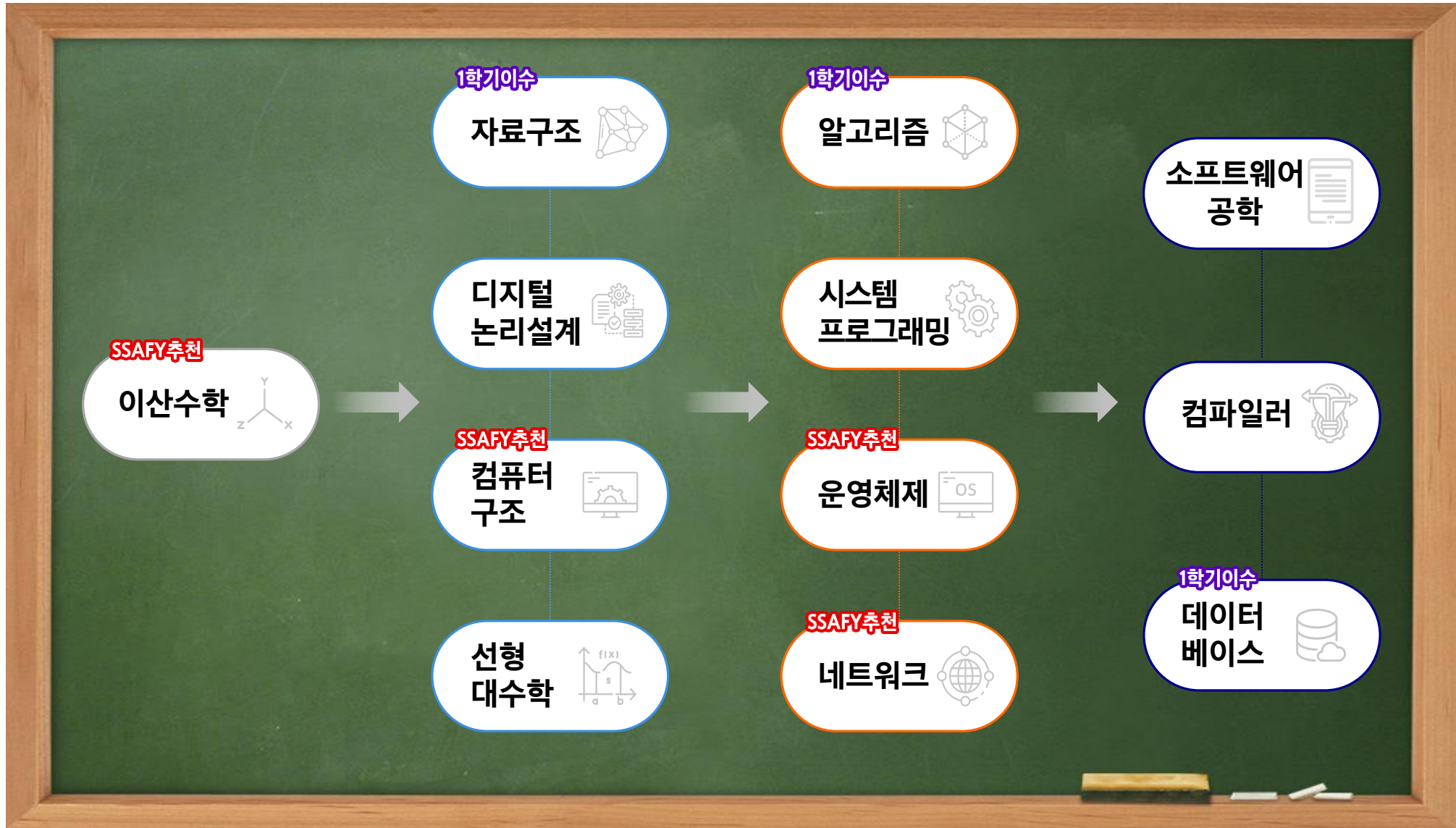
Computer Science는 현대의 정보화 사회에서 컴퓨터 시스템과 컴퓨터와 관련된 여러 기술을 개발하여 익히고 이를 각 분야에 응용함을 목적으로 하는 학문 분과이다. 기초 이론 분야와 프로그래밍 분야, 컴퓨터 시스템 분야, 응용 분야로 나뉩니다.

