

두산 Rokey Boot Camp

스터디 주간 활동 보고서

팀명	Robo:Loop	제출자 성명	홍송은
참여 명단	전효재, 장수근, 홍송은, 김사웅		
모임 일시	2025 년 04 월 15 일 16 시 40 분 ~ 18 시 00(총 1 시간 30 분)		
장소	온라인 구글 미팅	출석 인원	4
학습목표	<ul style="list-style-type: none">코딩테스트에 자주 출제되는 핵심 알고리즘을 학습한다.해시, 이분 탐색, 그리디, BFS 등 알고리즘 주제를 선택하여 문제를 해결한다.문제 풀이를 통해 알고리즘 사고력과 구현 능력을 향상시킨다.		
학습내용	<ol style="list-style-type: none">프로그래머스 코딩테스트 고득점 Kit 를 활용하여 하나의 알고리즘 주제를 각자 선택하여 문제 풀이 진행 (https://school.programmers.co.kr/learn/challenges?tab=algorithm_practice_kit) <ul style="list-style-type: none">홍송은<ul style="list-style-type: none">해시 알고리즘 - 베스트 앨범 풀이해시(Hash)<ul style="list-style-type: none">정의: 데이터를 키(key)와 값(value) 쌍으로 저장하는 자료구조		

- 특징: 키 → 해시함수 → 인덱스를 통해 값을 빠르게 검색 가능
- 시간 복잡도: 평균적으로 검색, 삽입, 삭제 모두 $O(1)$
- 키를 해시 함수로 처리해 곧바로 저장 위치를 계산 → 리스트처럼 순차적으로 탐색할 필요 없이 즉시 접근 가능
- 예시:

`my_dict["name"]` →

1. "name" 키를 해시 함수에 전달
2. 계산된 해시값을 통해 메모리상의 위치(버킷) 결정
3. 해당 위치에서 값을 바로 반환

- 장수근


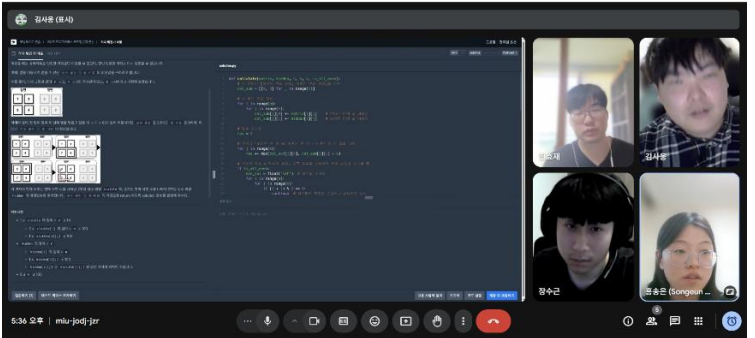
- 그래프 - 가장 먼 노드
- defaultdict
- collections 모듈중 하나로, 키가 존재하지 않을 때 자동으로 기본값을 생성
- 간결성과 가독성을 높일 수 있음.

- 김사웅

- 그리디 알고리즘 문제 - 격자 뒤집기 미로
- calculate()는 열마다 보이는 면 vs 숨겨진 면 - k 중 큰 값을 선택해서 점수를 계산함.
- 격자 크기가 짝수×짝수이면, 체스판 기준 검정칸 중 최소값 하나를 빼서 점수 과다를 보정함.
- solution()은 가능한 모든 행의 뒤집기 조합(2^n 개)을 비트마스킹으로 순회함.

