1_Introduction

Due Jan 14 at 3pm **Points** 2 **Questions** 2 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	less than 1 minute	2 out of 2
LATEST	Attempt 2	less than 1 minute	2 out of 2
	Attempt 1	less than 1 minute	1 out of 2

Score for this attempt: **2** out of 2
Submitted Jan 10 at 11:46am
This attempt took less than 1 minute.

Question 1 — 1 / 1 pts

What was the acronym used to describe transactional processing systems?

- TP
- OLAP
- OLTP (Correct!)
- DBMS

Question 2

1 / 1 pts

What company is the largest database software vendor by ***sales volume***?

- Microsoft
- Oracle

Correct!

- IBM
- Google

Quiz Score: **2** out of 2

2_Storage

Due Jan 19 at 3pm **Points** 10 **Questions** 10 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	less than 1 minute	10 out of 10
LATEST	Attempt 2	less than 1 minute	10 out of 10
	Attempt 1	2 minutes	4 out of 10

Score for this attempt: **10** out of 10
Submitted Jan 16 at 7:44am
This attempt took less than 1 minute.

Question 1 — 1 / 1 pts

Does an in-memory database need a secondary storage device for persistence?

Correct!

- (•) Yes
- ( ) No

Question 2 — 1 / 1 pts

What device would be the fastest to read 1 MB of data?

- DRAM with bandwidth of 20 MB/sec.
- SSD with read 400 IOPS for 100 KB data chunks.

Correct!

Question 3

1 / 1 pts

What is the parity bit with *odd parity* and the bit string: 11111110?

Correct!

- 0
- 1
- 2

Question 4

1 / 1 pts

What RAID level offers the high performance but no redundancy?

Correct!

- RAID 0
- RAID 1
- RAID 5
- RAID 6

Question 5

1 / 1 pts

JSON and XML are best described as:

- fixed format, fixed size
- fixed format, variable size
- variable format, fixed size
- variable format, variable size

Correct!

Question 6

1 / 1 pts

A relational table uses a VARCHAR field for a person's name. It can be best described as:

- fixed format, fixed size
- fixed format, variable size
- variable format, fixed size
- variable format, variable size

Correct!

Question 7

1 / 1 pts

We can represent a person's name in MySQL using either CHAR(50) or VARCHAR(50). Assume that the person's name is 'Joe'. How much space is actually used?

- CHAR = 3 ; VARCHAR = 3

- CHAR = 50 ; VARCHAR = 3
- **Correct!** CHAR = 50 ; VARCHAR = 4
- CHAR = 50 ; VARCHAR = 50

Question 8

1 / 1 pts

What does the **VACUUM** command do in PostgreSQL?

- Cleans up your dirty house for you
- Deletes records from a given table
- **Correct!** Reclaims space used by records marked as deleted
- Removes tables no longer used

Question 9

1 / 1 pts

What buffer replacement policy does MySQL InnoDB use?

- **Correct!** LRU
- MRU
- 2Q

Question 10

1 / 1 pts

Does PostgreSQL support column layout?

- Yes
- No

Correct!

Quiz Score: **10** out of 10

2_Storage_RAID

Due Jan 19 at 3pm **Points** 2 **Questions** 2 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	1 minute	2 out of 2

Score for this attempt: **2** out of 2
Submitted Jan 16 at 7:46am
This attempt took 1 minute.

Question 1 — 1 / 1 pts

A SSD has storage capacity of 1 TB. What is the **total capacity** of RAID 0 with 6 drives (in **TB**)?

Correct!

6

Correct Answers

6

6 TB

RAID 0 - non-redundant (full capacity of drives) = drive capacity * 6 = 1 TB * 6 = **6 TB**

Question 2

1 / 1 pts

A SSD has storage capacity of 1 TB. What is the **total capacity** of RAID 5 with 10 drives (in **TB**)?

Correct!

9

Correct Answers

9

9 TB

RAID 5 - parity redundancy - uses 1 drive for parity so have 9 data drives = drive capacity * 9 = 1 TB * 9 = **9 TB**

Quiz Score: **2** out of 2

2_Storage_Spanning_Utilization

Due Jan 28 at 3pm **Points** 8 **Questions** 8 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 7	1 minute	8 out of 8
LATEST	Attempt 7	1 minute	8 out of 8
	Attempt 6	less than 1 minute	1 out of 8
	Attempt 5	less than 1 minute	0 out of 8
	Attempt 4	1 minute	7 out of 8
	Attempt 3	1 minute	3 out of 8
	Attempt 2	1 minute	3 out of 8
	Attempt 1	941 minutes	3 out of 8

Score for this attempt: **8** out of 8
Submitted Feb 18 at 10:07am
This attempt took 1 minute.