- ① **기초 분야**: **이산수학**, **자료구조**, 오토마타, **알고리즘** 등 컴퓨터 기술을 발달시키기 위한 기반이 되는 로직을 다루는 부분
- ② **프로그래밍 분야**: 컴퓨터 기술의 핵심을 이루는 분야로 C, C++, Java 등 여러 가지 **프로그래밍** 언어로 컴퓨터에게 명령하여 작동시키는 기술을 연구
- ③ **시스템 분야**: 컴퓨터의 하드웨어 구조와 컴퓨터 시스템을 구성하는 **운영체제**, **네트워크** 등을 취급
- ④ **응용 분야**: 위의 기술들을 기반으로 하여 필요에 따라 세부적이고 구체적으로 응용·발달, 분산 시스템 및 병렬 시스템, 인공지능, **데이터베이스**, 멀티미디어, 패턴인식 및 영상처리, 소프트웨어공학(-工學, software engineering) 등을 학습

SSAFY에서는 위의 분야를 비전공자(인문계/이공계)트랙과 전공자트랙으로 나누어 1학기에서 이수하는 과목과 자기주도적으로 학습하는 과목으로 구성하였으며, 자기주도학습 방법에 대한 가이드를 제공합니다.

# 3. Computer Science 과목 체계

학습 가이드

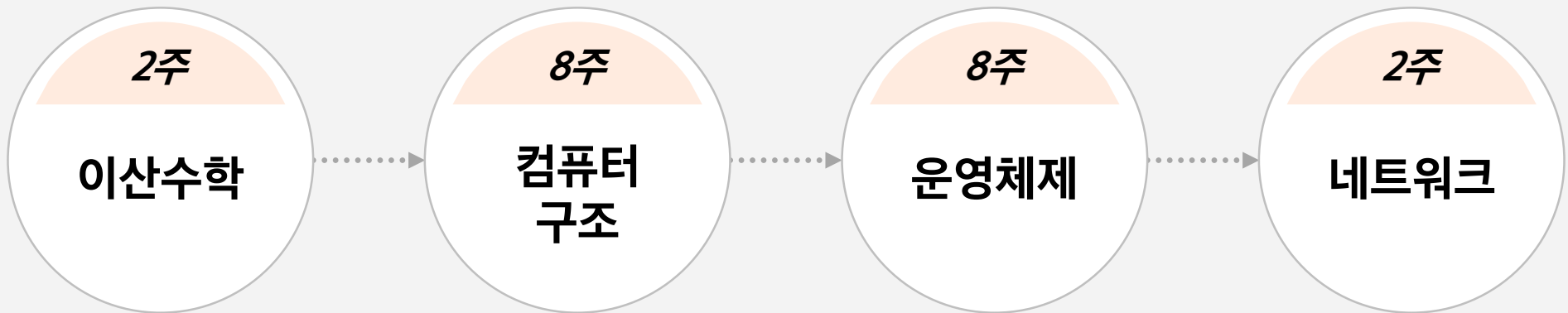


## 4. 자기주도 학습 진행 순서

학습 가이드

### ✓ 자기주도학습 과목 Flow

총 20주 학습



“

이산수학/ 컴퓨터 구조 과목을 통해 **이론 지식을 학습**하고  
운영체제/네트워크 과목을 통해  
**컴퓨터시스템의 동작과 통신에 대한 지식을 학습**합니다.

”

# 5. 맞춤 학습 가이드

모든 고급 개발 지식과 업무의 근간은 알게 모르게 우리가 배웠던 Computer Science 기본지식으로 연결됩니다. 한 단계 더 높은 개발자로 거듭나기 위해 **가장 핵심적인 지식, Computer Science 학습을 비전공자 / 전공자 모두에게 추천드립니다.**

## ✓ 자기주도학습 과목 Flow



### 비전공자 (인문/기타)

실력 있는 개발자로 커리어를  
이어나가기 위한 기초를 탄탄히  
다질 수 있음

### 비전공자 (이공계)

Computer Science 과목 간의  
관련성을 이해해보고 개발 실무와  
어떻게 연결 지어야 할지 고민해  
보며 학습 할 수 있음

### 전공자 (SW)

Computer Science적 뼈대를  
튼튼하게 다지고 실력을  
향상시키기 위해 추천 과목이  
아니더라도 복습해 볼 수 있음

※ 비전공자 권장 학습 기간

※ 전공자 복습 위주 학습 기간

이산수학 2주 → 컴퓨터 구조 8주 → 운영체제 8주 → 네트워크 2주 (총 20주)

이산수학 2주 → 컴퓨터 구조 4~8주 → 운영체제 4~8주 → 네트워크 2주 (총12~20주)



# 6. 자기주도 학습 진행 방법

필수

## 유튜브 강의 시청



유튜브를 통해 추천 강의를 언제든지 자유롭게 학습하고 시청한 내용을 학습플랜템플릿에 작성합니다.

