

# 취업 준비 스터디 1주차 멘토링

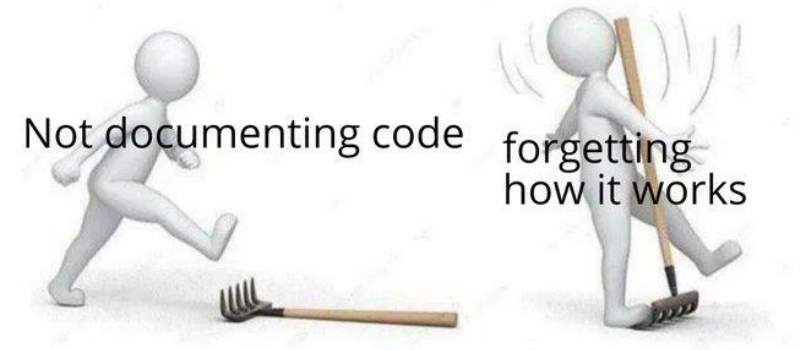
2022.09.14



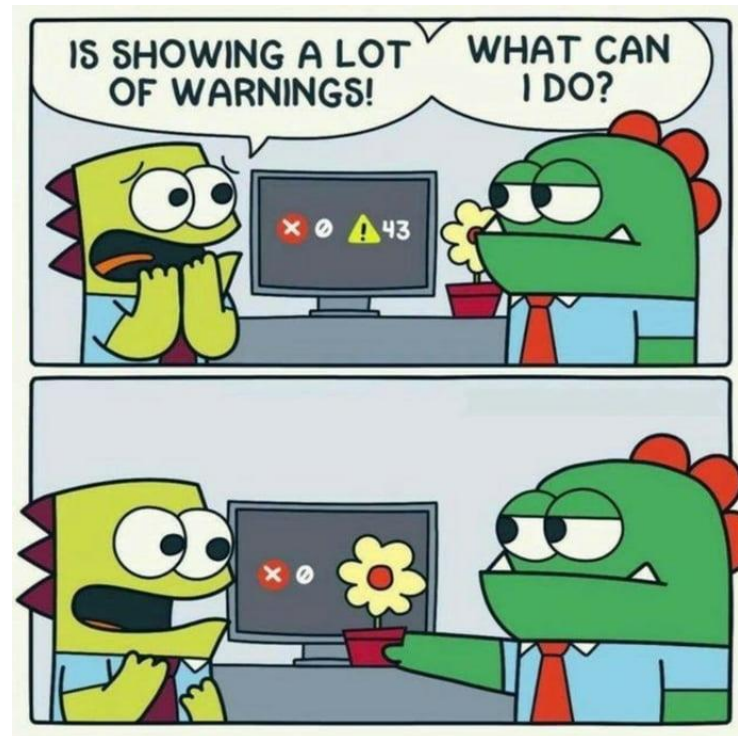
발표자 : 박동학

1. 스터디 소개
2. 개발자
3. 서류 전형 준비하기
4. 코딩테스트 준비하기
5. Git 사용법 설명
6. QnA

Junior devs:



Senior devs:



# 스터디 소개

## Member

Mento	박동학	박민혁			
Mentee	황승환	김채은	최재민	강세준	주연
	이정현	차현민	김정민	최지윤	허경환
	목요빈	고산하	이준석	지누	최승용
	이서현	지민	으뉴	쥬원	조연주

✅ 과제 진행현황 ✅

Name	Pre	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12
황승환	➡ soon												
김채은	➡ soon												
최재민	➡ soon												
강세준	➡ soon												
주연	➡ soon												
이정현	➡ soon												
차현민	➡ soon												
김정민	➡ soon												
최지윤	➡ soon												
허경환	➡ soon												
목요빈	➡ soon												
고산하	➡ soon												
이준석	➡ soon												
지누	➡ soon												
최승용	➡ soon												
이서현	➡ soon												
지민	➡ soon												
으뉴	➡ soon												
쥬원	➡ soon												
조연주	➡ soon												

## 스터디 개요

→ 취업 목적의 스터디 (약 12주 3개월 계획 중)

→ 주 1회 멘토링을 통한 가이드 라인 잡아주기

→ 각자의 상황이 달라 원하는 분들은 **상담가능**

## 주요 멘토링

→ 취업 전체적인 가이드

→ 코딩테스트 준비

→ 면접 준비 (CS, 기술면접)

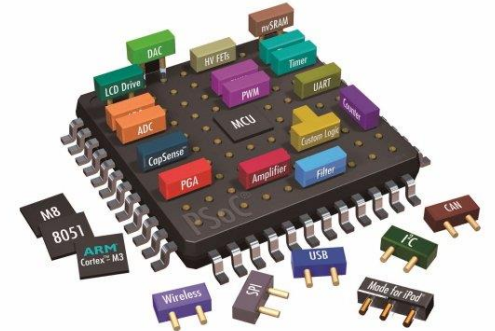
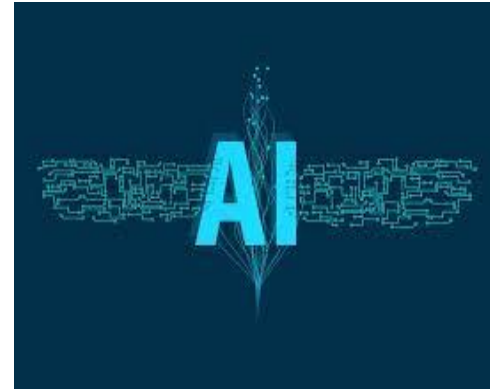
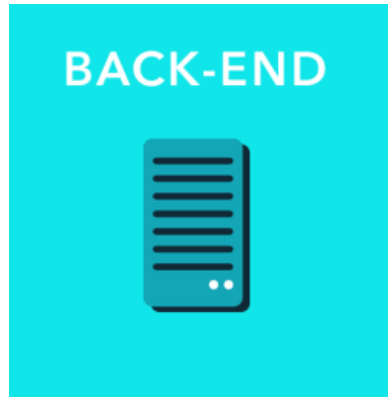
## \*당부의 말

