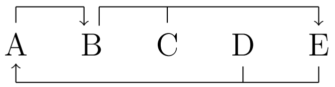
CIS 551: Assignment 5

Steven Walton

November 25, 2019

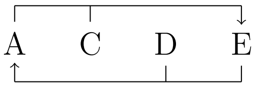
Problem 1: 19.2

Consider a relation R with five attributes $ABCDE$. You are given the following dependencies.



Part 1)

List all keys for R
We can see that $A \rightarrow B$ which means we can rewrite the relationships as $ACDE$. Giving us



With A and C we are given E. Thus we have

A C D

And we no longer have any relationships.
Similarly we can get the other keys. Giving us the answer:
ACD, BCD, CDE

Part 2)

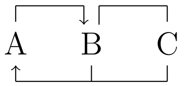
Is R in 3NF?
Yes. B,E,A (the right sides from the FDs) are all part of keys.

Part 3)

Is R in BCNF?
No. None of the left sides of the FDs contain a key and there is no super key.

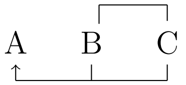
Problem 2: 19.6

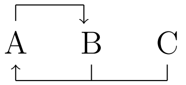
Suppose that we have the following three tuples in a legal instance of a relation scheme S with three attributes ABC (listed in order): $(1, 2, 3), (4, 2, 3),$ and $(5, 3, 3)$.

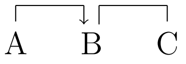


Part 1)

Which of the following dependencies can you infer does *not* hold over scheme S?
(a) $A \rightarrow B$, (b) $BC \rightarrow A$, (c) $B \rightarrow C$
We'll look at the following three pictures







We see that the third diagram is the only legal dependency, which is where the dependency $BC \rightarrow A$ is removed. $\therefore$ (b) does **not** hold.

Part 2)

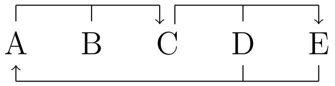
Can you identify any dependencies that hold over S?
We are only given an instance of S and can't determine what relationships hold for **all** instances of S.

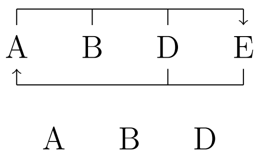
Problem 3

Removed from assignment

Problem 4

Consider relation $R = ABCDE$ with the below relationships. Convert to BCNF.





Following the same pattern we get all possible candidate keys ABD, BDE, and BCD. We notice that BD has to be in all keys because they have no dependencies.

Problem 5: 19.13

Consider a relation R with five attributes $ABCDE$

Part 1)

For each of the following instances of R, state whether it violates (a) the FD $BC \rightarrow D$ and (b) the MVD $BC \rightarrow\rightarrow D$

(a) $\{\}$ No.

(b) $\{(a,2,3,4,5),(2,a,3,5,5)\}$ $BC \rightarrow D$ is violated if $a = 2$

(c) $\{(a,2,3,4,5),(2,a,3,5,5),(a,2,3,4,6)\}$ $BC \rightarrow D$ and $BC \rightarrow\rightarrow D$ are both violated if $a = 2$. We need another tuple that ends in $5,6$.

(d) $\{(a,2,3,4,5),(2,a,3,4,5),(a,2,3,6,5)\}$ $BC \rightarrow D$ is violated because if $a = 2$ then there is a duplicate row.

(e) $\{(a,2,3,4,5),(2,a,3,7,5),(a,2,3,4,6)\}$ Both $BC \rightarrow D$ and $BC \rightarrow\rightarrow D$ are violated. $BC \rightarrow D$ is violated because of the second tuple. $BC \rightarrow\rightarrow D$ is violated because we must have another tuple that pairs with the second tuple.

(f) $\{(a,2,3,4,5),(2,a,3,4,5),(a,2,3,6,5),(a,2,3,6,6)\}$ $BC \rightarrow D$ is violated because there is not a unique value defined by BC. $BC \rightarrow\rightarrow D$ is violated because the entry for the Dth column in the 4^{th} tuple isn't the same as in the 1^{st} tuple.

(g) $\{(a,2,3,4,5),(a,2,3,6,5),(a,2,3,6,6),(a,2,3,4,6)\}$ Just $BC \rightarrow D$ is violated because the third tuple's last entry isn't also a 5.

Part 2)

If each instance for R listed above is legal, what can you say about the DF $A \rightarrow B$? We cannot say anything, because there are not unique relationships.

Problem 6

Let R be a relation, X a set of attributes of R, and A an attribute of R. (Also denote that XA the result of adding A to X) Define the support of X, written as

$\#X$, as the number of distinct tuples in $R|X$ (R restricted to, or projected onto, the attributes of X). Prove that if $X \rightarrow A$, then $\#X = \#XA$.
Proof by contradiction:
Let $X = BC$ such that $BC \rightarrow A$.