: 제공한 GitLab 저장소 포크해서 마크다운으로 각자 정리함  
(<https://lab.ssafy.com/self-study/cs-study.git>)

선택

## 교재 및 블로그 학습



교재 및 블로그를 활용하여 부족한 과목을 추가 학습합니다.  
이외에 추가 필요한 교재는 교보문고 전자도서관을 이용하여 학습합니다.

: 기본 추천 교재는 개별로 구입함

추천

## 스터디 학습



CS교과목 강의 수강과  
그룹 스터디를 병행할 수 있습니다.

# 6. 자기주도 학습 진행 방법 \_ 학습플랜 템플릿 사용법

학습 가이드

## 1. 제공한 GitLab 저장소를 포크합니다.

self-study > CS Study > 상세 <https://lab.ssfy.com/self-study/cs-study>

**CS Study** 프로젝트 ID: 48670 | [프로젝트에서 나가기](#)

Fork 0 Clone

포크 진행중

Please wait while we import the repository for you. Refresh at will.

The project was successfully forked.

**CS Study** 프로젝트 ID: 49249

라이선스 추가 0 Commit 1 Branch 0 Tag 0바이트 Files

다른 프로젝트에서 포크됨: self-study / CS Study

master cs-study / +

이력 파일 찾기 앱 IDE

Fix link & study schedule  
15 분 전 에 최광호 님이 커밋하였습니다. 4bb3c8ce

README Add CHANGELOG Add CONTRIBUTING CI/CD 설정

이름	최근 커밋	최근 업데이트
네트워크	initial	2 주 전
운영체제	Fix link & study schedule	15 분 전
이산-수학	initial	2 주 전

# 6. 자기주도 학습 진행 방법 \_ 학습플랜 템플릿 사용법

학습 가이드

## 2. 첫 페이지의 전체 과목별 강의 내용을 확인하고, 학습 계획을 수립한 후 진행상태를 관리합니다.

### CS 학습 플랜/정리 템플릿

과목별 추천 동영상 강의를 중심으로 진도를 추적하고 배운 내용을 정리하기 위한 템플릿

#### 이산 수학

파트	진행상태	학습시간	강의내용	비고
			강의 개요	권장 학습기간: 2주, 학습분량: 주당 7.5시간
1	<input checked="" type="checkbox"/>	2	이산수학 기초	
2	<input type="checkbox"/>	5	명제, 추론, 귀납, 부울대수	
2	<input type="checkbox"/>	3	알고리즘, 그래프	
2	<input type="checkbox"/>	2	형식언어와 오토마타, 셀	
2	<input type="checkbox"/>	3	관계, 부분순서	

#### 진행상태 관리 예시

구분	코드	설명
<input type="checkbox"/>	:white_large_square:	학습할 대상(TODO)
<input checked="" type="checkbox"/>	:white_check_mark:	학습 완료(DONE)
<input type="checkbox"/>	:black_large_square:	학습 미대상(SKIP)

#### 컴퓨터-구조

파트	진행상태	학습시간	강의내용	비고
			강의 개요	권장 학습기간: 8주, 학습분량: 주당 5시간

# 6. 자기주도 학습 진행 방법 \_ 학습플랜 템플릿 사용법

학습 가이드

## 3. 학습 시작 전 과목별 강의개요 설명을 확인합니다.

### 이산 수학

파트	진행상태	학습시간	강의내용	비고
			강의 개요	권장 학습기간: 2주, 학습분량: 주당 7.5시간
1	☑	2	이산수학 기초	
2	☐	5	명제, 추론, 귀납, 부울대수	
2	☐	3	알고리즘, 그래프	
2	■	2	형식언어와 오토마타, 셈	
2	☐	3	관계, 부분순서	

#### readme.md

💡 이산 == Discrete로써 별개(불 연속적인) 데이터 흐름을 다루기에 적합한 컴퓨터의 특성을 살린 수학으로 컴퓨터 과학의 다른 과목들에 공기처럼 녹아 있는 개념들이 많은 가장 기본적인 과목이면서 수학적 사고와 컴퓨팅 사고력을 향상하는데 도움이 됩니다.