→ 개인적인 준비를 위해 언제든지 그만두셔도 괜찮습니다.

→ 스터디이자 코칭이라고 생각해주세요.

→ 이 스터디 외에도 꼭 자신의 분야에 맞는 공부를 해주세요.

→ 백엔드, 프론트엔드 등 본인의 직무에 대한 프로젝트나 학습은 별도로 꼭 진행



**프론트엔드** - 눈에 보이는 화면 (웹 모바일 PC 게임) → 사용자 관점에서의 뷰를 개발하는 직군

**백엔드** - 안보이는 정보 처리 (서버 개발자) → 프론트엔드에서 처리되는 보이지 않는 데이터를 다루는 직군

**데이터, AI** - 데이터 처리, 파이프라이닝, 빅데이터, 머신러닝, 인공지능 → 최근 유행하고 있는 AI 관련 직군

**임베디드** - 하드웨어 기반의 개발 (미들웨어, 운영체제) → 전통적인 AP 기반의 APP이나 커널, 드라이버를 개발

**보안** - 모의해킹, 취약점 분석, 시큐어 코딩 → 데이터가 오가는 모든 곳에서 작동해야할 보안 처리 개발

개발자가 필요없는 회사 ? --> 거의 없다!!!  
본인이 어디서 일하고 싶은지 찾아봐야 합니다.

## IT 기업



## 금융 관련



## 제조 대기업



## 대기업 SI



작년 기준 많은 기업들이 5X00 + 성과급의 연봉을 제시

2022년 연봉 협상이 끝난 지금 TOP이라고 불리는 회사는 평균 5000 후반대의 계약 연봉

+ 주식, 성과급, 등등으로 1000 ~ 3000만원의 추가보상 지급중 !!

→ 삼성은 개발자 유출방지를 위해 특정 조건 부합시 추가 수당 지급

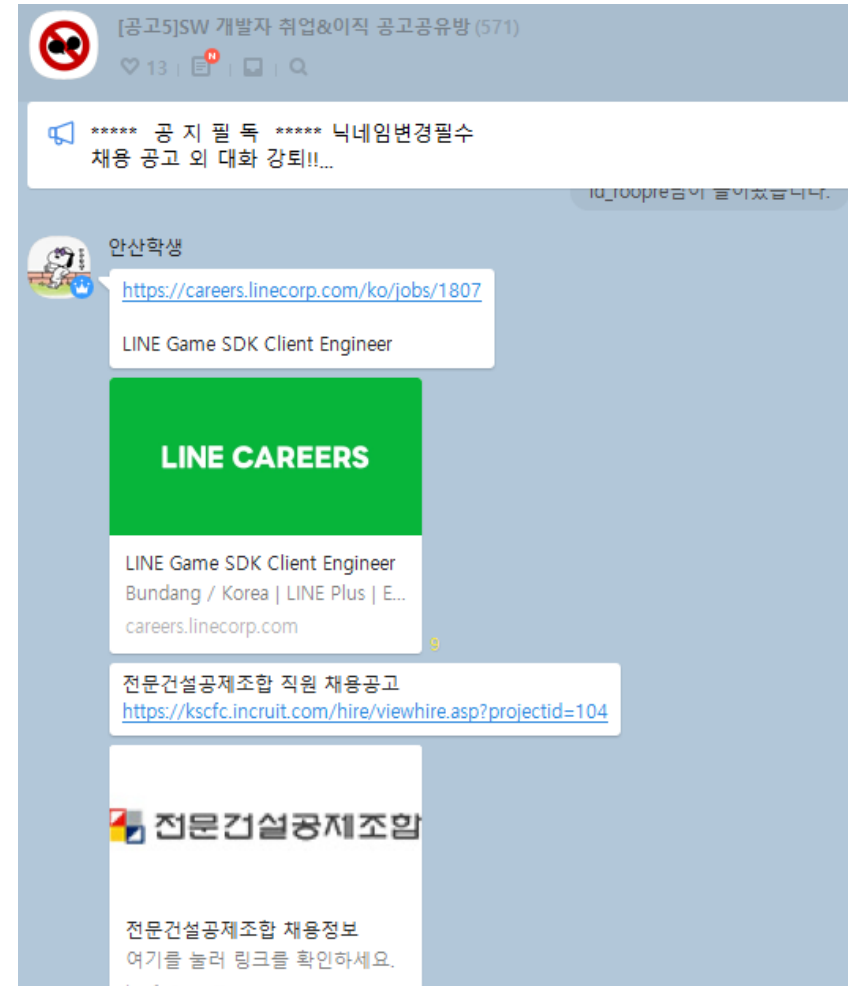
→ 현대모비스는 알고리즘 대회, 개발자를 위한 활동 등으로 개발자 우대 중

