

생성형 인공지능을 기반으로 한 알고리즘 학습 플랫폼 구현

Development of an Algorithm Learning Platform Powered by Generative AI



중간보고서

분과	B	번호	17
팀명	CodeSphere	팀장	202255513 김대욱
지도교수	조준수	팀원	202155525 김문경 202255535 김진우

1	목차	
2	연구 배경	4
2.1	초·중등교육과정 코딩교육 의무화	4
2.2	취업시장 코딩테스트 확대	4
3	문제 상황	5
3.1	코딩 교육의 어려움	5
3.2	기존 프로그래밍 학습사이트의 단점	5
4	과제의 목표	6
5	요구사항 및 제약 사항 분석에 대한 수정사항	6
5.1	요구사항 및 제약 사항 분석 및 전문가 자문의견	6
5.2	수정된 기능 목표 정의	7
6	설계 상세화 및 변경 내역	7
6.1	개발 언어 및 개발 도구	7
6.2	시스템 구조	8
6.3	전체 구조도	8
6.4	시스템 흐름	9
6.5	변경 내역	9
7	갱신된 과제 추진 내역	9
8	구성원별 진척도	10
9	보고 시점까지의 과제 내용 및 중간 결과	12
9.1	완성된 기능	12
9.2	추후 개발 기능	12
9.3	중간 보고 시점까지의 결과물	13
10	참고 문헌	15

2 연구 배경

2.1 초·중등교육과정 코딩교육 의무화

		현재	'22~'25	'25~
교육과정	초·중학교	초·중학교 SW교육 필수화 초등 5~6학년: 17시간 중: 34시간	AI 교육 선도 학교 확대	초·중학교 정보교과 시수 확대 초등: 34시간 이상/ 중: 68시간 이상 학교 자율 정보시수 증배 기제 마련
	고등학교	SW·AI 선택과목 신설 AI 융합교육 중심고 운영		공동교육과정 운영 확대 SW·AI 선택과목 확대
	교원	교육대학원 연계 AI 융합교육 전문과정 도입 (석사학위)	현직교원 역량 강화 지원	일정규모 이상 학교에 정보교육 담당 교원 배치

Figure 1 2022.08. 디지털 인재양성 종합방안, p19

2022년 8월, 교육부에서 발표한 '디지털 인재양성 종합방안'에 따르면, 2025년부터 초·중등교육에서 정보과목 확대 및 코딩교육을 의무화함. 현 교육과정에서는 'CODE(code.org)'나 네이버 커넥트재단에서 개발하고 운영하는 '엔트리' 등의 '블록코딩'을 통해 학생들이 코딩에 대해 처음 접하게 됨.

이후, 중·고등학교에서 python이나 javascript 등의 프로그래밍 언어를 학습하며 간단한 실습을 통해 언어를 배우고 자료구조 및 알고리즘, 데이터 처리 등의 내용을 습득함. 또한, 한국교육개발원의 교육기본통계에 따르면 2024년 기준 초등학교 249만명, 중학교 133만명, 고등학교 130만명으로 총 513만명의 학생들이 코딩교육을 바탕으로 프로그래밍을 학습할 것으로 기대됨.

2.2 취업시장 코딩테스트 확대

지난 2024년 개발자 채용 규모는 약 14만 건으로, 매년 약 20만 명 이상의 개발자를 채용했던 것과 비교하여 상당히 줄은 것으로 확인됨. 그러나, 파트너 기업 1,000개에서 약 77만 명의 개발자를 검증한 코딩테스트 운영 서비스인 '프로그래머스'의 이용자 수치를 바탕으로, IT 개발 직군에서는 코딩테스트를 활발히 사용하고 있음. 최근, '토스' 등의 기업에서는 오직 본인의 작업물 공유 서비스인 '깃허브'의 레포지토리로 채용을 진행하는 등 개발 실력만을 평가하는 회사도 존재함.

많은 분야가 IT 기술과 접목됨에 따라, 개발 직군이 아닌 비개발직군에 있어서도 코딩 기본 내용을 질문하거나, 개발 직군에 비해 난이도가 낮은 코딩 테스트를 시행하여 채용을 진행하는 회사도 확인이 되고 있음.