#### 학습 주제

- 기본적인 명제와 연산자, 역, 이, 대우 동치에 대한 이해
- 논리학, 집합론, 수론, 조합론, 관계, 그래프 이론과 형식언어와 오토마타의 이해

#### 추가 교재 및 참고자료

- ▼ 영상 : 이산수학 기초 (1H), 이산수학 (11.5H)
  - 이산수학의 개요와 수학적 논리 사고를 위한 기초적인 명제와 논리에 대해 쉽게 알려줍니다.  
<https://www.youtube.com/playlist?list=PLRx0vPvEmdDgOIBt9MKQl-uMVrtac4n>
  - 이산수학의 주요 이론들을 전반적으로 다루고 있으며 문제와 해설까지 해줘서 효과적으로 학습이 가능합니다. (순서를 재정리한 별도 링크 사용) <https://www.youtube.com/playlist?list=PLW8wOTYOlufFr4fajXEVMghqQNYOOql>
- ▼ 교재 : 컴퓨팅 사고력을 키우는 이산수학  
컴퓨터 연산을 이해하기 위해 필요한 수학적 이론을 쉽게 풀어낸 이산수학 입문서임. 기본적인 수학지식과 컴퓨터 관련 지식을 이해하고 있다면 학습에 많은 도움이 될것임
- ▶ 블로그 : <https://brunch.co.kr/@toughprogrammer/8>

# 6. 자기주도 학습 진행 방법 \_ 학습플랜 템플릿 사용법

학습 가이드

## 4-1. 웹 IDE 를 사용해서 학습한 내용을 마크다운 문서로 요약 정리하고 커밋이 가능합니다.

The screenshot shows the GitHub web IDE interface for the 'CS Study' repository. The repository name 'CS Study' and its ID '49249' are at the top. Below the repository name, there are links for '라이선스 추가', '0 Commit', '1 Branch', '0 Tag', and '0바이트 Files'. A dropdown menu shows 'master' and 'cs-study / +'. The 'web IDE' button is highlighted with an orange box. The file explorer on the left shows the file structure, with 'readme.md' selected. The main editor area displays the content of 'readme.md', which includes a video link and several sections of text. The 'commit' button is highlighted with an orange box at the bottom of the file explorer.

CS Study  
프로젝트 ID: 49249

라이선스 추가 0 Commit 1 Branch 0 Tag 0바이트 Files

다른 프로젝트에서 포크됨: self-study / CS Study

master cs-study / +

이력 파일 찾기 웹 IDE

readme.md

편집 Preview Markdown Open in file view

1 영상: <https://www.youtube.com/playlist?list=PLRx0vPv1EmdDg0IBt9MKQ1-uMvRxtac4n>

2

3 ### 1강 이산수학 개요

4 - 불연속적인 숫자를 다루는 수학

5 - 컴퓨터를 위한 수학, 참과 거짓으로 살펴보는 컴퓨터 수학, 전산, 정보 수학, 1학년이나 처음 컴퓨터를 배우는 입장에서 배우기 쉽, 자료구조 또는 알고리즘의 베이스, 논리적 사고, 컴퓨팅 사고력 향상

6 - 여기에 배운 내용을 요약 정리

7

8 ### [2강 명제와 연산자](이산-수학/이산수학-기초/명제와-연산자.md)

9

10 ### 3강 역, 이, 대우

11 -

12

13 ### [4강 동치 관계](이산-수학/이산수학-기초/동치-관계.md)