딜리버리히어로 코리아	배달	5100	?	?	?	3%	?	2021년 연봉 인상률 3배
당근마켓	중고거래/금융	6500	없음	?	있음	?	O	사이닝 보너스 있음
비바리퍼블리카	금융	?	최소 10%	있음	-	3%	X	통신비 월10, 체력단련비 월10, 점심/저녁비용 법인카드 100%지원
두나무	금융	최소 5000	있음	?	10%	?	?	초봉 SKY 기준
뱅크샐러드	금융	원천 6000	있음	-	-	?	?	리드급 스톡옵션 최소 1억, 경력 이직 기본급 1.5배 (2021년)
직방	부동산	원천 6000	?	?	?	?	O	이직 시 사이닝 보너스 직전 계약 연봉 100% (최대 1억), 원격 근무
리디북스	전자책	5000	?	?	?	?	?	시니어/PM 이직 시 사이닝 보너스 5000만, 직전 연봉 30% 인상 (현직 포함)
야놀자	레저	4000 ~ 5000	?	?	?	?	O	우리사주 1000만, 사이닝 보너스
채널톡	메신저	원천 6500	?	?	?	?	?	
디어유	메신저	5000	?	?	?	?	?	
쏘카	카셰어링	4200	?	?	?	8 ~ 10%	?	
그린랩스	스마트팜	?	?	?	있음	?	?	이직시 직전 계약 연봉 최대 130%, 최대 5000만원 사이닝 보너스
네이버	검색 / 플랫폼	5000	15 ~ 25%	1000만	1000만	3 ~ 10%	X	주5재택 vs 주3출근 선택, 2021년부터 3년 간 연 1000만원 상당 주식
카카오	메신저 / 플랫폼	5500	5 ~ 20%	500만	?	?	X	주5재택, 격주 출근 가능
카카오엔터테인먼트	콘텐츠	5000	5 ~ 20%	500만	있음	3 ~ 10%	X	안식 휴가 30일, 휴가비 200만원, 복포 360만, 명절 100만, 통신비 지원
라인플러스	메신저 / 플랫폼	5000	평균 15%	있음	있음	3 ~ 10%	X	영구 재택, 재택 지원금 월 17만, 복지포인트 256만, 신입 사이닝 보너스
TmaxSoft	미들웨어	5000	0	?	?	?	?	연구원 기준 (이후 5년 동결)
GS 리테일	유통	4600	?	?	?	?	?	
GS 네오텍	클라우드	4290	15%	-	-	?	?	자격증 수당 540만

## 지원할 회사 탐색



# 자소설닷컴



개발자 공고는 주로 자소설 닷컴이나 오픈톡방에 많이 올라옵니다.  
몰라서 지원하지 못하는 경우가 많은데 이를 방지 할 수 있어서 좋습니다.



## 전형 분석하기

# LINE

### 전형 절차

#### - Engineering

: 지원서 접수 > 코딩 테스트 > 필기 테스트 > 1차 면접 > 2차 면접 > 채용검진 > 입사

#### - Planning, Business, Design

: 지원서 접수 > 과제 테스트 > 1차 면접 > 2차 면접 > 채용검진 > 입사

# SAMSUNG

### 전형절차 및 일정





## 직무기술서 보기

### Requirements

- 전산/컴퓨터, 전기전자, 산공, 기계, 통계(이공), 수학, 물리 등 관련 전공자
- Machine Learning / Deep Learning 알고리즘 설계 및 구현 역량 보유자
- 개발에 사용되는 프로그래밍 언어(C/C++/Java 등)에 대한 이해 및 알고리즘 문제 해결 역량 보유자
- 유무선 통신 네트워크 관련 지식 기반 통신 SW 개발 역량 보유자

### Pluses

- 직무와 연관된 대내외 활동 및 수상 이력 보유자 (알고리즘 전국대회 수상자, 오픈소스 프로젝트 등)
- 직무와 연관된 우수 국내외 학회 논문 게재 및 특허 출원 이력 보유자
- 해외 법인과 커뮤니케이션 가능한 수준의 외국어 회화 역량 보유자
- SW/HW 플랫폼을 활용한 프로젝트 수행 능력 보유자
- x86 SIMD 아키텍처, GPU 기반 멀티 코어 프로그래밍 및 최적화 경험 보유자
- AR/VR/Game 등 Interactive Streaming 응용 서비스 개발 경험자
- AI 를 활용한 LTE, 5G 통신시스템 성능 최적화 Project 수행 경험자