3 문제 상황

3.1 코딩 교육의 어려움

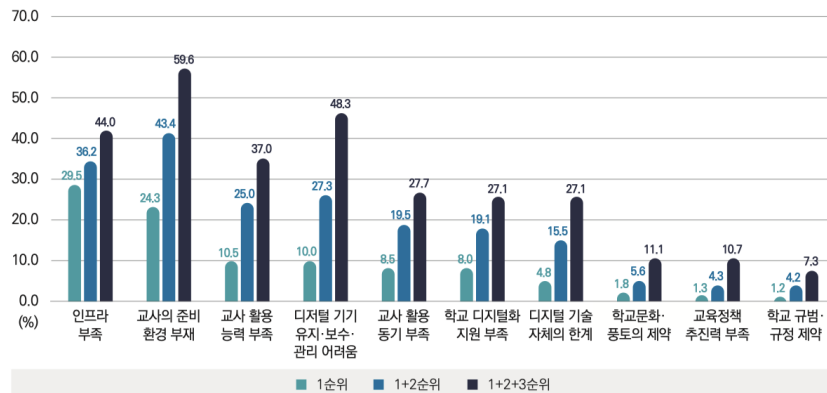


Figure 2 2023.01. 학교현장의 디지털 전환 실태와 과제, p.3

2023년 한국교육개발원에서 발표한 '학교현장의 디지털 전환 실태와 과제'의 결과에 따르면, 1~3순위를 모두 합산한 결과에서 약 60%의 교사가 준비 환경 부재를 현장에서 디지털 교육의 어려움으로 꼽음. 이미 현장에서 업무 부담이 높고, 시간이 부족하므로 새로운 디지털 기술을 수용하고 교육하기에 어렵다고 인식하고 있음.

3.2 기존 프로그래밍 학습사이트의 단점



Figure 3 프로그래밍 학습 사이트

백준, 프로그래머스와 같은 국내 사이트는 물론이고, 해외에서 제공하는 LeetCode 등의 서비스에서 제공한 문제를 사용자가 풀고, 채점 및 레이팅 기능을 사용하는 경우가 많음.

그러나, 사용자가 문제를 풀이하는 도중 특정 지점에서 막혔거나, 풀이 방법을 모른다면 직접 검색 또는 생성형 AI로 검색해야하는 어려움이 존재함. 또한, 난이도에 따른 레벨 분류 기능을 제공하는 사이트는 있지만, 사용자가 풀이한 문제를 바탕으로 유사한 문제를 추천해주는 기능과 추가적으로 문제 풀이를 원한다면 문제를 생성하여 제공하는 기능 등은 구현되어 있지 않음.

4 과제의 목표

4.1 착수보고서에서 정의된 목표 기능

본 과제에서는 사용자가 알고리즘 문제 풀이를 진행하면서 어렵거나 도움이 필요한 부분에서 사용가능한 힌트를 제공하는 것과 더불어, 사용자 제출 기록 기반 유사 문제 추천 시스템 및 문제 생성 기능을 구현하고자 함.

동일한 문제를 풀이한 다른 사용자의 풀이방법과 문제 정보를 참고하여 생성형 인공지능으로 힌트를 제공하며, 유사 문제 생성을 통해 본인에게 필요한 문제를 다양한 유형으로 풀이할 수 있는 환경을 제공할 예정임.

추가로, 부정행위 방지 시스템을 도입하거나 교내 코딩과제 제출 플랫폼으로 사용할 수 있도록 확장 가능성을 열어두어, 다양한 기능을 갖춘 서비스로 구현할 예정임.

5 요구사항 및 제약 사항 분석에 대한 수정사항

5.1 요구사항 및 제약 사항 분석 및 전문가 자문의견

착수보고서 작성 시 목표로 했던 기능 중 '교내 과제 제출 플랫폼으로의 확장'에 대한 수정이 필요하다고 판단함.

그 이유는 우리 CodeSphere팀에서 완성하고자 하는 서비스의 주 기능이 과제 제출 및 이에 따른 자동채점 및 부정행위 방지가 아닐 뿐더러, 알고리즘 학습이 필요하다고 판단되는 사용자가 흥미를 갖고 지속적으로 학습을 할 수 있도록 '힌트 제공'이나 '사용자 데이터 기반 문제 추천 시스템 및 문제 생성'의 완성도를 높여야겠다고 생각했기 때문임.