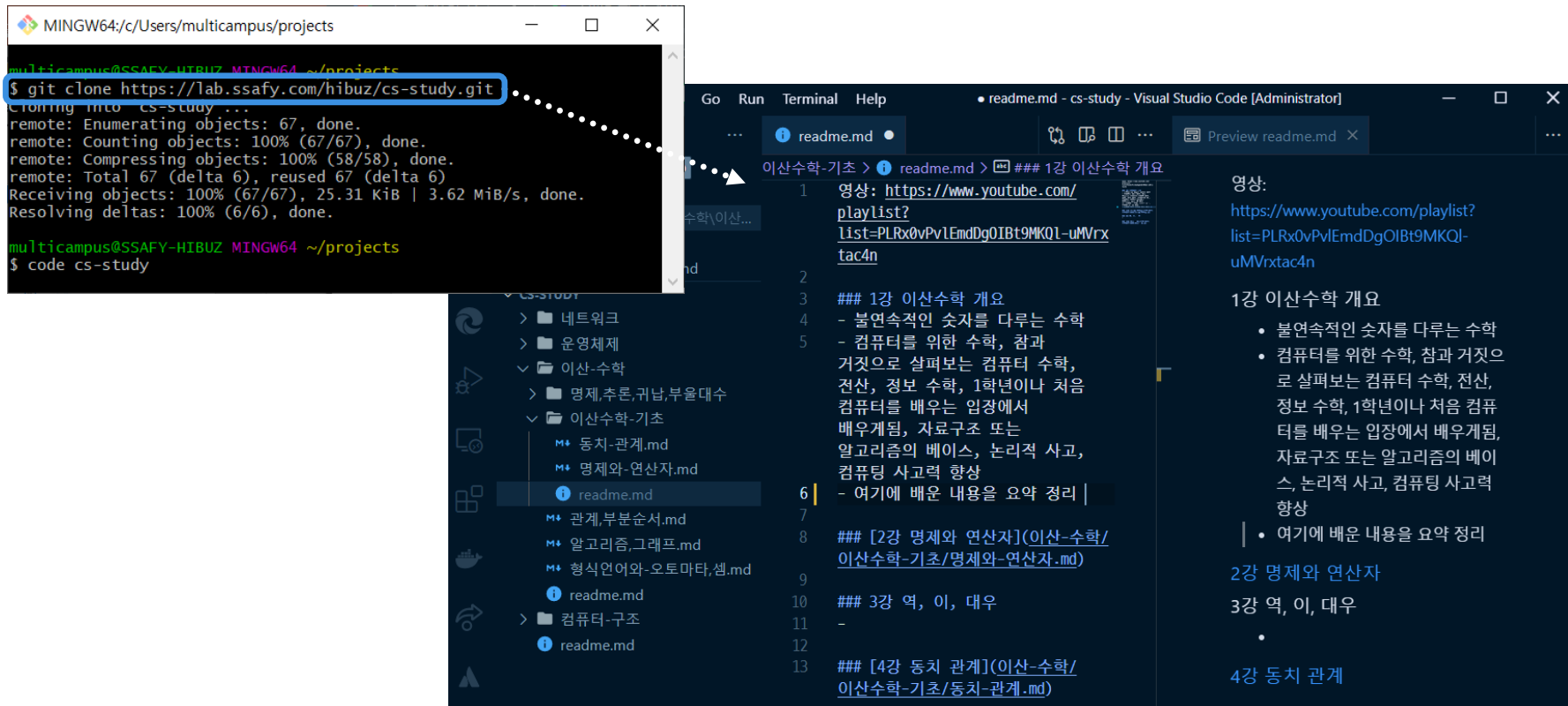
커밋...

1개가 스테이징되지 않았고, 0개가 스테이징된 변경사항

# 6. 자기주도 학습 진행 방법 \_ 학습플랜 템플릿 사용법

학습 가이드

4-2. 웹 IDE 대신 포크된 주소를 Clone 한 다음 VS Code 또는 개인별로 사용하고 있는 마크다운 에디터로 작성 후 커밋&푸시해도 됩니다.



# 7. 과목별 소개

과목	이산수학 (난이도 하)
학습목표	이산 == DISCRETE 로써 별개(불 연속적인) 데이터 흐름을 다루기에 적합한 컴퓨터의 특성을 살린 수학으로 컴퓨터 과학의 다른 과목들에 공기처럼 녹아 있는 개념들이 많은 가장 기본적인 과목이면서 수학적 사고와 컴퓨팅 사고력을 향상하는데 도움이 됨
학습주제	1. 기본적인 명제와 연산자, 역, 이, 대우 동치에 대한 이해 2. 논리학, 집합론, 수론, 조합론, 관계, 그래프 이론과 형식언어와 오토마타의 이해