### 모집 직무

※ 직무 간 중복지원이 불가능합니다

담당 업무	<div>· 국내 1위 네이버웹툰, 웹소설, 시리즈, 시리즈On, 글로벌 NO.1 라인웹툰, 라인망가의 서버 개발</div> <div>· 글로벌 환경에서도 통용될 수 있는 백엔드 공통 플랫폼 개발</div> <div>· 중국/일본 등 현지 오피스 개발자들과의 기술 교류 및 글로벌 개발환경 경험</div>
자격 요건	<div>· Java &amp; Spring Framework 기반 웹 서버 개발 경험이 있는 분</div> <div>· RDBMS 나 NoSQL 개발 경험이나 지식이 있는 분</div>
Back-End 개발	<div>· git, webpack 사용 경험이 있는 분</div> <div>· 웹 서비스 인프라에 대한 이해가 있는 분 (Apache, Nginx, Tomcat, Netty, Redis, Spring Boot, Spring WebFlux, Spring batch, Kafka 등)</div>
우대 사항	<div>· Restful API 설계/구현 경험이 있는 분</div> <div>· Kubernetes 및 Docker와 같은 기술 사용 경험이 있는 분</div> <div>· Message Queue 및 Elasticsearch 사용 경험이 있는 분</div> <div>· React, Vue, Angular 개발 경험이 있는 분</div>

## 자기소개서 쓰기

### 네이버 자소서 질문

→ 기술적으로 **어려운 문제를 해결한 경험을** 상세히 작성해 주세요. (학교수업/프로젝트/경진대회/대외활동 등)

맞닥뜨렸던 문제를 '구체적'으로 기술하고, 본인의 접근 방법과 해결 과정, 그리고 실제 결과를 '상세히' 기술해 주세요.

문제를 잘 해결하셨다면 그 경험에서 아쉬운 점 혹은 더 나은 방법은 없었을지에 대한 **고민 과정을** 함께 작성해 주세요.

해결하지 못한 경험이라도 해결을 **위해 얼마나 깊이 있게 고민을 했는지 그 과정에** 대해 이야기해 주세요.

\* 코드로 설명해 주셔도 좋습니다.

#해결 못한 경험도 좋아요 #과정을 보고 싶어요

### 삼성전자 자소서 질문

→ 지원한 **직무 관련 본인이 갖고 있는 전문지식/경험**(심화전공, 프로젝트, 논문, 공모전 등)을 작성하고, 이를 바탕으로 본인이

**지원 직무에 적합한 사유를 구체적으로 서술**해 주시기 바랍니다. ( 1000자 )

## 떨어져도 실망하지 말고 많이 쓰기!!!

📄 공고명	📅 공고 마감일	📂 분류
현대모비스	2021년 3월 8일 오후 11:59 🕒	대기업
LINE+	2021년 3월 15일 오후 5:00 🕒	IT기업
KT 채용전환형 인턴	2021년 3월 29일 오후 4:00 🕒	대기업
카카오 인턴	2021년 5월 3일 오전 12:00 🕒	
God Sung Research	2021년 9월 13일 오후 5:00 🕒	
현대엔지니어링	2021년 1월 6일 오후 3:00 🕒	대기업
현대 AutoEver	2021년 2월 8일 🕒	대기업
신세계 I&C	2021년 3월 18일 오후 6:00 🕒	대기업
삼성전자	2021년 3월 22일 오후 5:00 🕒	대기업
현대로보틱스	2021년 3월 24일 오후 3:00 🕒	대기업
VCNC	2020년 12월 31일 오후 7:00 🕒	IT기업
Coding Master	2021년 1월 11일 오후 4:00 🕒	교육
Line studio → Intern	2021년 2월 1일 오전 12:00 🕒	IT기업
Kakao → Intern	2021년 2월 2일 오전 12:00 🕒	IT기업
NTS 채용전환형 인턴	2021년 3월 7일 🕒	IT기업
CJ 올리브네트웍스	2021년 4월 2일 오후 6:00 🕒	대기업
크래프톤	2021년 5월 9일 오후 11:59 🕒	
Makeus	2020년 12월 21일 🕒	교육
미래에셋대우	2021년 3월 15일 오후 5:00 🕒	금융권
가비아	2021년 3월 22일 오후 11:59 🕒	중견기업
롯데정보통신	2021년 3월 31일 오후 5:00 🕒	대기업
2021 Dev-Matching	2021년 4월 2일 오전 12:00 🕒	IT기업
네이버	2021년 4월 12일 오전 12:00 🕒	IT기업
LINE AD Platform 인턴	2021년 4월 26일 오후 5:00 🕒	
SSAFY	2021년 5월 14일 오후 5:00 🕒	

각 기업마다 원하는 인재상이 있고 시기마다 지원자들이 다양하기 때문에 최대한 많이 지원하는게 좋습니다.

다만 너무 서류 준비만 한다면 시간적으로 쫓길 수 있기 때문에 업계는 구분하는 것이 좋습니다.

저 같은 경우에는 제조업, IT, 금융, SI 모두 지원해서 면접 준비할 때 시간이 부족

## Why Coding Test !!

- 여러 기업에서 서류보다 코딩테스트를 더 신뢰하고 있습니다.
- 알고리즘만 잘해도 입사제의를 받을 수 있다!
- 요즘 추세에서 서류전형은 면접을 위한 최소 요건이 되고 있다.
- 서류(간단히) - 코딩테스트 - 면접