Question 1 — 1 / 1 pts

Given a block size of 4096 bytes and a record size of 400 bytes. How many records can be stored per block (***blocking factor***) using unspanned configuration?

Correct!

10

orrect Answers 10 (with margin: 0)

blocking factor (unspanned) = floor(block size/record size) = floor(4096 bytes/block / 400 bytes/record) = **10 records/block (unspanned)**

Question 2

1 / 1 pts

Given a block size of 4096 bytes and a record size of 400 bytes. How many records can be stored per block (***blocking factor***) using **spanned** configuration?

Correct! 10.24

orrect Answers 10.24 (with margin: 0.1)

blocking factor (spanned) = block size/record size = 4096 bytes/block / 400 bytes/record = **10.24 records/block (spanned)**

Question 3

1 / 1 pts

Given a block size of 4096 bytes and a record size of 400 bytes. What is the **utilization (percentage)** for the unspanned configuration?

Correct! 97.7

rrect Answers 97.7 (with margin: 0.1)

utilization = used space/total space = (40 records/block * 400 bytes/record) / 4096 bytes/block * 100% = **97.7%**

Question 4

1 / 1 pts

The database stores an employee table consisting of 500,000 records each of which is 400 bytes long. A SSD has storage capacity of 1 TB, read bandwidth of 1 GB/sec., and a block size of 4096 bytes.

Assume the data is stored sequentially. What is the **time** (**in seconds**) to read all records in the **unspanned configuration**?

Correct! 0.2048

rrect Answers 0.2048 (with margin: 0.01)

#blocks (unspanned) = 500,000 records / 10 records/block = 50,000 blocks
transfer time = 50,000 blocks * 4096 bytes/block / 1 GB/sec. = **0.2048 seconds** (Note must calculate by transfering blocks not just records)

Question 5

1 / 1 pts

The database stores an employee table consisting of 500,000 records each of which is 400 bytes long. A SSD has storage capacity of 1 TB,

read bandwidth of 1 GB/sec., and a block size of 4096 bytes.

Assume the file is physically ordered on key *K*, what is the **average time** (**in ms**) to retrieve a record using key *K* in the **spanned configuration**?

Correct!

0.0655

rrect Answers

0.0655 (with margin: 0.001)

#blocks (spanned) = ceiling(500,000 records / 10.24 records/block) = 48,829 blocks
blocks searched for binary search = ceiling(log2(48,829 blocks)) = 16 blocks
binary search time = 16 blocks * 4096 bytes/block / 1 GB/sec. = **0.0655 ms**

Question 6

1 / 1 pts

The database stores an employee table consisting of 500,000 records each of which is 400 bytes long. A SSD has storage capacity of 1 TB, read bandwidth of 1 GB/sec., and a block size of 4096 bytes. The database designer wants to add a large description field of 1,000 bytes to each employee record. However, this field will only have data in it 20% of the time and is only accessed in 25% of the queries. The goal is to find the best record layout. These next three questions will ask you to calculate the time for different alternatives.

Calculate: The **average time** (**in seconds**) to read the whole table when using spanned, fixed records and no splitting.

Correct!

0.7

orrect Answers 0.7 (with margin: 0.01)

Choice 1: Spanned fixed length record
record size = 1400 bytes;
#blocks = ceiling(500,000 records * 1400 bytes/record / 4096 bytes/block) = 170,899 blocks
transfer time = 170,899 blocks * 4096 bytes/block / 1 GB/sec.
= **0.70 seconds**

Question 7

1 / 1 pts

Calculate: The **average time** (**in seconds**) to read the whole table when using spanned, variable records. Assume a variable length record requires 20 bytes of overhead to store record information.

Correct!

0.31

orrect Answers 0.31 (with margin: 0.01)

Choice 2: Variable length record (spanned)
average record size = 20 bytes (overhead) + 400 bytes + 1000 * 0.2 = 620 bytes
#blocks = ceiling(500,000 record * 620 bytes/record / 4096 bytes/block) = 75,684 blocks
transfer time = 75,684 blocks * 4096 bytes/block / 1 GB/sec.
= **0.31 seconds**

Question 8

1 / 1 pts

Calculate: The **average time** (**in seconds**) to read the whole table when using spanned records and splitting. Assume using splitting requires 20 bytes of overhead. The large 1000 byte field is only access 25% of the time.

