CS 437/537 (Fall 2016)

Assignment 5

Published: Nov. 11, 2016
Due: Nov. 18, 2016 (11:59pm)

Total: 50 points

Please upload your solutions to classes*v2. To do so, please enter the classes*v2 page of CPSC 437/537, then click the “Assignment” button on your left-hand toolbar, and finally click “Assignment5” to upload your assignment.

If you know you are going to submit your assignment late, please let us know in advance (send an email to cs437ta@cs.yale.edu). Any and all resources may be used as long as you cite them, with the exception of collaborating with other people.

Do not copypasta your solutions from our slides, or the Internet. We do not really care if you do this, but you are not really learning anything.

If you have ANY questions, please do not hesitate to let us know (email, office hours, etc.)
Part 1: Short questions (14 points)

Any reasonable answers are acceptable.

1. What is the difference between B+ tree and B-tree? (2 points)

There are many differences. The most important difference is B+ tree’s leave nodes contain all the data; on the contrary, leave nodes in B-tree do not contain all the data, since some data is located in intermediate nodes.

2. List two differences between programs and transactions? (4 points)

Transaction has atomicity requirement and durability requirement.

3. Suppose we have a DB system. This system creates two backups for each of its data. We now upgrade this DB by increasing one backup, which means the upgraded DB creates three backups for each of data now. For the upgraded DB, the consistency of its transactions becomes harder to maintain or easier to maintain? (2 points) Why? (2 points)

Consistency becomes harder to maintain, because there are more replicas. In the upgraded system, you need to ensure the three replicas have the same value, but in the older version, you only need to focus on two.

4. In the above DB (the upgraded version), the durability of its transactions becomes better or worse? (2 points) Why? (2 points)

More replicas enhance the durability of transactions. Assume each replica server fails independently. More replicas means you have lower probability that all the replicas become unavailable.

Part 2: Longer questions (16 points)

For 2.1 and 2.2, there might be different results, because you may use different algorithms. Any of them is correct.

2.1. The values below are inserted (in the order specified) into an empty order 4 B+ Tree.
(17, 23, 19, 44, 55, 30, 24, 12, 50, 39). Draw the tree. I only want the final result. (5 points)
2.2. In the above resulting tree, I delete 24, 30 and 50. Please draw the tree. Again, draw the final result. (5 points)

The above program is running in a single core system. P1 and P2 are two concurrent processes, and share a block of memory, $m$. Before running P1 and P2, the value of $m$ is 5. For a given process $i$, $\text{Read}(x)$ means this process $i$ reads the value of $m$ and assigns this value to $i$'s local variable $x$; $\text{Update}(x)$ means the process $i$ writes the value of $x$ to the memory $m$. Assume all the commands are atomic.

Please answer the following questions:

1. After P1 executes $\text{Update}(x)$, what's the value on the shared memory? (2 points) Why? (2 points)

The value should be 0, because P1 reads $x = 5$, and writes $0 = 5 - 5$ back to the shared memory.

2. After P2 executes $\text{Update}(x)$, what's the value on the shared memory? (2 points) Why? (2 points)

The value should be 10, because P2 reads $x = 5$ (at that time, the value of $m$ was not changed by P1), and writes $10 = 5 + 5$ back to the shared memory.