CHAPTER 11
DISCUSSIONS 1

Discussion 11-1

Indices speed query processing, but it is usually a bad idea to create indices on every (combination of) attribute. Explain why.

Discussion 11-2

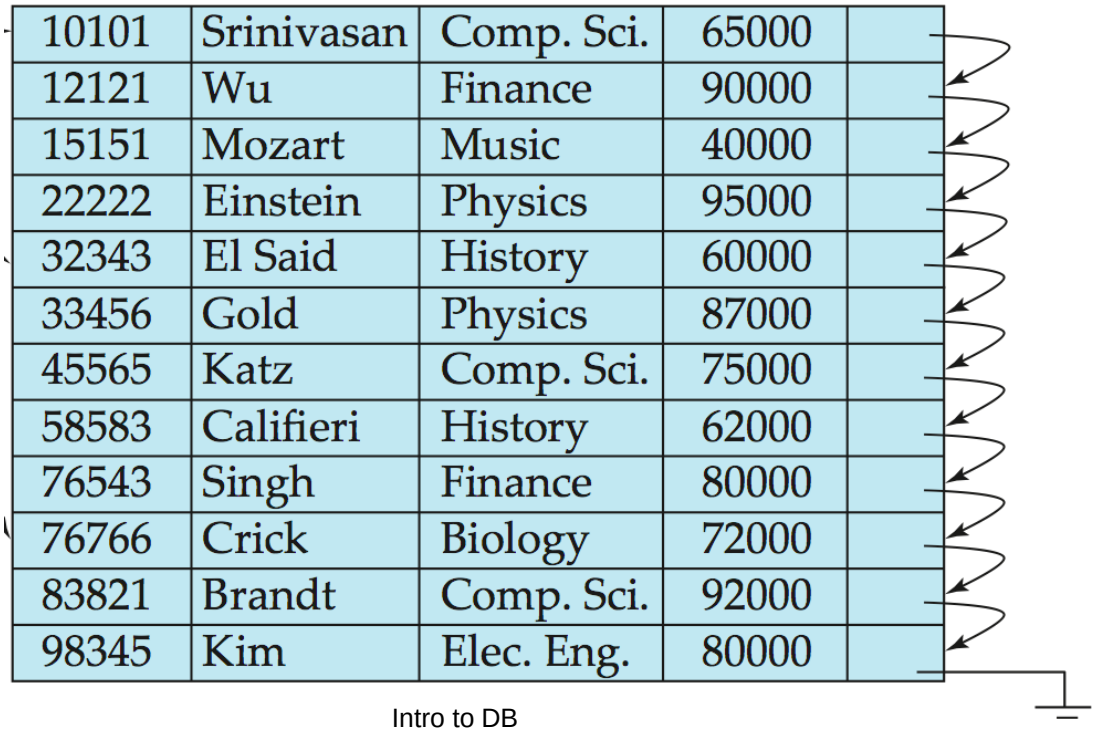
Define the following terms and compare the meaning of '*primary*' in these terms.

1) *primary key*
2) *primary index*
3) *primary residence* of a database
4) *prime attribute* (in 3NF)

Discussion 11-3

Consider the following table *instructors*(*ID, name, dept, salary*).

1) For which attribute can you build a *sparse index*? Build a sparse index with 4 entries.
2) Build a *dense index* for *dept*.



10101	Srinivasan	Comp. Sci.	65000	
12121	Wu	Finance	90000	
15151	Mozart	Music	40000	
22222	Einstein	Physics	95000	
32343	El Said	History	60000	
33456	Gold	Physics	87000	
45565	Katz	Comp. Sci.	75000	
58583	Califieri	History	62000	
76543	Singh	Finance	80000	
76766	Crick	Biology	72000	
83821	Brandt	Comp. Sci.	92000	
98345	Kim	Elec. Eng.	80000	

Discussion 11-4

Explain how you would utilize the B+-tree index shown below to answer the following queries.

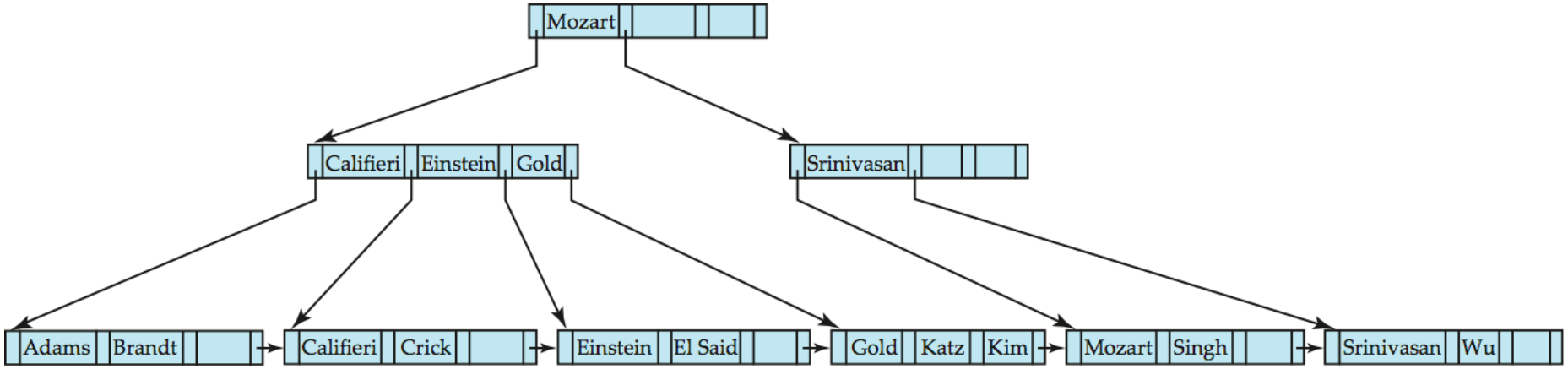
a) **select** * **from** instructors **where** name = ‘Kim’

b) **select** * **from** instructors **where** name **like** ‘K%’

c) **select** * **from** instructors **where** name >= ‘Kim’

d) **select** * **from** instructors **where** name = ‘Lee’

e) **select** * **from** instructors **where** name <> ‘Kim’



Discussion 11-5

Consider a B+-tree index of degree 100 on a candidate key *ID* of a table *student* with 100,000 records.

A. How many block accesses should we expect for an equality search on *ID*?

B. Suppose the size of an index node is 2 blocks (degree 200) instead of 1. What is the expected number of block accesses for an equality search on *ID*?

Discussion 11-6

Consider a (secondary) B+-tree index of degree 100 on the *name* attribute of *student* with 100,000 records.

- How many block accesses should we expect for an equality search on *name*?