Correct!

0.235

Correct Answers

0.235 (with margin: 0.02)

Choice 3: Split record (spanned for both splits)
Keep current records as is, and add new blocks for new field.
Will only store new field if present (variable record).
record size for main record = 420 bytes
#blocks main file = 500,000 records * (420 bytes) / 4096 bytes/block = 51,270 blocks
main transfer time = 51,270 blocks * 4096 bytes/block / 1 GB/sec. = **0.21 seconds**

record size for variable record file = 1000 bytes * 0.2 = 200 bytes
#blocks variable file = 500,000 records * 200 bytes/record / 4096 bytes/block = 24,415 blocks
variable file transfer time = 24,415 blocks * 4096 bytes/block / 1 GB/sec. = **0.10 seconds**

average time = 25% * newQueryTime + 75% * regularQueryTime
= 0.25 * (0.21 s + 0.10 s) + 0.75 * 0.21 s = **0.235 seconds**

Quiz Score: **8** out of 8

3_Index_Calculations

Due Jan 28 at 3pm **Points** 9 **Questions** 9 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 5	1 minute	9 out of 9
LATEST	Attempt 5	1 minute	9 out of 9
	Attempt 4	4 minutes	8.5 out of 9
	Attempt 3	less than 1 minute	5 out of 9
	Attempt 2	20 minutes	4 out of 9
	Attempt 1	1,491 minutes	2 out of 9

Score for this attempt: **9** out of 9
Submitted Feb 18 at 8:56am
This attempt took 1 minute.

Question 1 — 1 / 1 pts

Consider storing a database table of 5,000,000 records each of which is 500 bytes long. The disk block size is 4096 bytes, and the read bandwidth is 2 GB/sec. Records are stored using no spanning and in sorted order using an integer key field.

Calculate the **number of blocks** used to store the table.

Correct!

625,000

rrect Answers 625,000 (with margin: 0)

blockingFactor = floor(4096 bytes/block / 500 bytes/record) = 8 records/block
#blocks = #records / blockingFactor = 5,000,000 records / 8 records/block = **625,000 blocks**

Question 2

1 / 1 pts

Using previous question info, calculate the **number of blocks** to find a given record using linear search.

Correct! 312,500

rrect Answers 312,500 (with margin: 0)

linear search blocks = 625,000 blocks / 2 = **312,500 blocks**
Note: Below is time but student does NOT need to calculate.
linear search time = 312,500 blocks * 4096 bytes/block / 2 GB/sec. = 0.64 seconds

Question 3

1 / 1 pts

Using previous question info, calculate the **number of blocks** to find a given record using binary search.

Correct!

20

rrect Answers

20 (with margin: 0)

binary search blocks = ceiling(log2(625,000 blocks)) = **20 blocks**
Note: Below is time but student does NOT need to calculate.
binary search time = 20 blocks * 4096 bytes/block / 2 GB/sec. = 0.04096 ms

Question 4

1 / 1 pts

Assume an index record occupies 8 bytes (4 for key, 4 for disk pointer). What is the **size and number of blocks read** to find a record for the following a dense, primary index.

Size (in # of blocks): 9766

Blocks read: 15

Answer 1:

rrect Answer

9766

dense primary index has one index record per data record = 5,000,000 index records
#index blocks = 5,000,000 index records / (4096 bytes/blocks / 8 index records/block) = 9766 index blocks

Answer 2:

Correct!

15

dense primary index has one index record per data record = 5,000,000 index records
#index blocks = 5,000,000 index records / (4096 bytes/blocks / 8 index records/block) = 9766 index blocks

binary search blocks on index = ceiling(log2(9766 blocks)) = 14 blocks
total blocks read = 14 blocks + 1 block (for data record) = **15 blocks**
Note: Below is time but student does NOT need to calculate.
search time = 15 blocks * 4096 bytes/block / 2 GB/sec. = 0.0307 ms

Question 5

1 / 1 pts

Assume an index record occupies 8 bytes (4 for key, 4 for disk pointer). What is the **size and number of blocks read** to find a record for the following a **sparse, primary index**.