착수보고서에서 사용자 데이터 기반 '문제 추천 시스템' 및 '문제 생성' 부분을 깊게 정의하지 않았기 때문에, 중간보고서 이후 최종보고서 마무리까지 시간동안 문제 생성부분에 있어서 RAG 기술과 Prompt Engineering을 발전시켜 생성형 AI에 있어서 가장 큰 문제인 '할루시네이션' 문제를 최소화하기 위한 다양한 방법을 적용시켜보고자 함.

5.2 수정된 기능 목표 정의

① 알고리즘 문제 풀이 중 생성형 AI기반 '힌트 제공' 기능

- 기존에 존재하는 다양한 알고리즘 문제 풀이 사이트(예. 백준, 프로그래머스, 리트코드 등)에서 문제를 크롤링하여 문제 DB를 구성함.
- 또한, 문제에 대한 다른 사용자의 풀이를 크롤링하여 DB에 저장하고, 사용자의 문제 풀이 중 힌트 요청시 다른 사용자의 풀이와 알고리즘 관련 논문 및 서적을 통한 RAG 기술로 힌트를 제공함.

② 사용자가 풀이한 문제 기반 '문제 추천 시스템' 기능

- KNN을 통한 그룹 분석과 사용자별로 정답률이 낮은 문제를 통해 취약한 문제 유형을 추천할 수 있음.

③ 생성형 AI기반 '문제 생성' 기능

- Prompt Engineering과 RAG 기술을 활용한 생성형 AI를 통해 문제를 생성함.
- 생성한 문제는 문제 DB에 저장하여, 다른 사용자들도 함께 학습에 사용할 수 있도록 데이터를 관리하고자 함.

6 설계 상세화 및 변경 내역

6.1 개발 언어 및 개발 도구

6.1.1 개발 언어

- Front-end : HTML, CSS, JavaScript, TypeScript
- Back-end & AI Server : Python

6.1.2 개발 도구

- Front-end : React, Vite, Tailwind CSS, with ESLint
- Back-end : FastAPI, PostgreSQL, NginX
- Infrastructure : AWS(EC2, S3), Redis, Docker
- AI Server : FastAPI, Langchain
- LLM : ChatGPT-4.5, ChatGPT-4o, ChatGPT-o4-mini-high

6.2 시스템 구조

6.2.1 사용자 인터페이스(Front-end)

- Vite + React + tailwind CSS를 사용하여, 유사 서비스인 '프로그래머스'에서 제공하는 코드 IDE창을 구현하여 사용자 친화적으로 화면을 구성함.
- 사용자가 사이트에서 직접 문제를 풀이하면서 우리 서비스의 핵심 기능인 '힌트 제공'을 사용함으로써 학습의 능률을 높일 수 있도록 제작함.

6.2.2 서버 애플리케이션(Back-end)

- 개발 단계에서는 온프레미스 서버를 활용하여 비용을 절약하고, 어느정도 서비스 개발이 완료되면 AWS 클라우드 서비스의 EC2를 사용하여 많은 사용자가 접속하더라도 서비스 장애가 발생하지 않도록 Auto Scaling을 적절히 활용하고자 함.
- FastAPI를 사용하여 백엔드 서버를 구동하고, 힌트 생성이나 문제 생성 등 생성형 AI를 활용한 기능이 필요한 부분은 AI 서버를 별도로 운영하여, 원활한 서비스가 구현되도록 할 예정임.

6.2.3 데이터베이스(DB)

- 기존의 알고리즘 문제들을 크롤링하여 Amazon RDS로 PostgreSQL 환경에 저장하고, 백엔드 서버와 통신하여 기능을 구현함.

