

CS Study OT

2025.08.07

Computer Science

SSAFY

송현우

Index

1	개요	03
2	설문결과	06
3	진행방식	10
4	커리큘럼	13
5	QnA	18

개요

스터디 방향성

개요

- 발표를 통해 **나만의 무기**가 될 수 있는 경험을 만들어 나가기
- 정답을 아는 사람보다 **좋은 질문**을 하는 사람을 존중하고, 함께 답을 찾기

- **[온라인 세션]** 효율적인 정보 공유와 원활한 발표 진행을 위해 활용
- **[오프라인 세션]** 양방향 소통과 강제성을 통해 스터디의 밀도를 높이기

- 성장하는 스터디

CS를 대하는 자세

개요

기술면접

- 취업 과정에서 생각만큼 **CS**가 중요하지 않을 수도 있다.
- 하지만, 기술 면접의 주도권을 가져올 기회가 될 수도 있다.

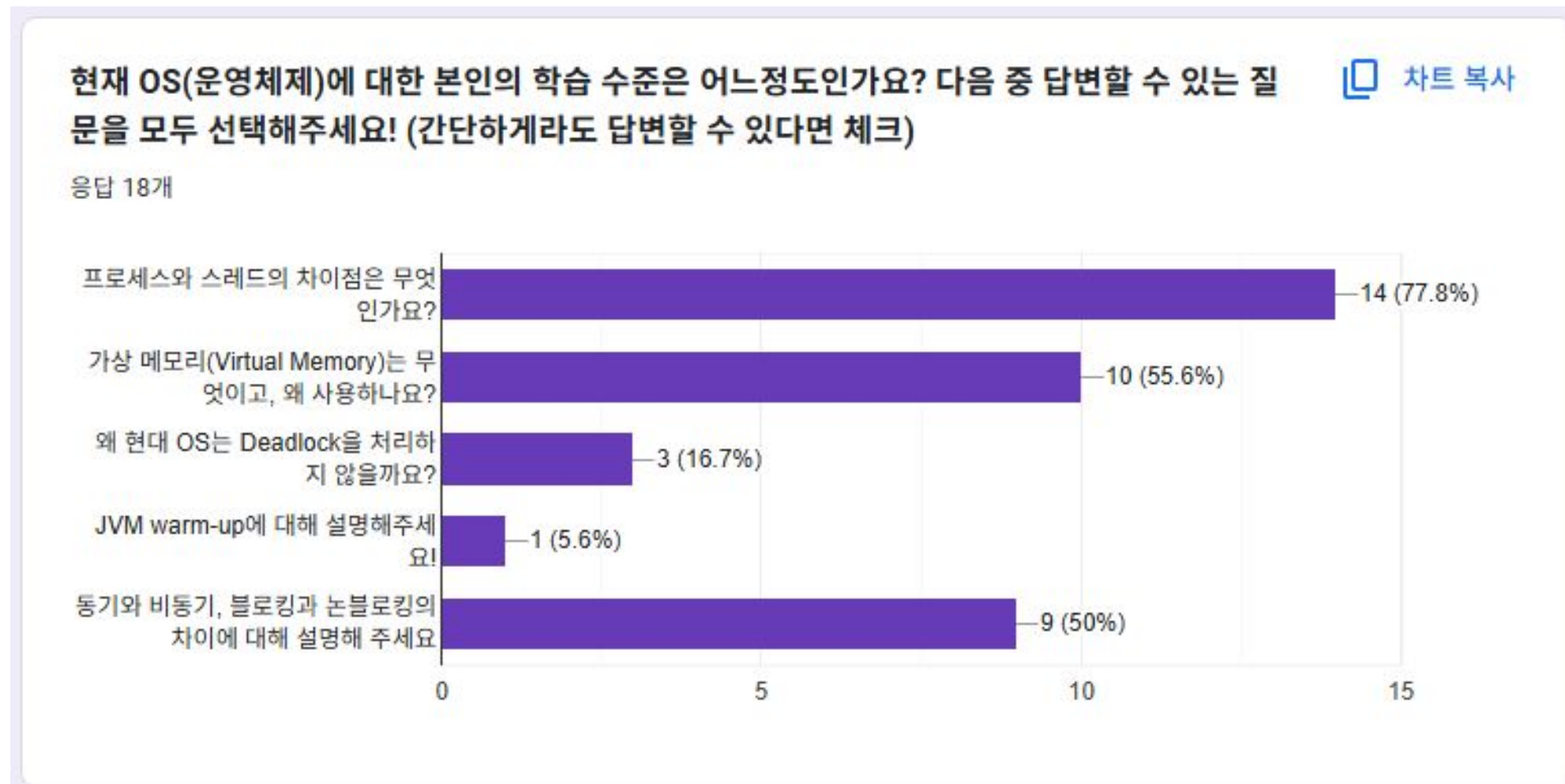
문제해결력

- ‘깊이 있게 파고들어 본 경험’ 그 자체가 중요하다.
- 추상화의 끝은 결국 **CS**다.

설문 결과

Operating System

설문결과



JVM warm-up

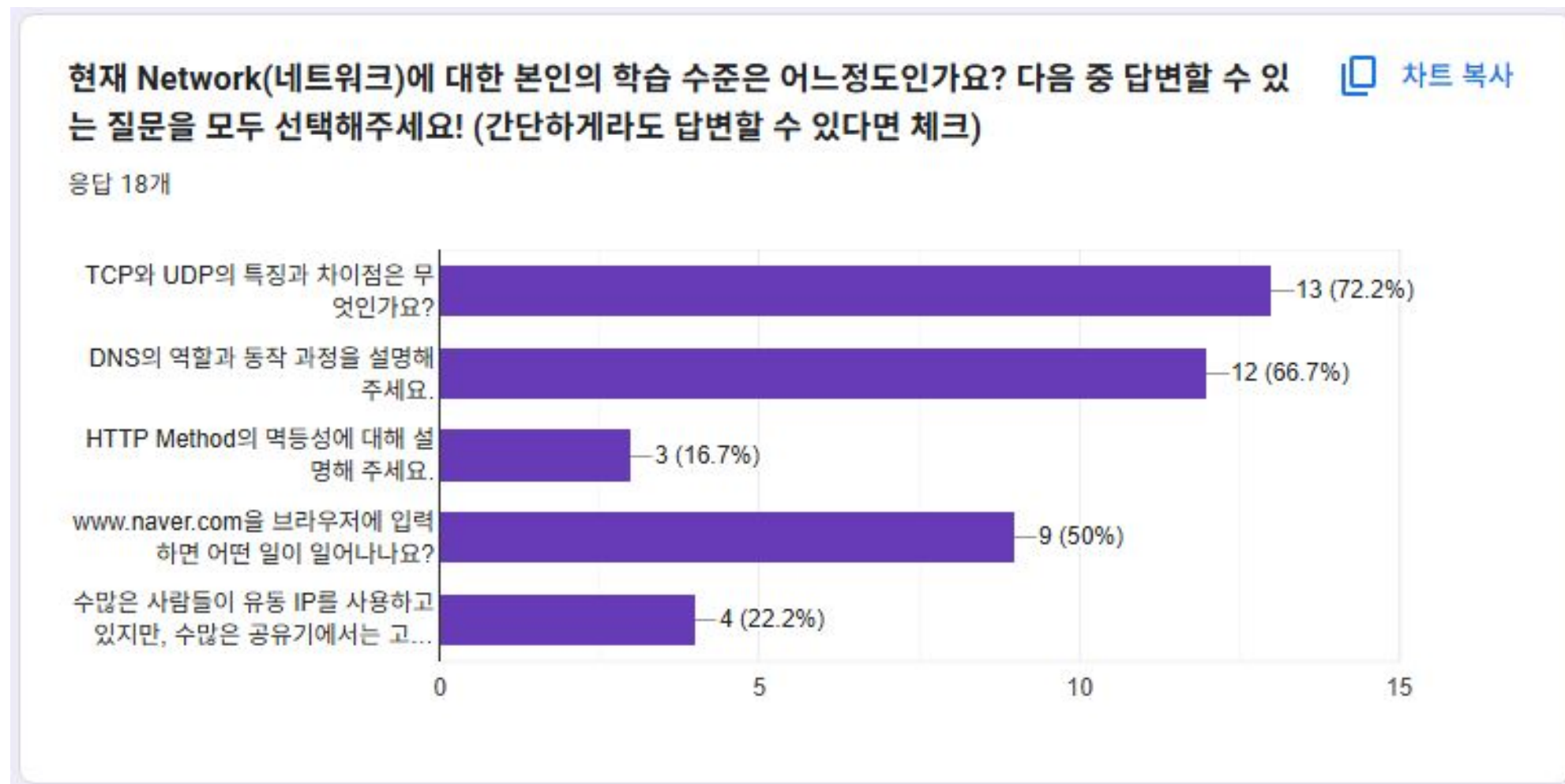
- 클래스 로더, lazy loading
- JIT compiler 관찰하기
- warm-up 활용 : ApplicationRunner, 카나리 배포, k8s + HealthIndicator