Size **(in # of blocks)**: 1221

Blocks read: 12

Answer 1:

Correct! 1221

Answer 2:

rrect Answer 12

binary search blocks on index = ceiling(log2(1221 blocks)) = 11 blocks
total blocks read = 11 blocks + 1 block (for data record) = **12 blocks**
Note: Below is time but student does NOT need to calculate.
search time = 12 blocks * 4096 bytes/block / 2 GB/sec. = **0.0246 ms**

sparse primary index has one index record per data block = 625,000 records
#index blocks = 625,000 index records / (4096 bytes/block / 8 index records/block) = 1221 index blocks

Question 6

1 / 1 pts

There is an integer foreign key in the table that is a candidate for secondary indexing. What is the **time (in seconds)** required to retrieve one record with the foreign key value ***without an index***?

Correct!

0.64

Correct Answers

0.64 (with margin: 0.1)

Time to retrieve a secondary key value without an index is the **same as linear search time or 0.64 seconds.** linear search time = 312,500 blocks * 4096 bytes/block / 2 GB/sec. = 0.64 seconds

Question 7

1 / 1 pts

What is the **time (in ms)** required to retrieve a record with a secondary index?

Correct!

0.0327

rrect Answers

0.0327 (with margin: 0.01)

total time with secondary index = index search time + read primary index block + read record block = 16 blocks * 4096 bytes/block / 2 GB/sec. = **0.0327 ms**

Question 8

1 / 1 pts

What is the **size in MB** of the secondary index?

Correct!

40

rrect Answers

40 (with margin: 1)

Secondary index occupies 9766 blocks * 4096 bytes/block = 40,001,536 bytes = **38.1 MiB** or **40 MB**

Question 9

1 / 1 pts

Determine how many **levels** of multi-level index are required for a **dense index** and a **sparse index** index, so that the minimum # of disk accesses are performed to find a record. **Hint:** *Determine the index level where all index pointers at that level fit into a single block.*

Correct!

3

Correct Answers

3 (with margin: 0)

Dense index: The number of blocks was calculated for a dense primary index in c.
#index blocks on level 1 = 9766 ; #blocks on level 2 = 9766 / (4096/8) = 19 blocks; #blocks on level 3 = 1
The number of levels is 3 for dense index.
Sparse index: Level 1 = 1221 blocks ; Level 2 = 3 blocks ; Level 3 = 1 block
The number of levels is 3 for sparse index.

Quiz Score: **9** out of 9

3_Indexing

Due Jan 21 at 3pm **Points** 6 **Questions** 6 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	less than 1 minute	6 out of 6
LATEST	Attempt 2	less than 1 minute	6 out of 6
	Attempt 1	43 minutes	2 out of 6

Score for this attempt: **6** out of 6
Submitted Jan 21 at 1:43pm
This attempt took less than 1 minute.

Question 1 — 1 / 1 pts

What statement is true for a non-empty, indexed table when searching for a single record?

- Using an index is always faster than scanning the file if the data is on a hard drive
- Using an index is always faster than scanning the file if the data is on a SSD
- Binary searching an index is more suited to a hard drive than a SSD.

Correct!

- None of the above.

Calculate the performance of a sparse index on an **ordered** file with the following parameters:

- Each disk block stores 2000 data bytes.
- Each index entry occupies 8 bytes.
- Each record has size 100 bytes.
- The data file contains 1,000,000 records.

Question 2

1 / 1 pts

How many block reads to retrieve a record based on its key?

Correct!

9

Answer

#indexBlocks = 1,000,000 records / 20 records/block / 250 entries/block

= 200 blocks

#diskBlocks = 1,000,000 records / 20 records/block = 50,000 blocks

Search sparse index using a binary search = log2N = log2(200) = 7.6 blocks

# of blocks retrieved = 8 index blocks + 1 data block = **9 blocks**

orrect Answers

9 (with margin: 0)

0 (with margin: 0)

Question 3

1 / 1 pts

Using previous data, how many block reads for a dense index?

Correct!

13

Search dense index using a binary search = log2N = log2(4000) = 12 blocks

- 4000 as one index entry per record as compared to per block

# of blocks retrieved = 12 index blocks + 1 data block = 13 blocks

orrect Answers

13 (with margin: 0)
0 (with margin: 0)

Question 4

1 / 1 pts

Calculate the performance of a multi-level index on an **ordered** file with the following parameters:

- Each disk block stores 2000 data bytes.
- Each index entry occupies 8 bytes.
- Each record has size 100 bytes.
- The data file contains 10,000,000 records.
- There are 3 levels of multi-level index.
- First level is a sparse index - one entry per block.

How many blocks must be read to retrieve a record based on its key?