6.3 전체 구조도

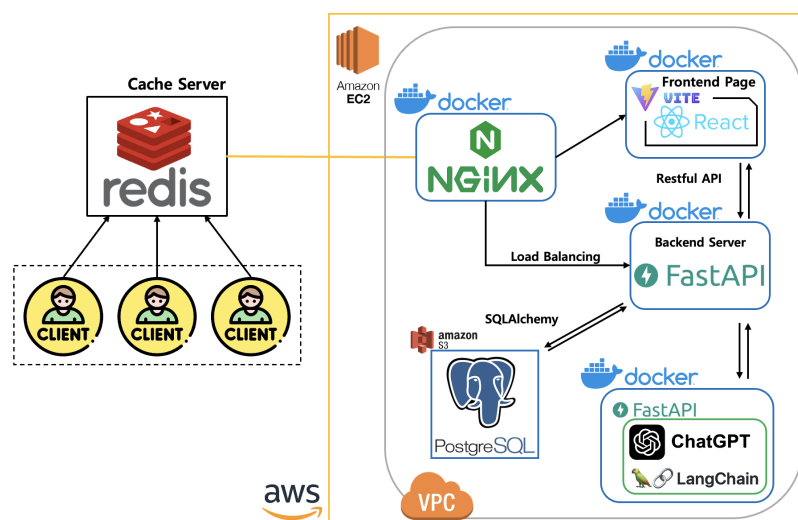


Figure 4 서비스 구조도

6.4 시스템 흐름

- 사용자는 브라우저를 통해 React로 구현된 서비스 홈페이지에 접속함.
- 최초 사용자는 회원가입 프로세스를 진행하고, 서비스를 이용하기 위해서는 로그인을 진행해야함.
- 비회원 및 비인가된 사용자의 서비스 요청은 차단됨.
- 인가된 사용자의 서비스 요청(예. 힌트 요청, 채점 요청 등)은 백엔드 서버인 FastAPI에 전달되어, 생성형 AI가 필요한 요청인 경우 AI 서버에 별도로 요청하는 과정을 거침.
- 위와 같은 과정을 거친 값은 다시 프론트엔드로 전달되어 사용자가 결과를 확인할 수 있도록 함.

6.5 변경 내역

- 우리 서비스의 주 기능인 '문제 채점' 및 '힌트 제공', 그리고 사용자가 풀이한 데이터를 기반으로 한 '문제 추천 시스템' 및 '문제 생성' 기능을 강화하는 방향으로 변경하였음.
- 착수보고서에서 언급한 '과제 제출 플랫폼'으로의 확장보다, 서비스의 만들새를 강화하며, 최신 기술을 활용한 완성도 높은 서비스를 구현하고자 함.

7 갱신된 과제 추진 내역

7.1 개발 일정

대분류	작업	5				6				7				8				9			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
기획	주제 선정																				
	자료 조사																				
설계	데이터베이스 설계																				
	데이터 구조 정의																				
데이터 구축	문제 데이터 수집																				
	문제 생성 연구																				
디자인	메인 페이지 디자인																				
	IDE 페이지 디자인																				
개발	프론트 개발																				
	백엔드 개발																				
	힌트 생성 AI 개발																				
	문제 생성 AI 개발																				
	추천시스템 개발																				
테스트	유닛 테스트																				
	통합 테스트																				
	부하 테스트																				
고도화	고도화 및 안정화																				

8 구성원별 진척도

8.1 구성원별 역할

이름	내용
김대욱	(팀장) Project Management, Backend Engineer 서비스 아키텍처 및 데이터 구조 설계, FastAPI 백엔드 및 AI 서버 개발, 추천시스템 개발
김문경	(팀원) Frontend Engineer 타 사용자 데이터 수집, 웹 UI/UX 구성 및 디자인, 프론트엔드 개발
김진우	(팀원) AI Engineer 문제 데이터 수집, 문제 힌트 및 생성 연구, 추천시스템 개발

8.2 구성원별 진척도

이름	내용
김대욱	<ul style="list-style-type: none"> PM <ul style="list-style-type: none"> 팀원별 작업 분배 및 일정 관리 아키텍처 설계 및 데이터 구조 설계 Back-end <ul style="list-style-type: none"> Home 페이지 개발 <ul style="list-style-type: none"> 오늘의 추천 문제 리스트 (문제 제목, 번호, 난이도, 카테고리 등) 회원가입/로그인 기능 (7월 내 구현 예정) SolveProblem 페이지 개발 <ul style="list-style-type: none"> 문제 설명 영역 (문제 제목, 태그, 난이도, 문제 설명, 예제 입출력) 힌트 제공 기능 (IDE창 내 힌트 요청시 힌트 제공, AI 서버와의 연동을 제외한 부분 완성) 채점 기능 (7월 내 구현 예정)
김문경	<ul style="list-style-type: none"> Front-end <ul style="list-style-type: none"> 와이어프레임 기반 UI 개발 Home 페이지 개발 <ul style="list-style-type: none"> 오늘의 통계 및 정보 페이지 오늘의 추천 문제 리스트 (문제 제목, 번호, 난이도, 카테고리 등) 로그인 / 회원가입 영역 이달의 랭커 (전체 순위 중 1~3등 유저 표시)