sync, async, block, non-block

- 독립적인 개념임을 이해하기
- 실제로 구현해보기
- I/O Multiplexing, AIO

Network

설문결과



브라우저에 주소를 입력하면

- DNS 질의와 http request를 나눠서 설명
- DNS : 브라우저 캐시(memory, disk), OS 캐시, 라우터 캐시, DHCP, DNS 리졸버(ISP) 캐시, 재귀적 질의
- request : 로컬 캐시, Conditional GET, CDN...

HTTP Method의 멍등성

- 멍등성을 유지하는 설계
- REST API란?
- Cacheable

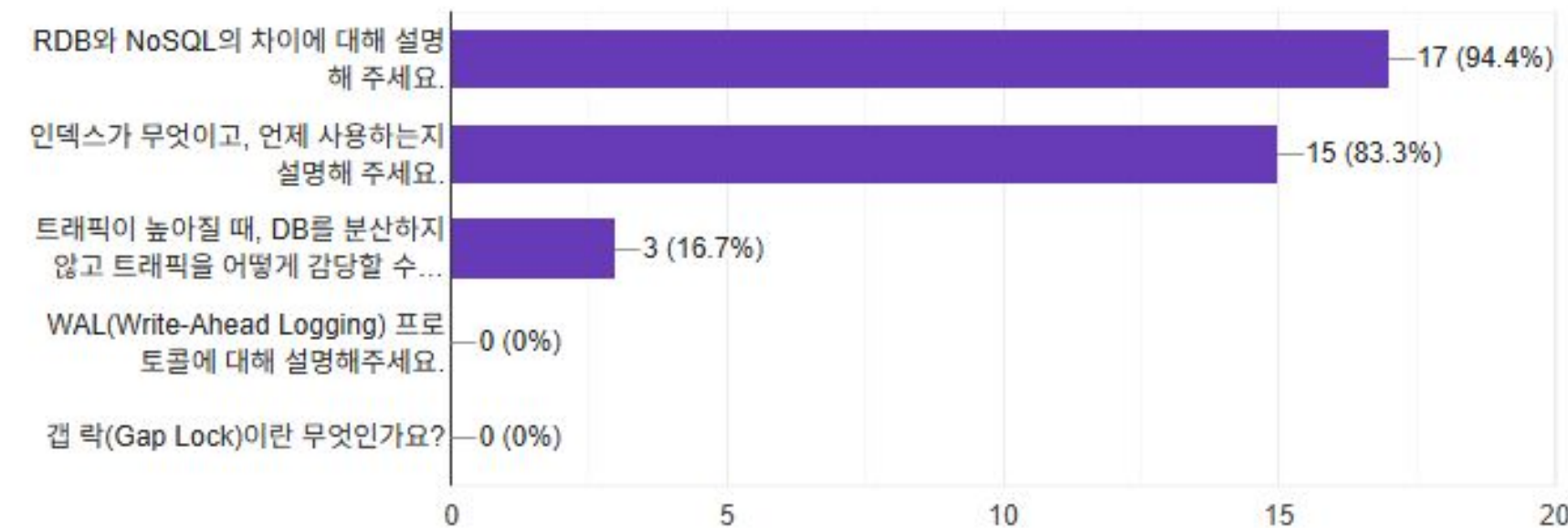
Database

설문결과

현재 Database(데이터베이스)에 대한 본인의 학습 수준은 어느정도인가요? 다음 중 답변할 수 있는 질문을 모두 선택해주세요! (간단하게라도 답변할 수 있다면 체크)

 차트 복사

응답 18개



ACID

- A : MVCC, pudge thread
- C : unique, normalization
- I : gap lock
- D : WAL protocol
- 분산 환경 : Raft algorithm, 2PC, Paxos...