학습내용 (유튜브강의)	주제		시간	방법
	Part I	• 명제와 집합의 개념과 부울 대수 이해	7시간	유튜브 학습시청 후 깃랩정리  ※권장학습기간: 2주 ※학습분량 : (유튜브강의기준) 1일 1.5시간 학습 추천★
	Part II	• 알고리즘과 관련된 관계의 종류 및 그래프, 계산이론	8시간	

# 7. 과목별 소개

		구분	설명	
학습 자료	필수	유튜브 강의		
		Part I • 이산수학 강좌 (Discrete Mathematics Tutorial For Beginners) • <a href="https://www.youtube.com/playlist?list=PLRx0vPvIEmdDgOIBt9MKQI-uMVrxtac4n">https://www.youtube.com/playlist?list=PLRx0vPvIEmdDgOIBt9MKQI-uMVrxtac4n</a>	이산수학의 개요와 수학적 논리 사고를 위한 기초적인 명제와 논리에 대해 쉽게 알려줍니다.	
		Part II • 이산수학 • <a href="https://www.youtube.com/playlist?list=PLW8wOTYOluvFr4favjXEVXMghqQNYOOql">https://www.youtube.com/playlist?list=PLW8wOTYOluvFr4favjXEVXMghqQNYOOql</a>	이산수학의 주요 이론들을 전반적으로 다루고 있으며 문제와 해설까지 해줘서 효과적으로 학습이 가능합니다.	
	선택	교재	• 컴퓨팅 사고력을 키우는 이산수학 • 박주미 저	컴퓨터 연산을 이해하기 위해 필요한 수학적 이론을 쉽게 풀어낸 이산수학 입문서입니다. 기본적인 수학지식과 컴퓨터 관련지식을 이해하고 있다면 학습에 많은 도움이 될 것입니다.
	참고	블로그	• <a href="https://brunch.co.kr/@toughprogrammer/8">https://brunch.co.kr/@toughprogrammer/8</a>	



# 7. 과목별 소개

과목	컴퓨터 구조 (난이도 중)
학습목표	논리 설계의 다음 과목으로 논리 게이트를 활용하여 실제 컴퓨터 구조가 어떻게 구성 되는지와 컴퓨터의 하드웨어적인 구성과 내부적인 동작에 대해 배우고 소프트웨어 엔지니어지만 하드웨어적인 부분까지 이해하면 프로그래밍에도 도움이 되고 이후 컴퓨터 구조와 가까이에 있는 운영체제 과목에 대한 이해도를 높일 수 있음
학습주제	<ul style="list-style-type: none"> <li>• 현대적인 컴퓨터 구조에 이르기까지의 역사를 살펴보고 컴퓨터 시스템 구성요소 이해</li> <li>• CPU, Memory, I/O 등의 하드웨어적인 구성과 이를 사용해 데이터를 처리, 저장, 이동 및 제어하는 컴퓨터의 내부 동작 이해</li> </ul>

	구분	주제	시간	방법
학습내용	Part I	• 디지털 부품과 연산을 위한 논리 회로 이론과 기본적인 컴퓨터의 구조와 설계	19시간	유튜브 학습시청 후 깃랩정리  ※권장학습기간: 8주 ※학습분량 : (유튜브강의기준) 1일 1시간 학습추천★
	Part II	• 프로그램이 CPU, Memory, I/O에 의해 처리되는 과정과 각각의 구조이해	21시간	