-
- **ProblemBoard** 페이지 개발
 - 필터 기능 (난이도, 카테고리별)
 - 검색 기능 (검색창, 정렬 드롭다운)
 - 전체 문제 리스트 (문제 제목, 번호, 난이도 등)
 - **SolveProblem** 페이지 개발
 - 문제 설명 영역 (문제 제목, 태그, 난이도, 문제 설명, 예제 입출력)
 - 코드 에디터 영역 (언어선택, 힌트보기, 코드작성)
 - 콘솔 영역 (코드 에디터, 콘솔 탭 전환)
 - 제출 결과 모달 팝업 (제출 결과 요약 제공)
 - 코드 작성 기능 페이지 구현
 - CodeMirror IDE 툴 적용
 - 코드 작성 및 편집 인터페이스 구축
 - 관련 컴포넌트 및 로직 분리 구현

-
- 데이터 전처리
 - 데이터 크롤링 코드 작성 및 DB 연결
 - BeautifulSoup 기반 HTML 파싱으로 BaekJoon 문제 데이터 수집
 - Test data의 I/O 생성 및 저장
 - 각 문제의 테스트케이스를 생성한 후 JSON 형식으로 직렬화
 - 추후 채점용 데이터로 활용

김진우

- AI
 - LLM 기반 자동 힌트 생성 시스템 개발
 - GPT-4o모델 활용하여 알고리즘 문제에 대한 단계별 힌트를 제공하는 Prompt Engineering 수행
 - RAG 구조 적용
 - 현재 DB에서 문제 정보 및 풀이 코드를 검색하여 프롬프트에 삽입하는 방식으로 RAG 구조 적용
 - 문제-풀이 정보를 벡터로 인덱싱하고, Dense Retriever + Generator가 연동된 실제 RAG 파이프라인 구축(7월 내 구현 예정)
-

-
- 실시간 코드 추적 시스템 개발
 - 사용자가 코드를 입력하면(현재는 한 줄 단위 기준), 이를 누적하여 흐름을 파악하고 힌트를 단계적으로 제공
 - 사용자별 코드 상태를 분석하여 맞춤형 힌트가 동적으로 제공되는 시스템 개발(7월 내 구현 예정)
-

9 보고 시점까지의 과제 내용 및 중간 결과

9.1 완성된 기능

- 메인 페이지
 - 회원가입/로그인 화면 제공 (기능 미연결)
 - 오늘의 추천 문제 제공 (추천시스템 기능 미구현)
- 문제 풀이 IDE 페이지
 - 문제 정보 표출
 - 문제 풀이 영역 구현
 - 힌트 제공 및 채점 요청 기능 (세부 기능 미구현)
- 채점 후 모달창
 - 통과된 테스트 케이스 및 풀이 여부 제공

9.2 추후 개발 기능

- 회원가입/로그인 로직 추가
- 알고리즘 관련 논문 및 서적 기반의 정보를 활용하여, RAG(Retrieval-Augmented Generation) 기술을 통해 보다 정교하고 근거 기반의 힌트를 제공하는 시스템 구현
- 사용자의 풀이 현황과 진행 상태에 따라 개인 맞춤형 힌트를 제공하는 기능 개발
- 개인의 학습 성향 및 문제 풀이 데이터를 분석하여, 맞춤형 문제 추천 시스템 개발