## 코딩테스트 유용한 사이트들

BAEKJOON  
ONLINE JUDGE

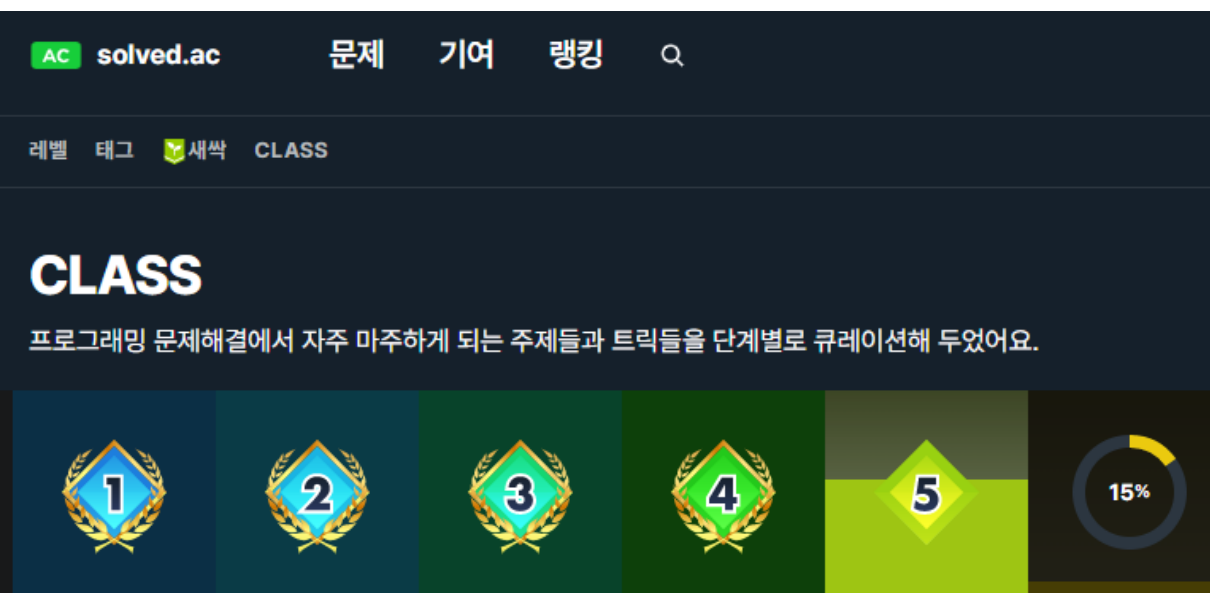


SOLVED. **AC**

 **programmers**

## How to Study

1. Solved.ac 기준 클래스 4~5수준까지 풀이
2. Programmers 고득점 Kit + 카카오 기출문제 풀이
3. 삼성전자 기출 문제 풀이 (구현 연습에 매우 좋음)



AC solved.ac 문제 기여 랭킹

레벨 태그 새싹 CLASS

## CLASS

프로그래밍 문제해결에서 자주 마주하게 되는 주제들과 트릭들을 단계별로 큐레이션해 두었어요.

1 2 3 4 5 15%

## 코딩테스트 고득점 Kit

코딩테스트에는 어떤 알고리즘/자료구조가 출제될까요?

사람들은 어떤 문제를 어려워할까요? 국내에서 코딩테스트를 가장 많이 운영해온 프로그래머스 팀이 코딩테스트 결과를 분석해서 만든 고득점 Kit. 코딩테스트에 자주 나오는 유형, 사람들이 많이 틀리는 유형을 간추렸습니다.



## 해시

Key-value 쌍으로 데이터를 빠르게 찾아보세요.

출제 빈도 높음 ↗ 평균 점수 보통 → 문제 세트 4 / 5

## 스택/큐

LIFO, FIFO, push & pop! 스택과 큐를 이용해서 문제를 풀어보세요.

출제 빈도 보통 → 평균 점수 높음 ↗ 문제 세트 6 / 6

## 힙(Heap)

힙은 특정한 규칙을 가지는 트리로, 힙을 이용해서 우선순위 큐를 구현할 수 있습니다.

출제 빈도 보통 → 평균 점수 높음 ↗ 문제 세트 3 / 3

## 정렬

정렬을 이용해서 문제를 효율적으로 풀어보세요.

출제 빈도 높음 ↗ 평균 점수 높음 ↗ 문제 세트 3 / 3

## 완전탐색

무식해 보여도 사실은 최고의 방법일 때가 있지요.

출제 빈도 높음 ↗ 평균 점수 낮음 ↘ 문제 세트 3 / 7

## 탐욕법(Greedy)

부분적인 최적해가 전체적인 최적해가 되는 방법!

출제 빈도 낮음 ↘ 평균 점수 낮음 ↘ 문제 세트 5 / 6

## 동적계획법(Dynamic Programming)

불필요한 계산을 줄이고, 효율적으로 최적해를 찾아야만 풀리는 문제들입니다.

출제 빈도 높음 ↗ 평균 점수 낮음 ↘ 문제 세트 4 / 7

## 깊이/너비 우선 탐색(DFS/BFS)

깊이/너비 우선 탐색을 사용해 원하는 답을 찾아보세요.

출제 빈도 높음 ↗ 평균 점수 낮음 ↘ 문제 세트 4 / 7

## 이분탐색

이분탐색 기법을 이용해 효율적으로 값을 찾아보세요.

출제 빈도 낮음 ↘ 평균 점수 낮음 ↘ 문제 세트 2 / 2



## 충분하냐? → 충분합니다! BUT 다다익선



함께 가는 친구, 롯데

### 코딩테스트·조직적합진단 결과 안내

박동학님은 2021년 3월 롯데정보통신 Software / Infra / BI 신입사원 채용 코딩테스트·조직적합진단에 **합격**하셨습니다.