분산 환경을 논하기 이전에

- 캐싱, 쿼리 최적화(프로파일링), 커넥션 풀
- innodb redo log, buffer pool 최적화

진행 방식

온라인 세션

진행방식

- 질문과 관련된 주제에 대한 발표
 - 모두가 의미있는 경험을 가져가기 위한 세션
 - 발표는 주차당 4명 선정
 - 기본적으로 랜덤이나, 원하는 주제가 있을 경우 선점 가능 -> 사유 필수
 - 다양한 관점을 얻기 위해 발표 주제가 겹칠 수 있음
 - 발표 시간은 15분 이내
 - 반드시 모든 꼬리질문에 답변할 필요 없음
 - 하나의 주제에 대해 깊게 파고들면 좋음
 - 실습 지향
-

오프라인 세션

진행방식

- 양방향 소통을 위한 세션
 - 질문에 대해 각자 간단히 답변을 정리해올 것
 - 모든 질문에 완벽하게 답변 X, 모르는 것을 정리해오기 O
 - 정리해온 답변을 공유하며 토론
 - 최종적으로 그룹 내에서 정리한 답변을 깃허브에 업로드 (템플릿 제공)
 - 원활한 소통을 위해 그룹 분배
 - 그룹은 각 커리큘럼(OS, Network, Database)마다 재분배
 - 각 커리큘럼 별 스터디 참여&탈퇴 가능
 - 오프라인 모임은 각 그룹별 그룹장을 선정해 관리
-

커리큘럼

커리큘럼

커리큘럼

	Description	기간
OS	23개의 메인 질문 5 / 4 / 5 / 4 / 5	5주 (8/14 ~ 9/17)
Network	21개의 메인 질문 6 / 6 / 4 / 5	4주 (9/18~10/1, 10/16~10/29)
Database	16개의 메인 질문 4 / 4 / 4 / 4	4주 (10/30~11/26)

Time Table(OS)

커리큘럼

날 짜	내 용
8/14 ~ 8/20	시스템 콜, 인터럽트, 프로세스, 스케줄러
8/21 ~ 8/27	컨텍스트 스위칭, 스케줄링 알고리즘, 뮤텍스&세마포어, Deadlock
8/28 ~ 9/3	컴파일, 인터프리터, IPC, Thread Safe, Thread Pool, 캐시
9/4 ~ 9/10	Thrashing, 가상 메모리, 세그멘테이션&페이징, TLB
9/11 ~ 9/17	동기화, 페이지 교체 알고리즘, 파일 시스템, 동기&비동기, blocking&non-blocking

추천 도서

[운영체제 | Abraham Silberschatz - 교보문고](#)

[운영체제 | Remzi H. Arpaci-Dusseau - 교보문고](#)

Time Table(Network)

커리큘럼

날 짜	내 용
9/18 ~ 9/24	쿠키, 세션, HTTP, 웹소켓
9/25 ~ 10/1	TCP, UDP, DHCP, IP, OSI 7 layer, 3/4 - Way Handshake
10/2~10/15	연휴 !! + 중간 점검
10/16 ~ 10/22	DNS, Conditional GET, Cache, Cors, Stateless
10/23 ~10/29	Routing, 로드밸런서, Subnet mask, multiplexing, XSS

Time Table(Database)

커리큘럼

날 짜	내 용
10/30 ~ 11/5	Key, RDB, NoSQL, Transaction
11/6 ~ 11/12	Index, Clustering, Replication, 정규화, View
11/13 ~ 11/19	Join, B+Tree, Lock, DB 최적화
11/20 ~ 11/26	Schema, 커넥션 풀, 쿼리 옵티마이저, SQL Injection

QnA