1.

a) the relation is in 2nd normal form, since for each relation the primary key(s) have no functional depencencies to their children. Book\_title doesn't imply the publisher or genre, book\_type does not imply list\_price, and author\_name does imply author\_affil.

b)

2.

Doctor#, Patient#, Date

Patient#, Diagosis, Treat\_code

-> No this is not in 2nd normal form, since we have been told that the treat\_code and charge is dependent on the diagosis. 2nd normal form explicitly states that that all non-key columns are dependent on the tables primary key, which is not true regarding charge which is dependend on diagonsis.

3.

ACID properties include Atomicity, Consistency, Isolation, and Durability. Atomicity says that a operation (transaction) must occur completely, or not at all, meaning we cannot have any partial transictions to prevent data corruption. Consistency states that any crud operation is consistient with the rules of the database, that is, a crud operation cannot violate the database conditions. Isolation is another form of mutual exclusion, that is, transactions cannot interfere with eachother otherwise data may get corrupted. Durability means that transactions will persist in the database, that is, a transaction will remain committed (not partially committed) even if a database fails. If a transaction commit fails mid commit, the commit will not fully go through to preserve the data.

4.

5.

6.

(512+128)\*20 = 12800bytes -> Total Capacity

(512)\*20 = 10240bytes -> Useful Capacity

Cylinders = 400 (same as tracks)

(512+128)\*20\*30 = 384000bytes -> Total Capacity of a cylinder

(512)\*20\*30 = 307200bytes -> Useful Capacity of a cylinder

(512+128)\*20\*30\*400 = 153600000bytes -> Total Capacity of a disk pack

(512)\*20\*30\*400 = 122880000bytes -> Useful Capacity of a disk pack

transfer rate = track capacity / rpm\*60\*1000 -> 12800 / 2400\*60\*1000 = 88.9 bytes a second

block transfer time = blocks/tr = 512/88.9 = 5.76 millsec

rot delay = 1/2(1/2400)\*60\*1000 = 12.5 millsec

average seek time is blocktrans + rot del + seek time = 5.76 + 12.5 + 30 = 48.26 millisec

20 random blocks = blocks\*(seek + rot + block trans) = (48.26 millisec) \* 20 = 965.2 millisec

20 consecutive w/ bufffer = (seek + rot + blocks\*block trans) = 157.7 millisec

7.

record = name + ssn + deptcode + addr + phone + bdate + sex + jobcode + sal = 30 + 9 + 9 + 40 + 9 + 8 + 1 + 4 + 4 + 1 = 115 bytes per record

blk fctr = average record per block = 115 / 512 = 4.45, or about 4 records per block

numofblks = 30000/4 approx 7500

index blk = index block size + key size = 6 + 9 = 15 bytes

index blk fctr = blk / index blk = 512/15 = 34

7500 entries in blk, 7500 first level keys

num of first-index-blks = 7500/34 = 221 blocks

(MORE HERE)

8.

student:

sname, snum, sssn, scaddr, scphone, apaddr, spphone, bdate, sex, class, majordeptdcode, minordeptcode, prog

dept:

deptname, deptcode, deptoffice, deptphhone, deptcollege

course:

cname, cdesc, cnum, cdept,

section:

instructorname, semester, year, seccourse, sectionnum

grade:

ssn, grade

majordeptcode, minordeptcode, cdept -> deptcode

seccourse -> course

ssn -> sectionnum

functional dependencies:

snum, sssn -> sname, scaddr, scphone, apaddr, spphone, bdate, sex, class, majordeptdcode, minordeptcode, prog

cnum -> cname, cdesc, cdept

sectionnum -> instructorname, semester, year, seccourse : ssn -> grade

3rd normal form:

snum, sssn -> majordeptcode, majordeptcode -> s