Carnegie Mellon University  
Department of Computer Science  
15-826 Multimedia Databases and Data Mining  
C. Faloutsos, Fall 2019

Homework 2
Due: hard copy, in class, at 1:30pm, on Wed 09/25/2019
Due: tarball, on Canvas, at 1:30pm, on Wed 09/25/2019

VERY IMPORTANT:
• For each question, we expect both a hard copy, and a tar file with your code - see details next, on how to package your code.
• Deposit hard copy of your answers, in class.
  1. Separate your answers, on different page(s) for each question
  2. Type the full info on each page: your name, Andrew ID, course#, Homework#, Question# on each of the pages.

Reminders:
• Plagiarism: Homework is to be completed individually.
• Typeset your answers. Illegible handwriting may get zero points.
• Late homeworks: please email it
  – to all TAs
  – with the subject line exactly 15-826 Homework Submission (HW 2)
  – and the count of slip-days you are using.

For your information:
• Graded out of 100 points; 2 questions total
• Rough time estimate: 10-20 hours (≈ 5-10 hours per question)

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>KD-Trees</td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>Z-ordering</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>Total:</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>
Code packaging instructions:
Submit your code to canvas, in a single tar file, called `andrewId-hw2.tar.gz` (where `andrewId` is your andrew id.) For your convenience, we provide a tar-file package, at https://www.cs.cmu.edu/~christos/courses/826.F19/HOMEWORKS/HW2/hw2.tar.gz. We will refer to it as the tar-file package from now on. It has 2 directories ./Q1, ./Q2, including the k-d-tree and Z-ordering source code, and place-holder code.

- `tar xvf hw2.tar.gz; cd Q1; make` # to work on kdtrees
- Replace the placeholder code with your solutions, `tar` everything into `andrewId-hw2.tar.gz` and submit to canvas. We expect to do `tar xvfz; make` and see your answers.

Hints:
We strongly recommend that you explore the makefiles we have created, for your convenience. For example, from the top directory:

- `make hw2` # will run the code for all the questions
- `make package` # will create the tar file, for submission
- `make spotless` # will clean up all the derived files (*.o, etc)

Make sure that you exclude redundant/derived files, in your tar file. Also, it is your responsibility to make sure that all necessary files are included in your tarball.

Thus, before submitting your file, do `tar xvfz; make` to make sure it works correctly.
Question 1: KD-Trees ........................................... [60 points]

On separate page, with ‘[course-id] [hw#] [question#] [andrew-id] [your-name]’

Problem Description: Consider the k-d-tree package written in C in the directory ./Q1 of the given [tar-file package]. Some functions, such as nn query, count and range query, have already been implemented. You may explore them by using the command lines (.main).

- Your task is to add a new function called minvalue, and indicated by m. It should take as input the dimension (0, 1, ...) and return the minimum value along that dimension.

For example, in a 2-dimensional tree with the following three points

- 0.3, 0.7
- 0.9, 0.2
- 1.5, -0.1

the minimum value along dimension 0 is: 0.3; and along dimension 1, it is: -0.1.

Input Format: For your convenience, we have already implemented the loading of input data in main.c. However, you should still check for input errors (eg., negative dimension-index).

$ ./main -d 2
num. dimensions = 2
i for insertion
n for nn search
r for range search
c for count
m for minimum value (to be implemented)
p to print the tree
x to exit
h to print this help message
kdtree> m
Find min value along dimension: 0
result: N/A (to be implemented)

In the example above, the user asks for the minimum value (m), on the 0-th dimension, on an empty, 2-dimensional k-dtree.
(a) [35 points] Modify the k-d-tree code to support the \textit{minvalue} queries. The output format is as the example above: a single line with the string “result:”, the minimum value along that dimension, and a new-line at the end, i.e. \texttt{“result:%%lf\n”}. Remember to remove all other debugging messages (if any), before you submit the code.

\textbf{IMPORTANT:} Test your code, for corner cases (empty tree, illegal query input, etc), and of course, on the provided datasets (\texttt{test[123].txt}). We will also test your code against other ‘secret’ data which will disclose after the due date.

(b) [25 points] A hard copy of the output of your code, on the three included input scripts (\texttt{test1.txt}, \texttt{test2.txt}, and \texttt{test3.txt}; they are in the ./Q1 directory). Ignore irrelevant lines and keep only the result of minvalue queries, to save paper (eg., \texttt{grep result})

\textbf{What to turn in:}

\begin{itemize}
  \item \textbf{Code:} Submit your code to canvas, in the ./Q1 directory of your tar.gz file.
  \item \textbf{Answers:} Submit hard copy for
    \begin{enumerate}
      \item your code (only relevant parts),
      \item and the output of your code on three input files.
    \end{enumerate}
\end{itemize}

Homework 2 continues...
Question 2: Z-ordering ............................... [40 points]

On separate page, with ‘[course-id] [hw#] [question#] [andrew-id] [your-name]’

Problem Description: The provided tar-file package contains a incorrect implementation of the Z-ordering.

(a) [25 points] Fix the bug(s) in the provided code for Z-ordering in zorder.c. (We injected one or more bugs, on purpose). The function zorder() is supposed to return the Z-order value for the given point. You may use the inverse version of z-order, izorder() function, for reference.

(b) [15 points] Use zplot.c to plot a 2-d Z-curve of 7-order (128 * 128 grid); The output should be produced in zcurve.png. We provide some template code, the makefile, and some helper functions. We recommend gnuplot for plotting.

What to turn in:

- **Code:** Submit your code to canvas, in the ./Q2 directory of your tar.gz file. Please change only the necessary files.

- **Answers:** Hard copy of
  1. your code in zorder.c,
  2. your code in zdist.c,
  3. your plot for (b)