### 네이버 신입 공채 : 기술직군

안녕하세요, NAVER입니다.

'2021 네이버 신입 공채 : 기술 직군'의 1차 면접 대상자로 선정되신 것을 다시 한번 축하드리며, 온라인 화상면접으로 진행될 1차 면접 진행 방식 및 인성검사 응시 사항을 아래와 같이 안내드리오니 확인하시고 요청 사항에 대해서는 기한 내 꼭 완료 부탁드립니다.

II

### kakao 인재영입

박동학님, 안녕하세요. 카카오 영입 담당자입니다.

2022 카카오 신입 개발자 블라인드 채용 1차 온라인 코딩테스트에 합격하셨습니다!

### kakao 인재영입

박동학님, 안녕하세요. 카카오 영입담당자입니다.

2022 카카오 신입개발자 블라인드 채용 2차 온라인 코딩테스트에 합격하셨습니다. 축하드립니다!

### 삼성전자

### 2021년 하반기 3급 신입사원 채용 공고

박동학님(donghark03@naver.com)은 SW역량테스트에 합격하셨습니다.



**!!!! 본인이 지원하는 직무에 대해서 프로젝트 수행해보기 !!!!**

**(+ 기술면접 준비, CS공부 → 다음시간에 설명)**

**→ 키워준다는 회사도 면접에서 직무에 관련된 질문을 물어봅니다.**

**→ 직접 프로젝트를 안해본 사람은 대답하기 힘듭니다.**

**ex) 라인AD플랫폼 개발**

**( 직무 요구사항 : 튼튼한 배경지식(운영체제, 네트워크 ..)**

**But 실제 면접 : 백엔드 안해봤는데 왜 지원했나요?**

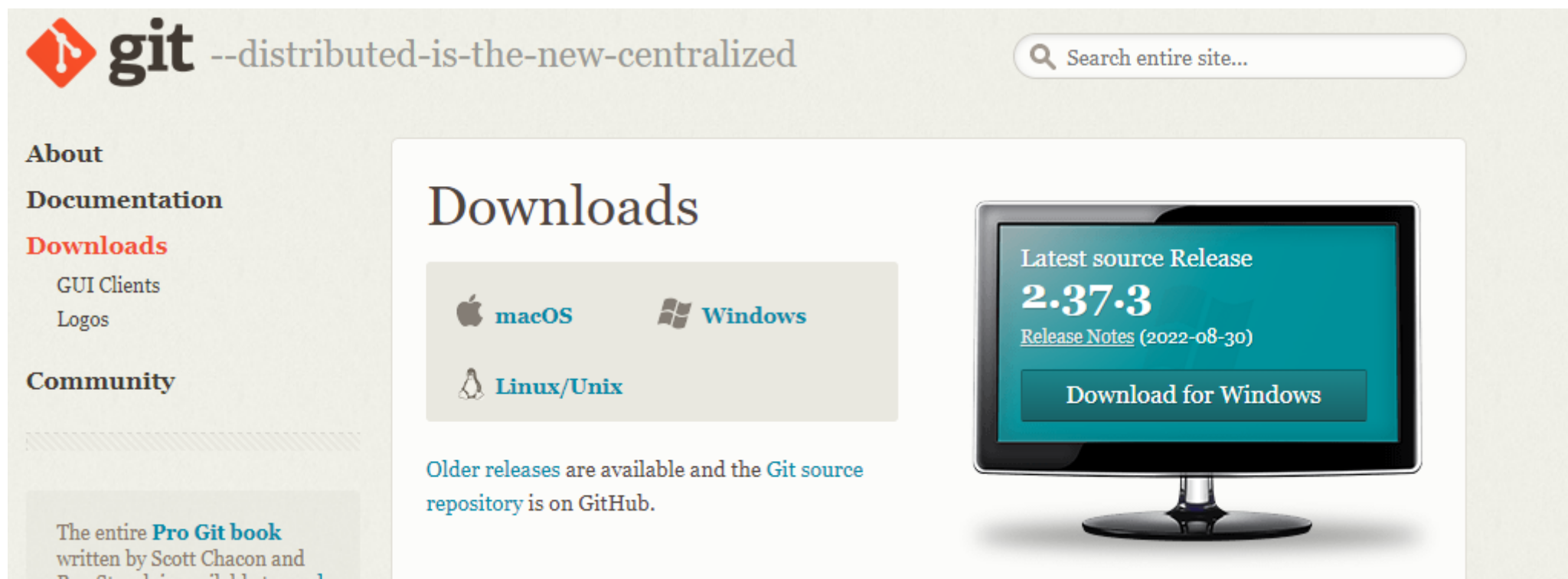
**→ JD를 작성하는 사람과 면접관은 다르다. 면접에서는 당연히 이 사람이 바로 지원한 직무에 대해서 알고있다고 생각함**

1. 최대한 많은 기업 작성해보기 (지원은 기간이 있기 때문에 신경쓰기)
2. 꾸준히 코딩테스트 + CS(기술면접) 준비하기
3. 남는 시간을 활용해서 지원직무관련 간단한 프로젝트라도 수행하기
4. GitHub이나 블로그 처럼 부가적으로 보여지는 부분 준비하기
5. 합격까지 무한 반복

스터디에서는 아래와 같은 부분을 챙겨가셨으면 합니다.

