CS3743 Hwk 2 Physical Performance (30 pts) – due 2/20

**© Copyright 2020 Larry Clark, this document must not be copied to any other website**

Consider a **Marketing** table that contains the following attributes:

custNr customer number, **9 bytes**, uniquely defines a customer. When combined with historyCd, it is the unique key for Marketing.

historyCd **5-byte** code (e.g., MSHOES, WSHOES, TV, WASHER, COMPUTER, MSHIRT, WSHIRT). When combined with custNr, this is the unique key for this table.

ageRangeCd customer's age range (e.g., 1:under 20, 2:20-29, 3:30-39, 4:40-49, 5:50-59, 6:60-69, 7:over 70), this is **1 byte**

lastActivityDt **4-byte** date representing the last activity for this row. (Dates are stored as yyyymmdd with each digit being a 4-bit nibble.)

email customer 's email, max of 220-character, average of **18 bytes**

streetAddress customer's street address, max of 120 characters. These average **48 bytes**.

**Assumptions:**

1. There are **10 million rows** in the Marketing Table.
2. Data and index blocks are **4096** bytes.
3. An index entry for a B+Tree contains: the key (size depends on the contents) and a **Row Id** which is **8 bytes**.
4. There are two variable-sized attributes (email and streetAddress). Each will use a **one-byte length**. As discussed in class for handling variable-size attributes, there will be an additional **2-byte offset** to access the streetAddress.

**See below**

**Part 1: Space**  (show your work for partial credit) **(15 pts )**

1. How big is one data row? You must consider:
   * size of each fixed-size attribute

custNr **= 9 +** historyCd = **5 +** ageRangeCd **= 1 +** lastActivityDt =  **4, total = 16 bytes**

* + For each variable-sized attribute, consider the **average size plus one for the length**. Also, each attribute after the first variable-sized attribute must have a **2-byte** offset.

email = **18+1**, streetAddress = **48+1, total = 68 bytes + 2 = 70**

* + Include a row size field of **2 bytes per row**.

**70 + 16 + 2**

Average row size = **88** bytes per row.

1. How many rows fit in a data block? You must consider:
   * average size of a row (from part 1A) (88)
   * **20-byte** overhead in the data block (4096 – 20)
   * **two bytes** for each entry in the offset array per row. (see how we represent a RowId in the course notes.) (88+2)

**4076 / 90**

Rows per data block = **45** rows per data block

1. How many data blocks are needed?

**10 million rows with 45 rows per data block, 10,000,000 / 45**

Number of data blocks = **222,223** data blocks

1. For a unique index (called **I1**) on the combination of custNr and historyCd:
   * how many index entries will there be at the leaf level? **10,000,000**
   * what is the size of an index entry? (refer to assumption #3) **22** bytes per index I1 entry

**custNr = 9, + historyCd = 5, assumption #3 includes a row Id of 8 bytes.**

**So the index entry is 9 + 5 + 8 = 22 bytes**

* + how many entries fit in an index block if we assume 20 bytes of overhead per index block?  
    **185** index entries per idx I1 block
  + how many leaf level index blocks are needed?   
    **54,055** leaf level index blocks in I1

1. For index I1, how many total **levels are needed in the index?**

First fill in this table, noting that some of the rows may be blank:

|  |  |  |  |
| --- | --- | --- | --- |
| **#levels** | **Level** | **# index entries** | **#index blocks (#idx entries / idx entries per blk)** |
| 1 | Leaf | 10,000,000 | 54,055 |
| 2 | level above leaf | 54205 | 293 |
| 3 | two levels above leaf | 370 | 2 |
| 4 | three levels above leaf | 185 | 1 |
| 5 | four levels above leaf | ?? | ?? |

Number of index levels for index I1= **4 Levels**

1. What is the total number of index blocks in index I1?   
   **54,055 + 293 + 2 + 1 = 54,351** index blocks for index I 1

**Part 2: Reads**  (show your work for partial credit) **(15 pts )**

Additional assumptions:

1. We have another index(called **I2**) on the combination of ageRangeCd and historyCd:
   * The index nodes for this index have **291** entries per index block.
   * There are **three** index levels for I2.
2. Do not worry if your answer to this is different in part 1. These values are arbitrary (but done to reduce taking off for also having a mistake in the first part):

* Assume an average of **40** data rows per data block.
* Assume **four** index levels in the unique index, **I1**, on **custNr** and **historyCd**.

1. Assume there are 100,000 customers in ageRangeCd=5. 10% (10,000) of those customers are interested in WSHOES.

Questions:

For **A-F** below, we want to find all customers in **ageRangeCd=5 who are interested in WSHOES** using the index **I2**. State your assumptions.

Find all customers in Age Range 5, (50-59), who are interested in WSHOES.

10,000 interested customers within age range, 4 rows per customer, 10,000\*5 = 50,000 entries to read.

1. We have three levels in index **I2** with a total of 10,000,000 index entries.
   * How many Marketing rows satisfy the query on the average based on the query’s criteria?   
     Marketing rows satisfying criteria = 10,000 rows
2. How many leaf-level index blocks are there satisfying this query ? (approximate)

* Number of leaf-level entries satisfying this query? (answer from part 2A) = 50,000 entries
* Number of leaf-level index blocks satisfying this request   
  = #entries satisfying / #index entries per I2 index block (Part 2 assumption 1)  
  = ?? leaf-level I2 index blocks satisfying this query

1. How many index blocks (in total) are read to satisfy this query?

* It isn't necessary to re-read index blocks when processing this query.
* I2 has three levels

Leaf level reads = ?? (from part 2B)

Level 2 reads = ??

Level 1 reads (root level) = ??

Total = ??

1. Determine the percentage of data rows satisfying our criteria.  
   How many rows satisfy our criteria? ?? (from 2A)  
   How many total rows in the data area? ?? (from assumptions)  
   What is the percentage of data rows that satisfy our criteria? ?? %
2. Assuming **I1** is the clustering index (i.e., I2 isn't the clustering index), how many total reads using index I2 are necessary to read our rows that satisfy the criteria?  
   Index reads = ?? (from Part 2C)

Data reads = ??

Total = ?? total reads

1. Assuming **I2** is the clustering index, how many total reads are necessary to satisfy the query?

Index reads = ?? (from part 2C)

Data Reads = ??  
Total = ?? total reads

For **G** and **H**, how many reads are necessary to find **a particular row** by the combination of custNr and historyCd using index **I1**?

1. Assuming **I1** is the clustering index, how many total reads are necessary to find a particular row using I1?

Index reads = ??

Data reads =??  
Total =??

1. Assuming **I2** is the clustering index, how many total reads are necessary to find a particular row using I1?

Index reads = ??

Data reads =??

Total = ??