Correct!

4

Answer

#indexBlocks1 = 10,000,000 records / 20 records/block / 250 entries/block

= 2,000 blocks

#indexBlocks2 = 2,000 blocks / 250 entries/block = 8 blocks

#indexBlocks3 = 1 (only holds 8 pointers)

#diskBlocks = 10,000,000 records / 20 records/block = 500,000 blocks

# of index blocks retrieved = 1 at level 3 + 1 at level 2 + 1 at level 1 + 1 data

= Total of 4 blocks retrieved

rrect Answers

4 (with margin: 0)
0 (with margin: 0)

Question 5

1 / 1 pts

Using the previous numbers, how many block reads if using a single level sparse index?

Correct!

12

Search single-level sparse index using a binary search

= log2N = log2(2000) = 10.96 blocks

# of blocks retrieved = 11 index blocks + 1 data block = 12 blocks

Time to find record using binary search (no index) = log2(500,000)

= 18.9 = 19 blocks

rrect Answers

12 (with margin: 0)
0 (with margin: 0)

Question 6

1 / 1 pts

A secondary index is constructed that refers to the primary index to locate its records. What is the minimum number of blocks that must be processed to retrieve a record using the secondary index?

- 0
- 1
- 2

Correct!

- 3

D – 3 – One secondary index block, then one primary index block, then one block to retrieve record. Note that may retrieve many secondary index blocks when searching and if multiple values then multiple primary index/data blocks as well.

- 4

Quiz Score: **6** out of 6

4_BTree_Questions

Due Feb 2 at 3pm **Points** 8 **Questions** 4 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	1 minute	8 out of 8
LATEST	Attempt 2	1 minute	8 out of 8
	Attempt 1	89 minutes	3.17 out of 8

Score for this attempt: **8** out of 8
Submitted Feb 18 at 9:10am
This attempt took 1 minute.

Question 1 — 5 / 5 pts

Insert the following keys into an empty B+-tree of **order 2 (maximum keys=4)**: 12, 22, 66, 7, 30, 68, 1, 75, 4, 87, 9, 88,99, and 92.

Answer these questions that describe your final tree.

What key(s) are in the root node? 66

How many leaf nodes? 6

How many nodes have three children? 2

Answer 1:

Correct! 66

Answer 2:

Correct! 6

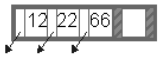
Answer 3:

