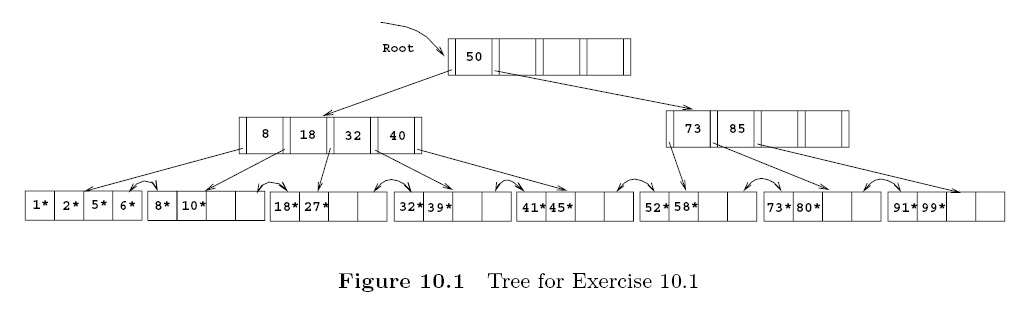
CS1555 Recitation 13

Objective: To practice B+-tree and Concurrency Control

**Part 1: B+Tree**

1. Consider the B+ tree index of order *n* = 5 shown in Figure 10.1.
   1. Show the tree that would result from inserting a data entry with key 9 into this tree.
   2. Show the B+ tree that would result from inserting a data entry with key 3 into the original tree.
   3. Show the B+ tree that would result from deleting the data entry with key 8 from the original tree, assuming that the left sibling is checked for possible redistribution.
   4. Show the B+ tree that would result from deleting the data entry with key 8 from the original tree, assuming that the right sibling is checked for possible redistribution.
   5. Show the B+ tree that would result from starting with the original tree, inserting a data entry with key 46 and then deleting the data entry with key 52.
   6. Show the B+ tree that would result from deleting the data entry with key 91 from the original tree.
   7. Show the B+ tree that would result from starting with the original tree, inserting a data entry with key 59, and then deleting the data entry with key 91.
   8. Show the B+ tree that would result from successively deleting the data entries with keys 32, 39, 41, 45, and 73 from the original tree.

**

**Part 2: Concurrency Control**

1. Consider the following two transactions:

T1: r1(A) ; T2: r2(C);

r1(B); r2(B);

If A=0 then B:= B+1; r2(A);

w1(B); if B>C then {A:= A+1; C:=C+1;}

w2(C)

w2(A)

* For each of the following histories/schedules:

a) Is it a valid history?

b) Use *serializability graphs* to check whether it is serializable or not, and if it is, what is

the equivalent serial history/schedule?

H1: r1(A) r1(B) r2(C) w1(B) r2(B) r2(A)w2(C) w2(A)

H2: r1(A) r1(B) r2(C) r2(B) w1(B) r2(A)w2(C) w2(A)

1. Consider the following history, with lock and unlock statements added for each transaction:

a) Does the history follow 2PL protocol?

b) Did deadlock happen?

|  |  |
| --- | --- |
| T1 | T2 |
| rl1(A)  r1(A) |  |
| rl1(B)  r1(B) |  |
|  | rl2(C)  r2(C) |
|  | rl2(B)  r2(B) |
| wl1(B)  w1(B) |  |
|  | rl2(A)  r2(A) |
|  | wl2(C)  w2(C) |
|  | wl2(A)  w2(A) |
| Commit | Commit |