A	B	C
1	2	3
4	5	6
7	2	3

We see here that there is a violation in the third row because $\#X = 2$ and $\#XA = 3$. This is not a legal relationship. We note that we only have legal schemas if there is an injective relationship. If we have an injective relationship (onto) then we would have the relationship $\#X \leq \#XA$ (allows for surjection). For strict equality to hold, there needs to be a bijective relationship.

Problem 7

Prove that if R is in 3NF and has only one candidate key, then R is in BCNF.

Problem 8

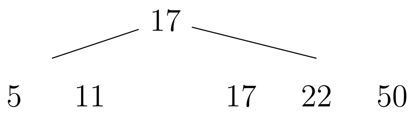
Insert the following values into an initially empty B+ tree with parameter $d = 2$ and values $17, 11, 50, 22, 5, 35, 42, 60, 15, 30, 25, 27, 37, 40, 20$.
If $d = 2$ then we can have a maximal of 4 elements in a leaf.

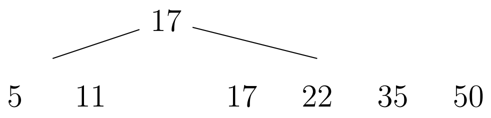
$$11 \quad 17 \quad 22 \quad 50$$

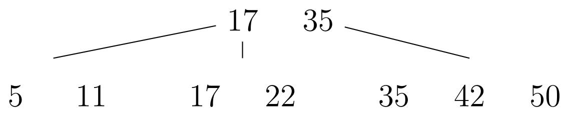
Inserting 5 causes an overflow and promotes the middle number

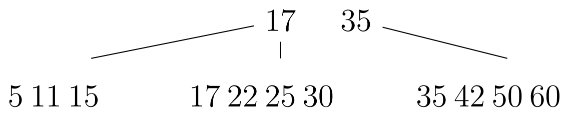
$$5 \quad 11 \quad \cancel{17} \quad 22 \quad 50$$

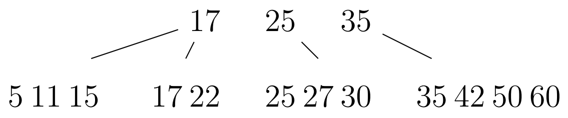
Next part goes smoothly
42 causes an overflow, with 35 getting promoted
Continuing
Adding 27 causes an overflow, with 25 getting promoted
37 causes an overflow with 42 getting promoted, and finishing off

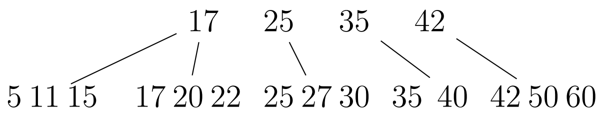

17
5 11
17 22 50


17
5 11
17 22 35 50


17 35
5 11
17 22
35 42 50

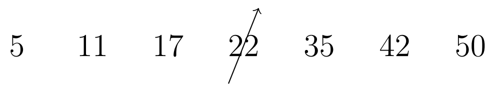

17 35
5 11 15
17 22 25 30
35 42 50 60

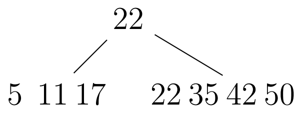

17 25 35
5 11 15
17 22
25 27 30
35 42 50 60


17 25 35 42
5 11 15
17 20 22
25 27 30
35 40
42 50 60

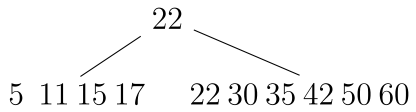
Problem 9

Repeat the previous problem with $d = 3$
Writing the first part

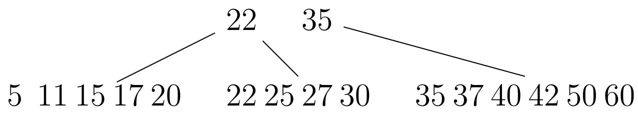




Continuing



25 causes the overflow with 35 being promoted



Problem 10: 10.8 part 1

Assume that you have just built a dense $B+$ tree index using Alternative (2) on a heap file containing 20,000 records. The key field for this $B+$ tree index is a 40-byte string, and it is a candidate key. Pointers (le., record ids and page ids) and (at most) 10-byte values. The size of one disk page is 1000 bytes. The index was built in a bottom-up fashion using the bulk loading algorithm, and the nodes at each level were filled up as much as possible.

Part 1)

How many levels does the resulting tree have?
An index page is at most d keys and $2d + 1$ pointers.

$$\begin{aligned} 2d \cdot 40 + (2d + 1) \cdot 10 &\leq 1000 \\ 80d + 20d + 10 &\leq 1000 \\ 100d &\leq 990 \\ d &\leq 9.9 \end{aligned}$$

This gives a max of 18 keys and 19 pointers per index page. A leaf page is composed of a key and a pointer ($40 + 10 = 50$ bytes). $\therefore$ we have $1000/50 = 20$ entries and $\lceil \log_{19}(\frac{20{,}000}{20}) + 1 \rceil = 4$ levels.