Phuong Ngo
USC ID: 8312646237

1. Consider a hard disk with 2 platters. Each platter has two recording surfaces (top and bottom). Each surface has 100,000 tracks. Each track holds 2000 sectors of 4K bytes each. The disk rotates at 10,000 RPM. Assume the disk has a Read/Write bandwidth of 100MB/sec. Note that a block is the unit of data transfer between disk and main memory. Each block may contain a number of sectors. Here we assume there is only one sector per block. Note also that traditionally the sector size is 512 bytes, although 4KB is now more commonly used. In addition, we assume that the time it takes the head to move n tracks is 1 + 0.0002n milliseconds.

a. What is the capacity of the disk?
2 platters * 2 recording surfaces * 100,000 tracks per surface * 2000 sectors per track * 4K per sector = 3.2 TB

b. What is the maximum and average seek time? (assume that the average seek time is approximately 1/3 of the maximum seek time.)
Max seek time = 1 + 0.0002 * 100,000 ms = 21ms
Avg seek time = 1/3 * 21ms = 7ms

c. What is the maximum and average rotational latency?
Max. rotational latency = 60,000 ms / 10,000 rotations = 6ms
Avg. rotational latency = 1⁄2 * 6ms = 3ms.

d. What if all the 32 sectors to be read are located next to each other on the same track?
Then only one average seek and rotational latency incurred:
7ms + 3ms + 32 * 4K/100M * 1000ms = 10ms + 32 * .04ms = 11.28ms

e. How long does it take to read 8MB, i.e., perform 2000 reads of size 4KB each from random locations on the drive? You may assume that each 4KB occupies a sector.
2000* (7ms + 3ms + 4K/100M * 1000 ms) = 2000* (10ms + .04ms) = 2000* 10.04 ms = 20080 ms
f. How long does it take to read 8MB, i.e., perform 2 reads of size 4MB each from random locations on the drive? Now, assume that each 4MB occupies a group of sequential sectors on the same track which can be read without extra seek or rotational latency.
2 * (7ms + 3ms + 4M/100M * 1000 ms) = 2 * (10ms + 40 ms) = 32 * 50 ms = 100 ms

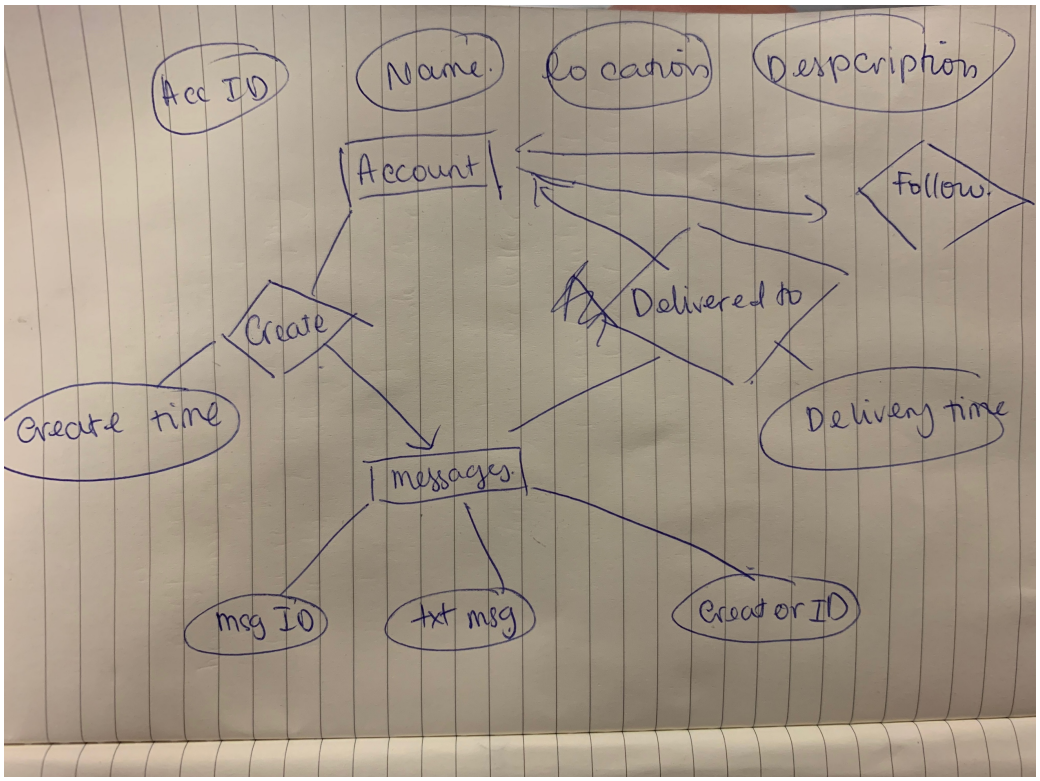
2. Suppose you are given the following requirements for a simple database for Twitter messages:

- there are many Twitter accounts (account entity) and many messages (message entity)
- each account has an account ID, name, location, account description, and can have many received messages
- each message has a message ID, creator account ID, creation time, textual message

- one message belongs to only one original creator account
- one message can be delivered to multiple accounts at different times
- one message can belong to multiple accounts (followers)

a) Construct a clean and concise ER diagram for this database using the notation as in the lecture note. Explain your assumptions in your ER diagram.

One message might have two time marks for creation time and delivered time.
An account can follow and/or be followed by another account(s).



b) Create tables to implement your database design. And insert several actual values per table.

a. Account table

Acc_id	Name	Location	Description	Txt_msg
001	akg	LA		love
002	abb	DC		hate
003	abg	MN		love

b. Message table

Msg_id	Txt_msg	acc_id
0001	love	022
0002	hate	033

0003	like	024

c. Delivery table

Msg_id	Deliver_to(acc_id)	Deliver_time
0001	001	2pm
0001	003	2pm

d. Create table

Acc_id	Create_time	Msg_id
022	1pm	0001
033	2pm	0002

We can essentially merge d and b with each other.

e. Following/follower

Name	Follower Name
akg	abg
abb	abg

c) Explain how to search the name(s) of account holder(s) who is (are) the receiver(s) of a specific message in your database. Plain step-by-step description is fine.

SELECT *
FROM Account
WHERE Txt_msg = ‘love’