

강의계획서

과목명	국문	알고리즘 및 컴퓨터적 사고	개발대학/교수	한양대학교 류호경
	영문	Algorithms and Computational Thinking	운영교수	한양대학교 류호경
* 개발교수와 운영교수는 다를 수 있습니다.				
			이수학점	3학점

국문개요	본 강의는 단순한 코딩 기술(Skill) 습득을 넘어, 복잡한 문제를 해결하는 '알고리즘 및 컴퓨터적 사고(Computational Thinking)'능력을 배양하는 것을 목적으로 한다. Phase 1: Brain Logic에서는 컴퓨터 없이 머리와 종이만으로 문제를 분해하고 추상화하여 해결 논리를 설계하는 'Unplugged Thinking'을 훈련하고 Phase 2: Python Logic에서는 설계된 논리를 파이썬(Python)을 통해 자동화하여 해결 속도를 극대화하는 과정을 학습한다. 이를 통해 학생들은 데이터중심의 인공지능시대에서 "컴퓨터처럼 생각하고", "사람처럼 해결하는" 것이 필요한지를 이해한다.																						
영문개요	This course aims to develop Algorithm and Computational Thinkingskills, focusing on problem-solving capabilities rather than mere coding techniques. The curriculum is divided into two phases. Phase 1: Brain Logic (Unplugged Thinking) focuses on designing algorithms to solve complex problems using only the brain and paper, utilizing concepts like abstraction, decomposition, and pattern recognition. Phase 2: Python Logic (Automate the Solution) involves implementing these designed logics into Python to maximize efficiency and automation. Students will learn to "think like a computer and solve like a human," bridging the gap between logical design and computational implementation.																						
학습목표	- Unplugged 사고력 확보: 컴퓨터 없이 복잡한 문제를 논리적으로 분해(Decomposition), 추상화(Abstraction), 패턴 인식(Pattern Recognition)하여 알고리즘을 설계하는 능력을 확보한다. - 알고리즘의 이해: 기본적인 컴퓨터 알고리즘의 설계 원리를 이해하고 일상생활의 문제에 적용할 수 있다. - 설계한 논리를 파이썬(Python)의 순차, 선택, 반복, 모듈화 기능을 활용해 코드로 구현하고 이해할 수 있다. - 복합 문제 해결: 단순 연산이 아닌 제약 조건이 많은 '8시간짜리 문제'를 정의하고, 이를 해결하는 독창적인 솔루션을 도출한다.																						
수업운영방식	- 액션 러닝 (Action Learning): 이론 학습 후 즉시 실습을 진행 (예: 배달 지도 추상화, 배달 기사 배정 프로세스 분해, 비 오는 날 배달 지연 패턴 분석 등) - 플립드 러닝 지원 (Flipped Learning): 각 주차별 핵심 개념에 대한 동영상 자료를 활용하여 학습 지원 - 협업 및 토론 Think-Pair-Share 방식을 통해 주제를 선정하고, Peer Review를 통해 타 그룹의 결과물을 상호 평가 - 실습(Lab): 실무 적용 및 문제 해결 (Python 환경 – Google Colab활용)																						
평가항목	<table><tr><td>출석</td><td>10%</td><td>퀴즈</td><td>10%</td></tr><tr><td>중간고사</td><td>-</td><td>토론</td><td>-</td></tr><tr><td>기말고사</td><td>20%</td><td>팀 프로젝트</td><td>30%</td></tr><tr><td>과제 (개인과제)</td><td>30%</td><td>학습참여도</td><td>-</td></tr><tr><td>총합</td><td colspan="3">100%</td></tr></table> ※ 4주차 이상 결석 시 타 항목의 점수와 무관하게 자동으로 미이수(F) 처리됩니다. ※ 출석 점수: 지각 시 1점, 결석 시 2점 ※ 평가 방법 및 배점은 담당 교수 재량에 의해 변경될 수 있습니다.			출석	10%	퀴즈	10%	중간고사	-	토론	-	기말고사	20%	팀 프로젝트	30%	과제 (개인과제)	30%	학습참여도	-	총합	100%		
출석	10%	퀴즈	10%																				
중간고사	-	토론	-																				
기말고사	20%	팀 프로젝트	30%																				
과제 (개인과제)	30%	학습참여도	-																				
총합	100%																						

주차별 강의 내용

주차	수업 주제	학습활동
1주차	융합의 시대	[Project 1] 8hr문제 정의
1 단 계	2주차 [추상화] 쓸데없는 것 버리기: 본질의 발견	배달 지도에서 '가로수'나 '건물 이름'은 버리고 '이동 시간'만 남기는 추상화 훈련 [숙제 1] 동일한 지역에 대한 두가지 추상화 [퀴즈 1] 물류관리 [Project 1] 8hr문제 정의
	3주차 [분해] 복잡한 것 간단하게 하기 [분해의 논리적 연결] IPO: Input→Process→Output	거대한 문제를 관리 가능한 작은 단위로 쪼개기 [숙제 2] 8hr문제 분해
	4주차 [패턴] 본질의 반복을 통한 답의 발견	문제 속 규칙성을 찾아 해결 모델 구축 → Pseudocode 구현 [숙제 3] 8Hr문제- 분해, 추상화, 패턴 [퀴즈 2] 패턴찾기
	5주차 [8시간짜리 문제] 나는 전세계 몇 등?	이진검색, 순차검색, 쌍비교검색 등의 알고리즘을 통해서 나의 팔로워수는 전세계 몇 등인지 알아맞추어 보자. → Pseudocode 구현 [숙제 4] 8Hr문제- 패턴 (알고리즘) [퀴즈 3] 물류관리
	6주차 [Project 1] Crack the Paper Code	8시간짜리 문제' 정의 및 알고리즘 설계 [숙제 5] 8Hr문제- elevator speech
	7주차	[숙제 6] 8Hr문제- TPS [Project 1] 피드백 반영본 제출
8주차	중간고사 미시행(콘텐츠 없음)	
2 단 계	9주차 [Sequence] 순차적 실행의 미학	Python 기초: 데이터 입력받기와 순차적인 프로세스 처리 [숙제 7] 8Hr문제- Sequence
	10주차 [Modularisation, Refactoring] 모듈화, 재사용의 미학	Python 기초: 함수선언 및 매개변수, 함수호출 활용 [숙제 8] 8Hr문제- Function
	11주차 [Selection] 조건에 따른 의사결정	Python 기초: If/Else 문을 통한 프로세스 분기 구현 [숙제 9] 8Hr문제- Selection
	12주차 [Iteration] 반복의 힘	Python 기초: Loop(반복문)을 활용한 수많은 데이터 처리 [숙제 10] 8Hr문제- Iteration
	13주차 파이썬이 왜 인공지능에서 가장 쉬운 도구인가?	Python 기초: List를 활용한 효율적 데이터 관리 [숙제 11] 8Hr문제- Data Structure
	14주차 [Project 2] 8시간짜리 문제 해결하기	8시간짜리 문제 (Project1) PYTHON으로 해결하고 동영상 작성 (3분 이내 발표)
	15주차	
16주차	기말고사 + Project 2 발표(Miro 플랫폼을 이용한 그룹프로젝트)	

※강의 내용 및 학습 활동이 일부 수정될 수 있으므로, 개강 이후 공지사항 및 학습 콘텐츠를 확인 바랍니다.