Exercise 10

Date of Issue: 09.05.2017 Deadline: 16.05.2017, 12:15

1 Physical Database Design

Consider a relation $\mathbf{r}$ with relation schema $R(A, B, C)$ and the following additional information:

- $|\mathbf{r}| = 3,000,000$
- Relation $\mathbf{r}$ is stored unsorted on disk.
- The domain of attribute A are integers with range between 1 and 12,000,000
- There are 70% as many distinct values for attribute A as there are tuples, i.e., $|\pi_{\mathrm{A}}(\mathbf{r})| = 0.7 \cdot |\mathbf{r}|$. These values are spread uniformly throughout the entire range.
- Size of attribute A: 4 bytes
- Size of a tuple: 32 bytes
- Size of a disk block: 4096 bytes
- Size of a B+-tree pointer: 8 bytes
- The size of a B+-tree node corresponds to the size of a disk block
- The B+-tree fits entirely into main memory (no block transfer cost).

The following two queries are evaluated on relation $\mathbf{r}$:

(a) $Q_1 : \sigma_{4,000,000 \leq A \wedge \mathrm{A} < 7,000,000}(\mathbf{r})$

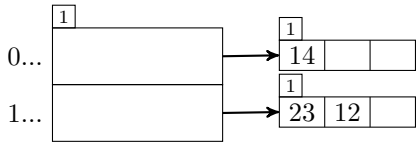
(b) $Q_2 : \sigma_{\mathrm{A}=5,000,000}(\mathbf{r})$

Tasks:

1. Determine the minimum and maximum path length in a B+-tree on attribute A and relation $\mathbf{r}$. Explain your approach.
2. Determine the number of nodes that are traversed in the worst case in the B+-tree on attribute A in order to fetch all tuples for query Q_1. Explain your approach.
3. Determine the average number of blocks fetched for query Q_1 using the B+-tree index on attribute A. Explain your approach.
4. Determine the number of blocks fetched for query Q_1 *without* using the B+-tree index on attribute A. Explain your approach.
5. Determine the average number of blocks fetched for query Q_2 using the B+-tree index on attribute A. Explain your approach.
6. Determine the number of blocks fetched for query Q_2 *without* using the B+-tree index on attribute A. Explain your approach.

2 Hash Index

Consider the 4 bit hash function $h(x) = \text{bin}((x + 2) \bmod 15)$ and the following extendable hashing scheme with bucket size 3:



Determine the extendable hashing scheme after each of the following insertions:

(a) Insert 61, 51, 75

(b) Insert 69

(c) Insert 9

(d) Insert 28

In the hash function, bin(y) returns the binary value of y and mod indicates the modulo operation. For instance, the values 12, 14, and 23, which are contained in the hashing scheme, are hashed as follows:

$$h(12) = \text{bin}(14 \bmod 15) = \text{bin}(14) = 1110$$

$$h(14) = \text{bin}(16 \bmod 15) = \text{bin}(1) = 0001$$

$$h(23) = \text{bin}(25 \bmod 15) = \text{bin}(10) = 1010$$