Extra exercises 2: Storage Hierarchy

Question 1: Sorted file organization only supports range searches.

A) True
B) False

Question 2: The indexing field in a clustered indexing is always a key.

A) True
B) False

Question 3: Hash-based indexes are not efficient for range searches.

A) True
B) False

Question 4: Assume that we have a Tree-based index that is clustered and sparse. The index key is also the primary key of the table. The data entries and the data files are stored in a Heap file. Which of the following is not true? (multiple choice)

A) The index is efficient for range searches.
B) The cost (number of needed I/Os) of retrieving records in the range searches is the number of matching index data entries.
C) The cost (number of needed I/Os) of an insert is 2.
D) The index is efficient for frequent insert/delete

FDB is a database that keeps information about the Films in the following table:

Film(<u>flimId</u>, directorName, year, title, length)

The size of each record of the table is **80B: fimId is 10B, directorName is 30B, year is 10B, title is 20B, and length is 10B**. All database page sizes are equal to **2048B**. Each page can store a whole or an integer number of records (a record can not be split into two pages.), along with **24B** as a footer and **24B** as a header of metadata.

Assume that we have 100000 records available from 2014 until the end of 2023 and the number of films per year is uniformly distributed.

Question 5: How many records fit on each page?

Question 6: How many pages are needed for this database?

Question 7: If we use a heap file organization, what is the maximum I/O cost of the following query?

```
SELECT AVG(length)
FROM Film
WHERE year=2023
```

A) The cost of finding the qualifying records.
B) The total cost of the query.

Question 8: If we use a sorted file organization based on filmId, what is the maximum I/O cost of the following query?

```
DELETE
FROM Film
WHERE year=2023
```

A) The cost of finding the qualifying records.
B) The total cost of the query.

Question 9: Now assume that we want to transmit the database to another machine that uses the DSM page layout and each page size is **1024B.** The footer and the header size are both **12B**.

A) How many pages are needed for the sub-table (filmId, directorName)?
B) How many pages are needed for the sub-table (filmId, year)?
C) How many pages are needed for the sub-table (filmId, title)?
D) How many pages are needed for the sub-table (filmId, length)?
E) How many pages are needed for moving all data into this device?

Data Structure: [No Index(default) | B-Tree | Hash]

Format: [No Index(default) | Clustered | Unclustered]

On what key: [No Index(default) | filmId | directorName | year | length]

Density: [No Index(default) | Sparse | Dense]

Question 10: Assume that we have 20M tuples with infrequent insert/updates and queries of format SELECT * FROM Film WHERE year = 2019. Construct the best available index.

Question 11: Assume that we have 20 tuples with frequent updates and queries of format SELECT MAX(length) FROM Film WHERE year = 2019 AND length > 60. Construct the best available index.

Question 12: Assume that we have 10000 tuples with frequent insert\delete and queries of format SELECT * FROM Film WHERE year > 2010 AND length > 60. Construct the best available index.

Extra exercises 2: Storage Hierarchy
Solutions

Answer 1: B
Answer 2: B
Answer 3: A
Answer 4: B, D
Answer 5: (2048-48)/80 = 25
Answer 6: 100000/25 = 4000
Answer 7:

A) 4000
B) 4000

Answer 8:

C) 4000
D) 4000 + 4000 / 10 = 4400

Answer 9:

A) (1024 - 24) / 40 = 25 records per page => 100000 / 25 = 4000
B) (1024 - 24) / 20 = 50 records per page => 100000 / 50 = 2000
C) (1024 - 24) / 30 = 33 records per page => 100000 / 33 = 3031
D) (1024 - 24) / 20 = 50 records per page => 100000 / 50 = 2000
E) Total = 11031 pages

Answer 10: Hash | Clustered | year | Sparse
Answer 11: No index
Answer 12: B-Tree | UnClustered | year | Dense