80 minutes duration (15:00-16:20)
You should have 9 non-empty pages, including this cover
Graded out of 100 points
Numbers in square brackets indicate points.

Contents

Q1 Linear hashing ..................................................[5 pts] 2
Q2 B-trees .............................................................[10 pts] 3
Q3 Kd-trees .............................................................[15 pts] 4
Q4 Z- and Hilbert-ordering .......................................[15 pts] 5
Q5 Power laws .........................................................[15 pts] 6
Q6 Fractals - warm-up ...............................................[2 pts] 7
Q7 Fractals .............................................................[18 pts] 8
Q8 SQL for Document Retrieval ...............................[20 pts] 9
A hash table using linear hashing started with the following hash function:

$$h_0(x) = x \text{ mod } 3$$

The buckets were numbered 0, 1, 2. At some later point of time, the hash table has \(N=13\) buckets, numbered 0, 1, \ldots, 12.

1. ([1 pts]): How many hash functions are active?
2. ([2 pts]): Which one(s)?
3. ([2 pts]): Where is the split pointer? (0, 1, 2, \ldots)? That is, which bucket is next in line to be split?
Consider a B-tree of order $d=2$, that is, every node has at most $2 \cdot d + 1 = 5$ pointers, and at most $2 \cdot d = 4$ keys. Thus, such a B-tree with only one level ($l=1$) consists of the root node only, and it can hold at most 4 keys.

1. (5 pts) What is the maximum count of keys it can hold when it has $l=2$ levels?
2. (5 pts) Repeat, for $l=3$ levels?
Consider a 2-dimensional address space and the data points $P_1 = (10,20)$, $P_2 = (20,10)$, $P_3 = (5,30)$.

1. (8 pts) Draw the resulting k-dtree after these points are inserted, in the above order. Assume that the discriminator alternates, starting with the first ($x$) dimension. Follow the format of foil 29, 110_SAMS1.pdf: For each node, show the $x$ and $y$ coordinates; for each edge, show the condition, say $x \geq 35$. Show the edges and the conditions, even for empty sub-trees.

2. (7 pts) Also draw the data points and the splits, in the address space, as in foil 29 above.
Q4  Z- and Hilbert-ordering ________________[15 pts]

Consider a 2-d grid of sides (0,7) × (0,7). Recall that the z- or Hilbert values always start from 0.

1. ([3 pts] ) Give the z-values in decimal of the cells (1,1); (3,3); (7,7)
2. ([8 pts] ) Give the Hilbert values in decimal of the same cells.
3. ([4 pts] ) What is the Hilbert value of (15,15), in a (0,511)×(0,511) grid? (Hint: try to discover the pattern.)
Q5  Power laws

Consider a collection of \( N=10^6 \) lakes, whose area follows Korcak’s law with slope 1. Assume that the area \( a_{\text{min}} \) of the smallest lake is 1 unit of area. That is,

\[
\operatorname{Prob}(\text{area} \geq a) = C * a^{-1} \quad a \geq a_{\text{min}} = 1
\]

where \( C \) is a constant, to be estimated.

1. ([2 pts]) Estimate the value of \( C \).
2. ([4 pts]) Estimate the area \( a_{\text{max}} \) of the largest lake, up to 2 significant digits.
3. ([4 pts]) Explain your answer.
4. ([5 pts]) Estimate the median area \( a_{\text{med}} \) (Hint: by definition, the median \( a_{\text{med}} \) satisfies: \( \operatorname{Prob}(a \geq a_{\text{med}}) = 0.5 \)).
Q6 Fractals - warm-up [2 pts]

Consider a large number of points uniformly distributed in the unit square. Point $P$ has $n=100$ neighbors within radius $r=0.01$ (that is, neighbors at radius 0.01 or less). Estimate the number $n'$ of its neighbors within radius $r'=0.02$. 
Consider a large number of points on the Sierpinski triangle, in $E$-dimensions, with (correlation) fractal dimension $D_2 = \log 3 / \log 2 \approx 1.5849$. Point $P$ has $n=50$ neighbors within radius $r=10$ (that is, neighbors at radius 10 or less). We want to estimate the number $n'$ of its neighbors within radius $r'=20$.

1. ([8 pts] ) What is your estimate for $n'$ (up to 3 significant digits)?
2. ([10 pts] ) Justify your answer
Consider the (correct, but inefficient) representation of documents as \((document, term)\) pairs, in the relation \(DT(docID, term)\). See Figure 1 for an example, where, say, \(d1\) is ‘Moby Dick’, \(d2\) is ‘The old man and the sea’, and so on.

<table>
<thead>
<tr>
<th>docID</th>
<th>term</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>sea</td>
</tr>
<tr>
<td>d1</td>
<td>captain</td>
</tr>
<tr>
<td>d1</td>
<td>whale</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>d2</td>
<td>old</td>
</tr>
<tr>
<td>d2</td>
<td>man</td>
</tr>
<tr>
<td>d2</td>
<td>sea</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Figure 1: Example of a document-term table

1. **[5 pts]** Write the SQL query that will return all the docIDs of the documents that have at least one word in common with \(d1\).

2. **[15 pts]** Complete and/or correct the SQL code of Figure 2 that wants to retrieve all the documents that have at least two terms in common with \(d1\). Make sure that it retrieves each document once, and that it does not retrieve \(d1\).

```
1. select DT3.docID
2. from DT as DT1, DT as DT2, DT as DT3, DT as DT4
3. where DT1.docID = 'd1'
4.     and DT2.docID = 'd1'
5.     and DT1.term = DT3.term
6.     and DT2.term = DT4.term
7.     and DT3.docID = DT4.docID
8.     ....
```

Figure 2: (possibly correct) SQL code to find documents with at least two words in common with ‘\(d1\)’

END OF EXAM QUESTIONS __________________________GOOD LUCK