- 지원서 작성 후 피드백 + 기업 정보 탐색
- 코딩테스트 + CS(기술면접 준비)
- Github 활용법
- 같이 준비하는 사람들과 정보교환

## Git 다운로드



The screenshot shows the Git website's 'Downloads' page. At the top left is the Git logo (a red diamond with a white branching diagram) followed by the text 'git --distributed-is-the-new-centralized'. To the right is a search bar with the placeholder text 'Search entire site...'. On the left sidebar, there are links for 'About', 'Documentation', 'Downloads' (highlighted in red), 'GUI Clients', 'Logos', and 'Community'. Below these is a section for the 'Pro Git book' by Scott Chacon and Ben Straub. The main content area is titled 'Downloads' and features three operating system options: 'macOS' (with an Apple logo), 'Windows' (with a Windows logo), and 'Linux/Unix' (with a Tux penguin logo). Below these options, it states 'Older releases are available and the Git source repository is on GitHub.' On the right, there is a large monitor graphic displaying the 'Latest source Release' as '2.37.3' with a 'Download for Windows' button. Below the monitor, it says 'Release Notes (2022-08-30)'.



**git** --distributed-is-the-new-centralized


Search entire site...

**About**  
**Documentation**  
**Downloads**  
GUI Clients  
Logos  
**Community**

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to read

## Downloads


 **macOS**  **Windows**

 **Linux/Unix**

Older releases are available and the [Git source repository](#) is on GitHub.

Latest source Release  
**2.37.3**  
[Release Notes](#) (2022-08-30)  
[Download for Windows](#)


## Fork 하기

 Search or jump to... [Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

[Dev-StudyGroup / Algorithm](#) Public [Edit Pins](#) [Watch 0](#) [Fork 11](#) [Starred 3](#)

[Code](#) [Issues 16](#) [Pull requests 6](#) [Discussions](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

[main](#) [12 branches](#) [0 tags](#) [Go to file](#) [Add file](#) [Code](#)

 DonghakPark Update README.md 1e5273e 10 hours ago 29 commits

Week_0	Create 1152.py	8 days ago
Week_1	Update README.md	9 days ago
LICENSE	Initial commit	13 days ago
README.md	Update README.md	10 hours ago

[README.md](#)

### Algorithm

#### GROUND RULE

- 상호 리뷰를 원칙으로 합니다. ( 멘토 이외에도 서로 리뷰하면서 배울 수 있길 바랍니다.)
- BASE Problem은 모두가 풀기를 원합니다.
- 최소 2명이상의 리뷰를 받아서 Merge 하도록 합니다.

### About

This Repo target for Algorithm

- [Readme](#)
- [MIT license](#)
- [3 stars](#)
- [0 watching](#)
- [11 forks](#)

### Releases

No releases published  
[Create a new release](#)

### Packages

No packages published  
[Publish your first package](#)

## 본인 계정 밑에 Fork해주세요. Fork 후에 본인 repo clone하기

### Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Owner \*



DonghakPark ▾

Repository name \*

Algorithm



By default, forks are named the same as their parent repository. You can customize the name to distinguish it further.

Description (optional)

This Repo target for Algorithm

☒ Copy the `main` branch only

Contribute back to Dev-StudyGroup/Algorithm by adding your own branch. [Learn more.](#)

You are creating a fork in your personal account.

Create fork

## Remote upstream 설정 (원본 repo)

```
PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (test1)
$ git remote -v
origin  https://github.com/Dev-StudyGroup/Algorithm.git (fetch)
origin  https://github.com/Dev-StudyGroup/Algorithm.git (push)

PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (test1)
$ git remote add upstream https://github.com/Dev-StudyGroup/Algorithm.git~

PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (test1)
$ git remote -v
origin  https://github.com/Dev-StudyGroup/Algorithm.git (fetch)
origin  https://github.com/Dev-StudyGroup/Algorithm.git (push)
upstream      https://github.com/Dev-StudyGroup/Algorithm.git~ (fetch)
upstream      https://github.com/Dev-StudyGroup/Algorithm.git~ (push)
```

## 작업 할 branch 생성

```
PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git branch
* main

PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git branch test1

PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git branch
* main
  test1
git
PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git checkout test1
Switched to branch 'test1'
```

## 파일 작성 후 git status 명령어로 변경사항 보기

```
PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git status
On branch main
Your branch is behind 'origin/main' by 1 commit, and can be fast-forwarded.
(use "git pull" to update your local branch)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    [boj]1008.py

nothing added to commit but untracked files present (use "git add" to track)
```

## Git add로 변경파일 추적

```
PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git add \[boj\]1008.py

PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git status
On branch main
Your branch is behind 'origin/main' by 1 commit, and can be fast-forwarded.
(use "git pull" to update your local branch)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   [boj]1008.py
```



## 커밋 메시지와 함께 커밋하기

```
PARK@DESKTOP-35E8N21 MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (main)
$ git commit -s -m "[Add] solve boj 1008 problem"
>
> solve 1008 problem
> "
[main 8ce25d1] [Add] solve boj 1008 problem
1 file changed, 7 insertions(+)
create mode 100644 Week_0/DonghakPark/[boj]1008.py
```

