

알고리즘

개요

이영석

yslee@cs-cnu.org

II. 교과목 개요(Course Overview)

1. 수업개요

- 비대면(원격)수업으로 운영함
- 비대면(원격)수업: 동영상 + 화상회의
- 주의사항: 코딩테스트를 매주 실시하는 이론+실습 결합 수업으로 알고리즘을 학습!
- 코로나 상황에 따라 비대면과 대면 방식을 변경가능함

2. 선수학습내용

컴퓨터프로그래밍1, 컴퓨터프로그래밍2, 자료구조 및 실습

3. 수업방법(%)

강의	토의/토론	실험/실습	현장학습	개별/팀 별 발표	기타
50		50			

4. 평가방법(%)

중간고사	기말고사	퀴즈	출석	과제	포트폴리오	참여도	기타
25	25		5	45			

III. 교과목표(Course Objective)

교과목의 목표는
자료구조 기초, 알고리즘 기초 및 중급(트리, 그래프, 검색/탐색, 정렬, 동적프로그래밍 등)에 대해서
코딩테스트의 중급 실력향상을 목표로 한다.
이 목표 달성을 위하여 매주 알고리즘 주요 코딩테스트를 실시합니다.

구분	구분	구분
1	학습목표 및 주요학습내용	개요
	과제	코딩테스트연습
	기타 사항	
2	학습목표 및 주요학습내용	자료구조기초
	과제	코딩테스트
	기타 사항	
3	학습목표 및 주요학습내용	알고리즘 복잡도
	과제	코딩테스트
	기타 사항	
4	학습목표 및 주요학습내용	정렬
	과제	코딩테스트
	기타 사항	
5	학습목표 및 주요학습내용	트리1
	과제	코딩테스트
	기타 사항	
6	학습목표 및 주요학습내용	트리2
	과제	코딩테스트
	기타 사항	
7	학습목표 및 주요학습내용	그래프1
	과제	코딩테스트
	기타 사항	
8	학습목표 및 주요학습내용	중간고사
	과제	
	기타 사항	
9	학습목표 및 주요학습내용	탐색1 그래프2
	과제	코딩테스트
	기타 사항	
10	학습목표 및 주요학습내용	동적프로그래밍1
	과제	코딩테스트
	기타 사항	
11	학습목표 및 주요학습내용	동적프로그래밍2
	과제	코딩테스트
	기타 사항	
12	학습목표 및 주요학습내용	그리디 알고리즘
	과제	코딩테스트
	기타 사항	
13	학습목표 및 주요학습내용	문자열
	과제	코딩테스트
	기타 사항	
14	학습목표 및 주요학습내용	탐색2
	과제	코딩테스트
	기타 사항	스크린샷
15	학습목표 및 주요학습내용	기타사항

학습 목표

- 문제해결 전략 학습
 - Exhaustive search (완전 탐색) or Brute-force search
 - Divide and conquer (분할 정복)
 - Dynamic programming (동적 계획법)
 - Greedy algorithm (탐욕적 해결법)
- 자료구조 활용 문제해결 학습
 - 핵심 자료구조: 힙, 해쉬, 트리, 그래프, 문자열
 - 언어별 STL (Standard Template Library)
- 제한조건 이해 및 문제 해결
 - 복잡도: 시간, 공간
- 코딩 테스트
 - 취업 1차 통과 관문
 - IT/SW기업: default
 - 금융공공기관
- 컴퓨터공학 문제 해결 능력
 - 알고리즘은 일부
 - 동시접속자 10만 서버 ?
 - 수억 가입자 검색속도 향상?
 - IP라우팅 테이블 검색 속도 향상?
 - 웹페이지 검색 속도 향상?
 - 영화/쇼핑몰/광고 추천 정확도 향상?

수업 주의 사항

- Python3 언어로 진행함
 - C/C++, Java 언어는 코딩테스트를 허용하지만, 질문 대답에서는 우선 순위가 떨어지고 대답이 늦거나 없을 수 있음 (질문 수가 많기 때문)
- 실습시간 준비
 - 스마트폰 + 노트북/컴퓨터
 - Closed web search/social network (Language reference 사이트만 허용)

코딩 테스트: 기초 문제 해결 능력 점검

- IT/SW 취업

- 카카오, Naver, NHN, SK, KT, 삼성 등
- 현대자동차
- 조폐공사

- 문제출제 목표

- 자료구조 활용
 - 스택, 큐, 해시, 배열, 리스트
- 문제에 따른 정확한 알고리즘
 - Brute force, BFS, DFS, DP, Greedy, Tree, Graph, Flow, Recursion
- 알고리즘 성능(복잡도 이해)
 - 메모리 계산: 배열 vs. 리스트, 변수 타입의 최대 크기
 - 시간 제한: Recursion, 동적프로그래밍, 자료구조(해시), 알고리즘(병합정렬)

✓ 10. 코딩 테스트

온라인 혹은 오프라인에서 컴퓨터로 직접 문제를 푸는 것을 의미한다.

2000년대 후반까지만 해도 국내 기업에서는 거의 코딩 테스트가 없었다. 그러나 2010년대 들어서 대부분의 SW직무 및 기업은 인적성을 코딩테스트로 대체하는 곳이 많다. 6번 취업대상 문단에서 언급된 기업 대부분 역시 인적성 대신 코딩테스트로 그 전형을 대신한다. 코딩테스트를 통과하면 면접을 보며 면접시 자신이 코딩테스트 때 풀 문제의 코드를 중심으로 면접을 진행하는 편이다. 물론 코딩테스트 외에 자소서 내용도 추가적으로 물어 본다.

신입을 뽑는 코딩 테스트라고 해서 만만한 것은 아니다. 대한민국 95% 이상의 대학교에서는 컴퓨터 공학과에서조차 이러한 알고리즘 문제를 해결하는 방법을 가르치지 않는다. 심지어 컴퓨터 과학 **박사**나 업계에 근무하는 시니어급 **경력**자들도 코딩 테스트용 문제를 풀다가 낭패를 보았다는 경험담이 많다. 그래서 아무리 전공자라도 꾸준히 위의 웹사이트에 나온 문제들을 풀어보면서 숙달되어야 한다.

삼성전자 SW직무는 그룹사 전체에 적용되는 인적성인 **GSAT**를 시행하지 않고 오프라인 코딩테스트로 대체한다. IDE는 2018년 하반기 기준 C/C++ **비주얼 스튜디오** 익스프레스 2013, Java **Eclipse** Mars, 파이썬 **PyCharm** 2018.2.3이다.

<https://namu.wiki/w/%EC%B7%A8%EC%97%85/SW>

알고리즘과 코딩 테스트

- 주제
 - Tree, Graph
 - BFS, DFS, shortest path, MST, Maximum flow
 - Brute-force, Recursion, Divide-and-Conquer, DP, Greedy, String, Math
 - 복잡도

정리

- 코딩 테스트 알고리즘 수업