9.3 중간 보고 시점까지의 결과물

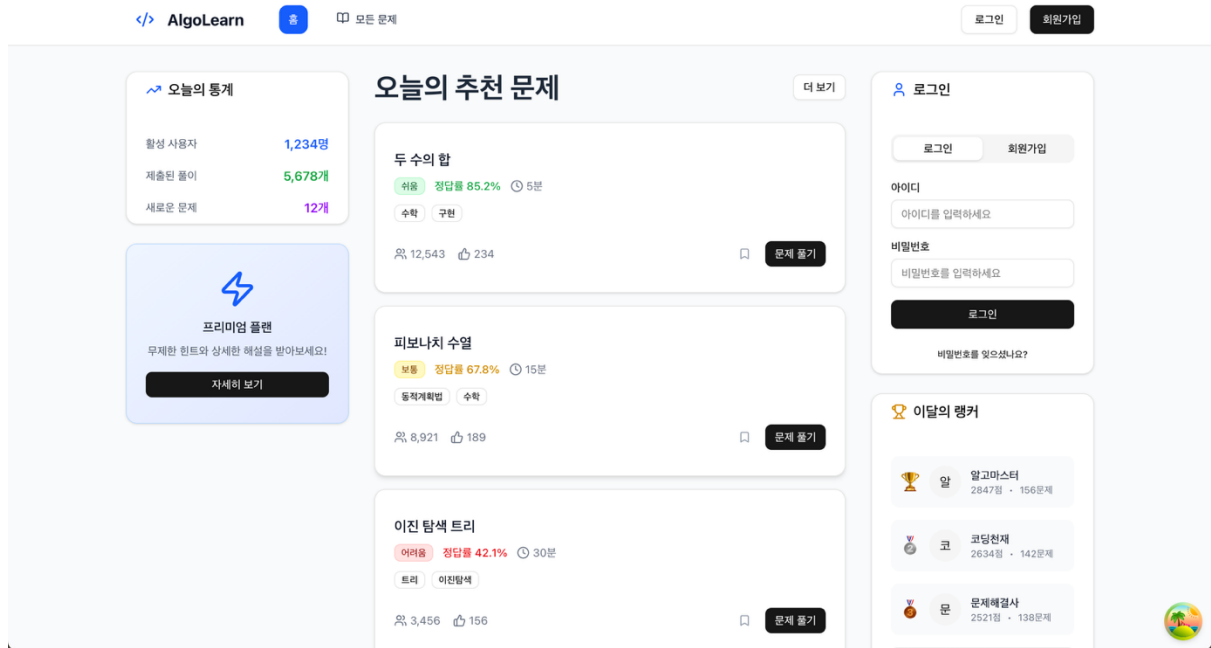


Figure 5 Home 페이지

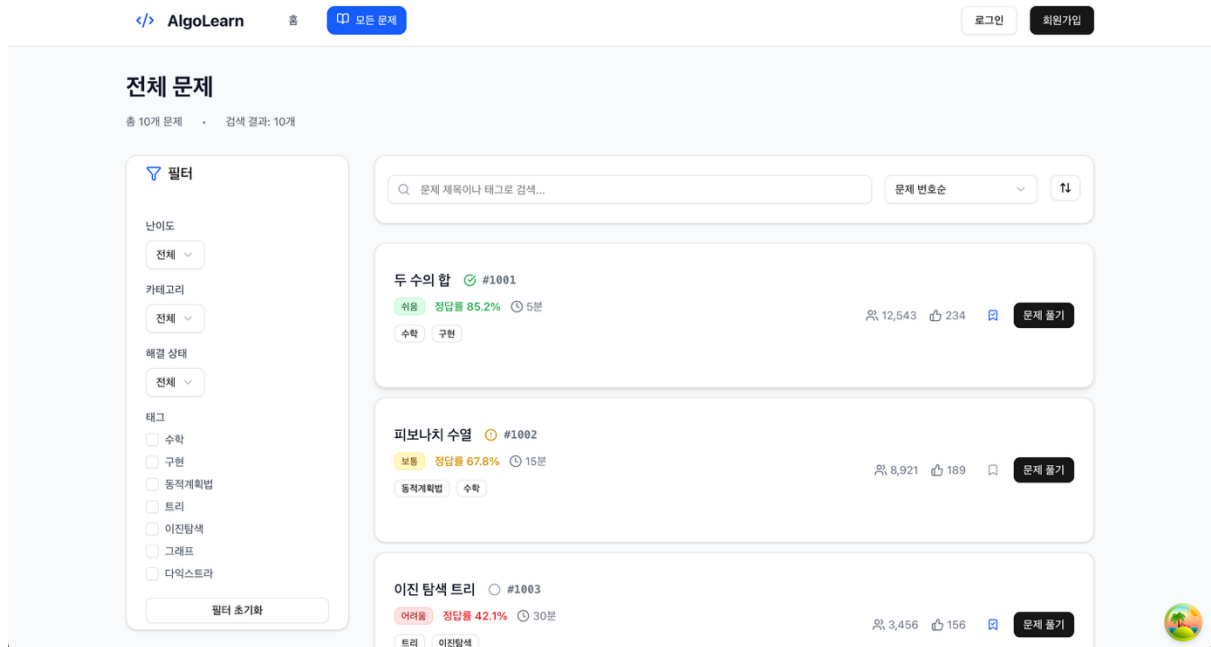


Figure 6 ProblemBoard 페이지

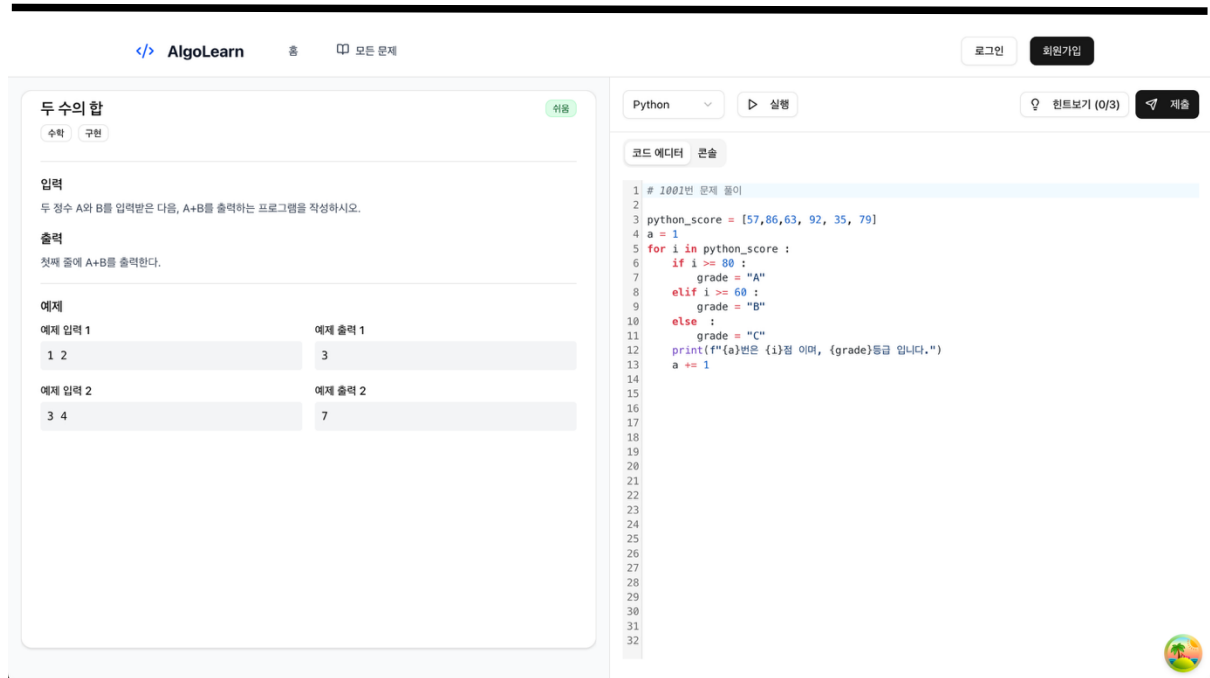


Figure 7 SolveProblem 페이지