# 7. 과목별 소개

구분		설명
학습 자료	필수	Part I • '의 구조는 어떻게 되어있을까? CPU와 메모리! (안필과학 - 랩미팅 15화) • <a href="https://youtu.be/SiC74U8aJbM">https://youtu.be/SiC74U8aJbM</a>
	유튜브 강의	Part II • 컴퓨터시스템구조 (Dr. Chang의파운데이션) <a href="https://www.youtube.com/playlist?list=PLc8fQ-m7b1hCHTT7VH2oo0Ng7Et096dYc">https://www.youtube.com/playlist?list=PLc8fQ-m7b1hCHTT7VH2oo0Ng7Et096dYc</a>
	선택	• 컴퓨터구조 이론 • <a href="https://www.youtube.com/playlist?list=PLNLxw7PyaZW_LqGzSLw-q4YFgd0GUvGfw">https://www.youtube.com/playlist?list=PLNLxw7PyaZW_LqGzSLw-q4YFgd0GUvGfw</a>
	유튜브 강의	• 컴퓨터구조 이론이라는 이름으로 만들어진 플레이 리스트이지만 하드웨어, 운영체제와 프로그래밍 언어의 내용도 포함된 방대한 내용으로 궁금한 내용만 선택적으로 활용하기 좋습니다.
	선택	MIT의 컴퓨터과학 교재로 사용되고 있으며, 기본적인 하드웨어 플랫폼과 현대적인 소프트웨어 계층을 밑바닥에서부터 차근차근 구성해 보는 12개의 장과 프로젝트로 이루어져 습니다. 이 과정을 통해 학생들은 하드웨어 아키텍처, 운영체제, 프로그래밍 언어, 컴파일러, 데이터 구조, 알고리즘 및 소프트웨어 공학에 대해 실용적인 지식을 얻을 수 있습니다.
	교재	• 밑바닥부터 만드는 컴퓨팅 시스템 • 노암 니산 (예루살렘 히브루대학교 교수) 저 역자: 김진홍
	참고	
	블로그	• <a href="https://brunch.co.kr/@toughrogrammer/14">https://brunch.co.kr/@toughrogrammer/14</a>

# 7. 과목별 소개

과목	운영체제 (난이도 중)
학습목표	시스템 하드웨어를 관리할 뿐 아니라 응용 소프트웨어를 실행하기 위하여 공통 시스템 서비스를 제공하는 시스템 소프트웨어로써 대규모 서비스를 운영하거나 서버의 성능 향상 또는 튜닝을 위해서 운영체제와 관련된 지식이 요구됨
학습주제	<ol style="list-style-type: none"> <li>1. 운영체제 및 컴퓨터 시스템의 구조를 살펴보고 운영체제의 프로그램 실행 과정에 대한 이해</li> <li>2. CPU 스케줄링, 프로세스 관리(동기화, 데드락), 메모리/가상메모리 관리, 파일 시스템 구조 및 디스크 관리 등의 운영체제 역할 이해</li> </ol>

학습내용	구분		시간	방법
	Part I	<ul style="list-style-type: none"> <li>• 운영체제 개요</li> <li>• 컴퓨터의 핵심인 프로세스와 관련된 운영체제 관리기법</li> </ul>	20시간	유튜브 학습시청 후 깃랩정리  ※권장학습기간: 8주 ※학습분량 : (유튜브강의기준) 1일 1시간 학습추천★
	Part II	<ul style="list-style-type: none"> <li>• 메모리, 파일시스템, 디스크 관리 등의 운영체제의 역할</li> </ul>	16시간	

# 7. 과목별 소개

구분		설명
학습 자료	<div>필수</div> <div>MOOC 강의</div>	<div>Part I , II</div> <div><div>• 운영체제</div><div>• <a href="http://www.kocw.net/home/search/kemView.do?kempld=1046323">http://www.kocw.net/home/search/kemView.do?kempld=1046323</a></div></div> <div>주요 내용이 잘 구성되어 있고 PPT가 영상에서 동시에 보이며, 전달 방법도 좋습니다.</div>
	<div>선택</div> <div>교재</div>	<div><div>• 그림으로 배우는 구조와 원리 운영체제</div><div>• 구현회 저</div></div> <div>운영체제 입문에 꼭 필요한 기본 개념과 원리를 그림을 중심으로 자세히 설명하고 있습니다. 최신 경향을 반영하였으며 전체 본문을 이해하기 쉽게 풀었습니다.</div>
	<div>참고</div> <div>블로그</div>	<div><div>• <a href="https://brunch.co.kr/@toughprogrammer/15">https://brunch.co.kr/@toughprogrammer/15</a></div></div>

