

<http://www.cs.arizona.edu/classes/cs460/fall08/>

Homework #1

(110 points)

Due Date: September 30th, 2008, at the beginning of class

Remember: All homeworks in this class must be electronically submitted (via ‘turnin’ on lectura)! This homework requires that you create a few diagrams. You may use a diagramming tool for this purpose; several open-source and/or free options exist (xfig, dia, etc.). You may also draw your diagrams by hand and scan/photograph them for inclusion within your solutions document, so long as the result is readable.

Write complete, legible answers to each of the following questions. A problem identified as “M.N” is exercise N from chapter M of the Elmasri/Navathe text, 5th edition. Show your work, when appropriate, for possible partial credit. This is not a group project; do your own work. We will post our solutions 24 hours after the due date (remember, you can use one late day on homeworks, so we can’t give solutions on the due date).

On the due date, by the start of class, hand-in a printout of your solutions **and** run ‘turnin’ on lectura to submit your electronically-formatted version of your solutions (the submission folder is `cs460h1`). If you need to submit your solutions within the 24-hour late window, place your printout in one of our mailboxes in CS 713 as soon as you are able to do so. Solutions submitted more than 24 hours after the due date and time will not be accepted.

1. (5 points) 1.12 (provide two examples)
2. (5 points) 2.3
3. (5 points) 2.15
4. (5 points) 3.11 (provide two examples)
5. (10 points) 3.16
6. (5 points) 3.25
7. (5 points) 3.28 (just parts: a, c, e, g, and i.)
8. (10 points) 4.17
9. (5 points) 4.26
10. (5 points) 13.26 (just parts a, b, and d)
11. (10 points) In Program #2, you extendibly-hashed using leading digits of a decimal-valued key. For this exercise, assume that our keys are in Base 3 instead of Base 10. Also assume that buckets are disk blocks that can hold at most three keys each. Build an extendible hash index structure using the keys listed below, and draw the final structure. Be sure to include all depth value labels.

2112, 0012, 0122, 1102, 2100, 1210, 0120, 1221, 0111, 0001, 2210, 1112
12. (20 points) 14.14 (just parts a, b, c, d, and g)
13. (10 points) 14.15. Assume a B⁺-tree of Order 2 (using our in-class definition), insert the values in the order presented, stop when your tree has three levels (the root counts as a level), and show this final tree. You will not need to insert all of the data! You may show intermediate trees if you wish to do so; doing so may allow us to give you partial credit in the event that you make an error.
14. (10 points) Delete, from your final tree from the last problem, the following keys, and show the final tree: 60 and 18 (in that order). Again, you may show intermediate trees if you so desire.