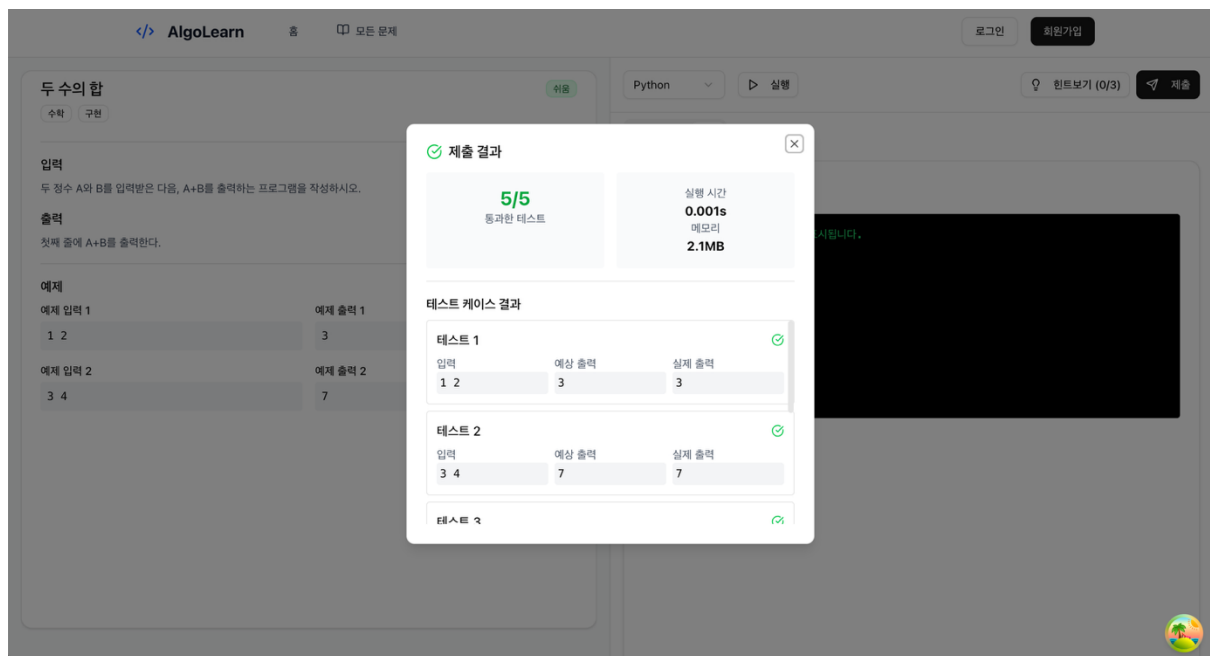


Figure 8 SolveProblem 페이지 내 제출 결과 모달

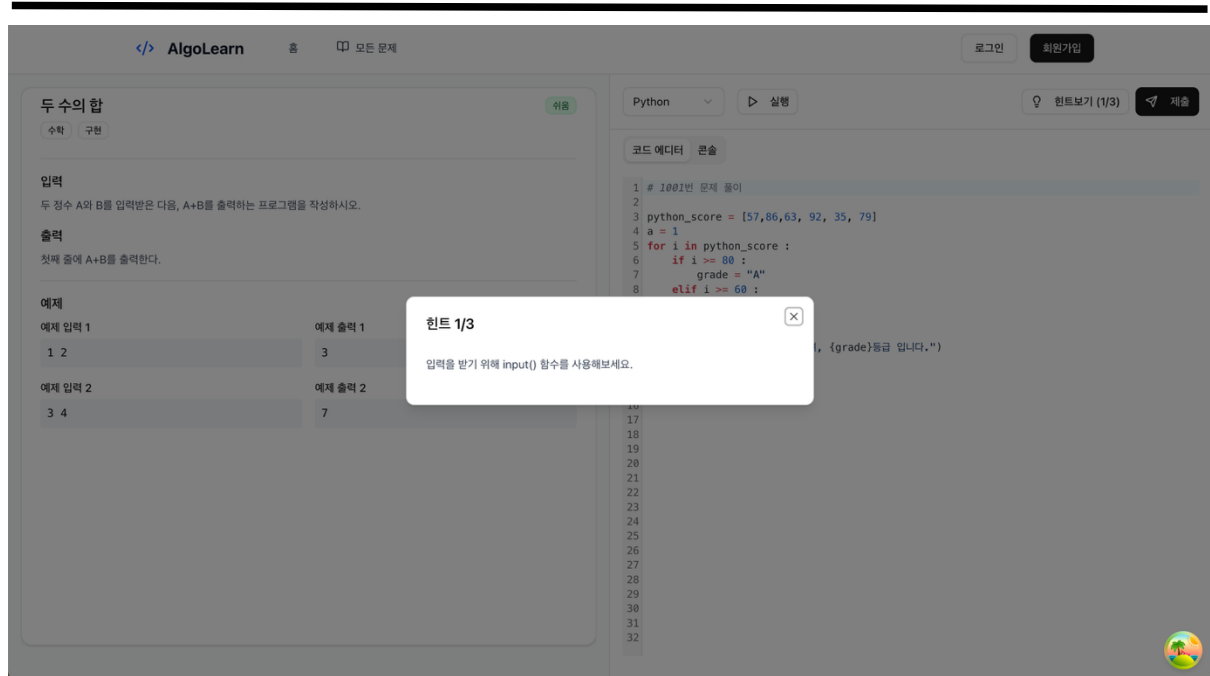


Figure 9 SolveProblem 페이지 내 힌트 내용 모달

10 참고 문헌

- 교육부, 2022, 디지털 인재양성 종합방안
- 한국교육개발원, 2023, 학교현장의 디지털 전환 실태와 과제
- Lewis, Patrick, et al. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- Wei, Jason, et al. "Chain of Thought Prompting Elicits Reasoning in Large Language Models." *NeurIPS*, 2022.
- OpenAI Cookbook, "Best practices for prompt engineering with OpenAI API"