# 7. 과목별 소개

과목	네트워크 (난이도 중)
학습목표	컴퓨터와 컴퓨터간 통신망으로 연결된 컴퓨터 네트워크는 OSI 모형에 의해 7계층으로 구분되고 데이터를 전송하기 위한 복잡한 과정에 대한 이해를 기반으로 응용 소프트웨어에서 부터 OS를 거쳐 하드웨어까지 걸쳐있는 다양한 네트워크 문제 상황에 대한 문제해결 능력을 키우고자 함
학습주제	<ol style="list-style-type: none"> <li>1. 네트워크 기초와 네트워크 계층과 관련된 라우팅, IP, ARP, ICMP 프로토콜의 개념 및 동작 방식과 인터넷 주소 체계에 대한 이해</li> <li>2. 데이터 패킷 전송계층과 관련된 TCP(연결 지향)/UDP(비연결) 프로토콜의 개념과 신뢰성 있는 통신을 위한 연결, 흐름, 혼잡 제어의 동작방식 이해</li> <li>3. 기타 계층과 웹과 관련된 응용(7) 계층의 대표적인 HTTP 프로토콜에 대한 이해</li> </ol>

학습내용	구분		시간	방법
	Part I	• 네트워크 기초와 인터넷 관련 주요 프로토콜과 주소체계	7시간	유튜브 학습시청 후 깃랩정리  ※권장학습기간: 2주 ※학습분량 : (유튜브강의기준) 1일 1.5시간 학습 추천★
	Part II	• 전송계층 주요 프로토콜과 7계층의 대표적인 HTTP 프로토콜	8시간	

# 7. 과목별 소개

구분		설명
학습 자료	필수 유튜브 강의	Part Ⅰ,Ⅱ • 네트워크 • <a href="https://www.youtube.com/watch?v=Av9UFzl_wis&amp;list=PL0d8NnikouEWcF1jJueLdjRIC4HsUIULi&amp;index=1">https://www.youtube.com/watch?v=Av9UFzl_wis&amp;list=PL0d8NnikouEWcF1jJueLdjRIC4HsUIULi&amp;index=1</a> 네트워크에 대한 계층별 프로토콜에 대한 쉬운 설명과 PPT 애니메이션으로 동작 흐름을 쉽게 이해할 수 있게 설명합니다.
	추천 SSAFY 강의 (추후 제공)	1. OSI 7 계층, Packet 구조 2. 2 계층, ARP, LAN/WAN, L2 Switch 3. 3 계층, ICMP, L3 Router, Network 분리 4. 4 계층, TCP #1, 개념 5. 4 계층, TCP #2, Sliding window, 흐름/에러/혼잡 제어, CRC 6. 4 계층, TCP #3, 3-way handshake, Kernel 구조 7. 방화벽 8. 공유기
	선택 교재	• IT 엔지니어를 위한 네트워크 입문 • 고재성, 이상훈 저 네트워크가 생소한 개발자와 서버 엔지니어에게 네트워크 기초를 쌓는데 도움이 되고 클라우드 데브옵스 시대에 필요한 가상화 기술까지 그림으로 이해하기 쉽게 풀어 설명합니다.
	참고 블로그	• <a href="https://brunch.co.kr/@toughrogrammer/16">https://brunch.co.kr/@toughrogrammer/16</a>

## 8. CS스터디 그룹

Computer Science 과목에 대한 협동학습으로 학습효과를 높이고, 학습 네트워크를 통하여 학업성취도 증진에 도움을 줄 수 있는 **CS스터디 그룹을 만들어 보세요!!**

**스터디 대상 :** Computer Science 에 대한 기초지식을 쌓고 싶은 교육생 (전공/비전공자 무관)

**스터디 영역 :** Computer Science 과목

**스터디 구성 :** 4인 이상

**스터디 진행 :** 학사게시판을 통해 스터디 양식 작성 후 제출

**스터디 방법**

1. 전체 학습일정/분량세분화 및 (예시\_2주단위)온라인 공유회 일정계획 수립
2. 기간 별 목표 학습분량을 전체 스터디원이 각자 학습 진행
3. 온라인 공유회에서는 사전에 선정된 발제자가 발제를 진행하고 자유롭게 토론 및 Q&A 진행

**기대효과★**

학습에 대한  
동기부여 제공

다양한 정보  
공유

지식을 좀 더  
명확하게 정리

교육생 간  
친목도모

단기간 빠른  
학습 효과

꾸준한 학습  
가능

The background is a solid blue color with various abstract geometric elements. There are several circles of different sizes and shades of blue, some with white outlines. There are also some white 'x' marks and small white dots. In the bottom left and bottom right corners, there are rectangular areas filled with a grid of small white dots. A large, thick, curved blue shape is visible in the top left and bottom right corners.

# Thank You