	<ul style="list-style-type: none"> ○ 각 조합마다 행을 뒤집고 열은 그리디하게 처리하여 최대 점수를 계산함. ○ 각 조합의 점수에서 행을 뒤집은 비용을 빼고, 최대 점수를 <code>res</code> 에 저장함. ○ 최종적으로 가능한 모든 조합 중 가장 높은 점수를 리턴함. <ul style="list-style-type: none"> • 전효재 <ul style="list-style-type: none"> ○ 이분탐색 알고리즘 문제 - 입국심사 ○ 이분탐색을 사용하는 이유는, 완전탐색은 시간 복잡도가 $O(nm)$이고 이분탐색은 $O(n \log m)$으로, 평균적인 탐색 시간을 줄일 수 있기 때문. ○ 최소 시간과 최대 시간을 정해두고, <code>mid</code>(중간값)을 계속 변경해가며 답이 될 수 있는 시간을 탐색. ○ 중간 시간에서 각 입국심사 시간을 나누면, 그 시간 동안 각 심사관이 처리할 수 있는 사람 수를 구할 수 있음. ○ 모든 심사관이 처리한 사람 수의 합이 주어진 인원 이상이 되는 시간 중에서, 가장 작은 시간을 최종 정답으로 출력함. <p>2. AI 교육 커리큘럼에 맞춰 스터디 운영 방안 논의</p> <ul style="list-style-type: none"> • Google Colab 환경에서 기본 분류 모델을 구현 • 모델을 학습시키고 테스트 데이터에 대한 정확도 확인 • 테스트 결과를 기반으로 모델 성능 해석 	
활동평가	전효재	<p>이분탐색 알고리즘과 관련된 문제인 입국심사 문제 풀이를 담당함. 이분탐색을 사용해야 하는 이유를 설명하고, 최대한 익숙한 예제를 활용해 이해하기 쉽게 전달하고자 함. 기본적인 원리는 단순하지만, 이분탐색을 실제 문제에 어떻게 적용하고 <code>mid</code> 값을 조정해 나가는지에 대한 감을 잡을 수 있는 기회가 되었음. 설명을</p>

		준비하면서, 이분탐색은 단순히 아는 것보다 '왜 사용하는지'를 이해하는 게 더 중요하다는 걸 느낌.
	홍송은	해시 알고리즘에 대한 설명과 관련 문제 풀이를 담당함. 스터디 중 해시의 효율성과 작동 방식에 대한 질문이 제기되어, 해시가 어떻게 동작하며 왜 빠른지를 중심으로 설명을 보완함. 코드 내 주석을 더욱 상세히 정리하고, 리드미에도 핵심 개념과 장점을 요약하여 추가함. 또한 작성한 코드를 설명하는 과정에서 <code>sort()</code> 함수의 <code>key</code> 옵션 활용법과 <code>sorted()</code> 함수와의 차이점을 비교하여 보다 적절히 사용할 수 있도록 안내함.
	김사웅	프로그래머스 5 단계문제가 얼마나 어려운지 풀어보고 싶어 도전했으나 풀지 못해서 더 공부하기로 결정함. 그리디 탐색 알고리즘을 사용하여 풀어야 하며 BFS 와 DFS 로 배웠던 방식과는 조금 다른것 같음.
	장수근	BFS 코딩이 익숙치 않아 복습. 행렬 형식이 아닌 그래프의 형식은 처음이라 검색을 통해 <code>defaultdict</code> 를 학습하였고, 간결화와 가독성을 높일 수 있게 사용. 코드 초반부와 후반부의 인덱스 계산에 어려움을 겪어, 좀 더 익숙해지도록 학습의 필요성 느낌.
과제	<ul style="list-style-type: none"> • MNIST 숫자 이미지 분류 실습 <ul style="list-style-type: none"> ◦ Colab 에서 기본 분류 모델 구현 ◦ 모델 학습 후 테스트 정확도 확인 및 결과 해석 ◦ 각자 모델 성능 개선을 위한 방법 시도 ◦ 실습한 코드와 결과를 정리하여 제출 및 공유 	
향후 계획	<ul style="list-style-type: none"> • MNIST 데이터셋의 구조와 의미 이해 (훈련/테스트 데이터, 이미지 크기 등) • 각자 구현한 MNIST 분류 모델을 발표하고 결과를 비교 	

	<ul style="list-style-type: none"> • 모델 구성 방식, 성능 개선 방법 등을 중심으로 설명 및 피드백 진행 • AI 수업 수강 후 난이도와 이해도를 바탕으로 스터디 진행 방식 및 주제 조정 	
첨부 자료	스터디 화면 1	
	스터디 화면 2	
	결과물	<p>Github 주소:</p> <p>https://github.com/yellowHSE/ds_rokey4_study_team2/tree/main/codes/week5_AlgorithmPractice</p> 