## Git add로 변경파일 추적

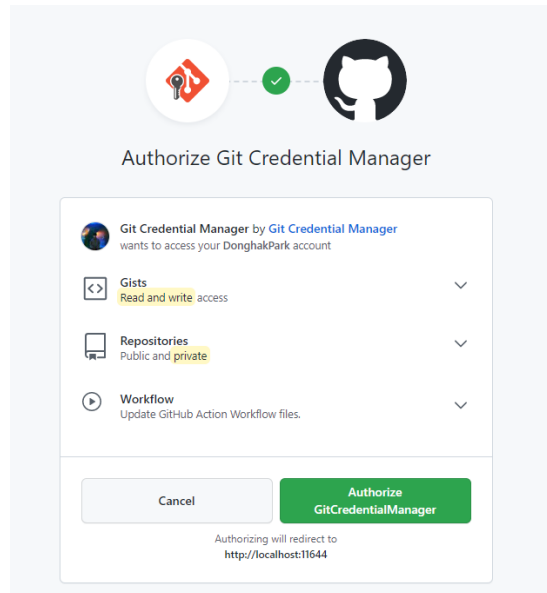
```
PARK@DESKTOP-35E8N21 MINGW64 /d/DongHak Drive/Github/Algorithm/Week_0/DonghakPark (test1)
$ git log --oneline
8ce25d1 (HEAD -> test1, main) [Add] solve boj 1008 problem
4197262 Update README.md
c9d0e78 Merge pull request #3 from Dev-StudyGroup/SAMPLE-Branch
d262ea0 Merge branch 'main' into SAMPLE-Branch
b96b7b4 Resolve Conflict
22bbf83 change dir name
7c6e2f0 Merge pull request #2 from Dev-StudyGroup/SAMPLE-Branch
73506ab Update README.md
```

## 본인 branch로 push

```
PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/
$ git config --global push.default current

PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/
$ git push
```

## Git 로그인 or Token 설정등으로 git 연동



## PR 날리기

test1 had recent pushes 2 minutes ago

Compare & pull request

### [Add] solve boj 1008 problem #34

Open DonghakPark wants to merge 1 commit into `main` from `test1`

Conversation 0 Commits 1 Checks 0 Files changed 1

DonghakPark commented now

solve 1008 problem

Signed-off-by: donghakpark [donghark03@naver.com](mailto:donghark03@naver.com)

[Add] solve boj 1008 problem ...

Add more commits by pushing to the `test1` branch on `Dev-StudyGroup/Algorithm`.

**Review required**  
At least 1 approving review is required by reviewers with write access. [Learn more](#).

**No unresolved conversations**  
There aren't yet any conversations on this pull request.

**Merging is blocked**  
Merging can be performed automatically with 1 approving review.

☐ Merge without waiting for requirements to be met (bypass branch protections)

Merge pull request You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

### own repo PR test #35

Open DonghakPark wants to merge 3 commits into `Dev-StudyGroup:main` from `DonghakPark:test1`

Conversation 0 Commits 3 Checks 0 Files changed 1

DonghakPark commented 41 seconds ago

own repo PR test

DonghakPark and others added 3 commits 24 minutes ago

- [Add] solve boj 1008 problem ... 8ce25d:
- [ADD] From own-repo ... 14d744:
- Merge branch 'Dev-StudyGroup:test1' into test1 Verified f9488f:

Add more commits by pushing to the `test1` branch on `DonghakPark/Algorithm`.

**Review required**  
At least 1 approving review is required by reviewers with write access. [Learn more](#).

**No unresolved conversations**  
There aren't yet any conversations on this pull request. [View](#)

**Merging is blocked**  
Merging can be performed automatically with 1 approving review.

## Fork한 repo 최신화 하기 (main branch에서)

```
PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm (main)
$ git fetch upstream
remote: Enumerating objects: 221, done.
remote: Counting objects: 100% (217/217), done.
remote: Compressing objects: 100% (164/164), done.
remote: Total 209 (delta 27), reused 196 (delta 19), pack-reused 0
Receiving objects: 100% (209/209), 30.25 KiB | 209.00 KiB/s, done.
Resolving deltas: 100% (27/27), completed with 3 local objects.
From https://github.com/Dev-StudyGroup/Algorithm
* [new branch]      30-week-0-이 정 현 -> upstream/30-week-0-이 정 현
* [new branch]      Juwon-week0       -> upstream/Juwon-week0
* [new branch]      chchaeun_week0    -> upstream/chchaeun_week0
* [new branch]      eunyu             -> upstream/eunyu
* [new branch]      eunyu_week0       -> upstream/eunyu_week0
* [new branch]      jeonghyunLee99_week0 -> upstream/jeonghyunLee99_week0
* [new branch]      jeongmin_week0    -> upstream/jeongmin_week0
* [new branch]      junseok_week0     -> upstream/junseok_week0
* [new branch]      main              -> upstream/main
* [new branch]      sejun_week0       -> upstream/sejun_week0
* [new branch]      test              -> upstream/test
* [new branch]      test1             -> upstream/test1
* [new branch]      yobinmok_week0    -> upstream/yobinmok_week0

PARK@DESKTOP-35E8N2J MINGW64 /d/DongHak Drive/Github/Algorithm (main)
$ git merge upstream/main
Already up to date.
```

**QnA**