Correct! 2

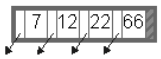
Insert 12

Insert 22

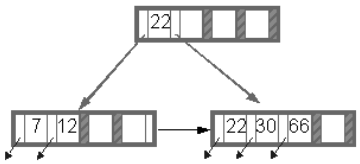
Insert 66



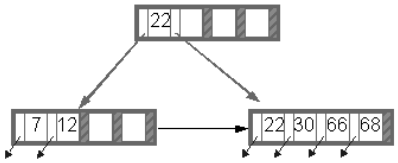
Insert 7



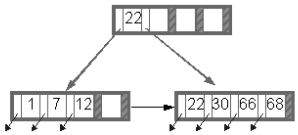
Insert 30



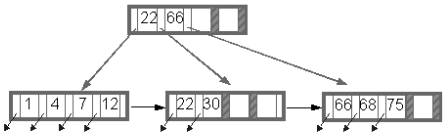
Insert 68



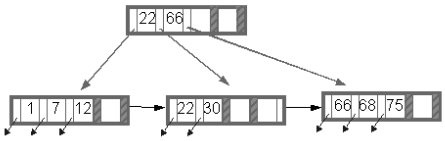
Insert 1



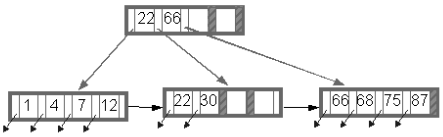
Insert 4



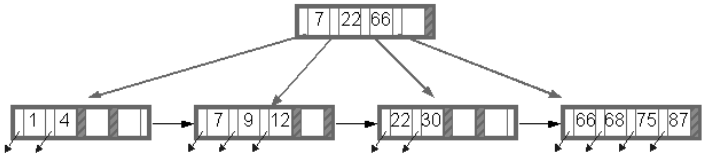
Insert 75



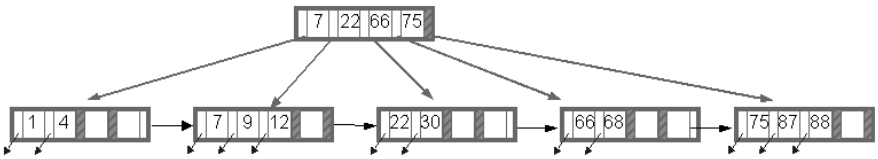
Insert 87



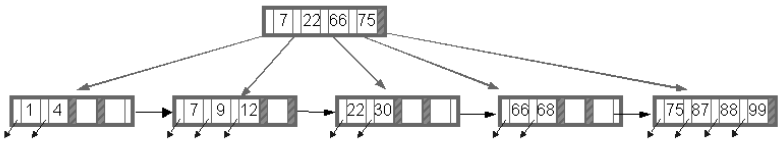
Insert 9



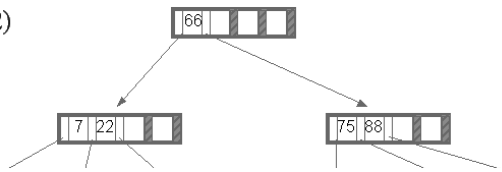
Insert 88

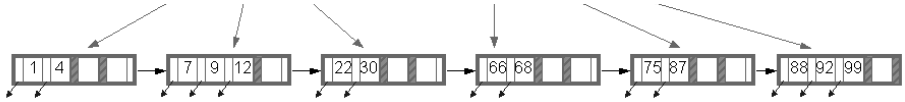


Insert 99



Final Tree
(Insert 92)





Question 2

1 / 1 pts

From previous tree, delete 7. Answer these questions that describe your final tree.

What key(s) are in the root node? 66

Is there a 7 in an interior or leaf node in the tree (yes/no)?

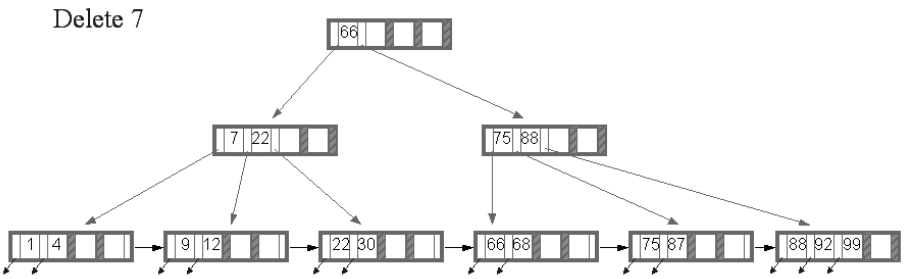
yes

Answer 1:

Correct! 66

Answer 2:

Correct! yes



Question 3

1 / 1 pts

From previous tree, delete 87. Answer these questions that describe

your final tree.

Was there an underflow to handle (yes/no)? yes

Did you borrow from a sibling or merge siblings? Remember our algorithm is to borrow before merge unless there are no keys to borrow.

Select borrow or merge: borrow

Answer 1:

Correct!

yes

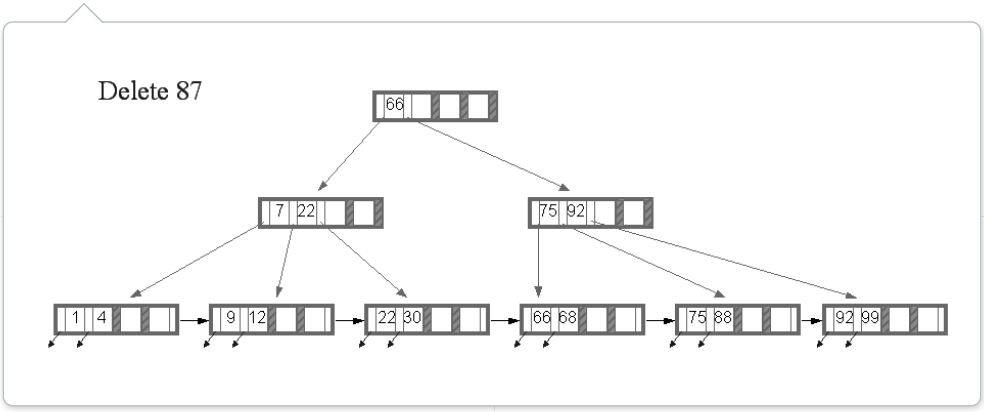
rrect Answer

y

Answer 2:

Correct!

borrow



Question 4

1 / 1 pts

From previous tree, delete 66. Answer these questions that describe your final tree.

Was there an underflow to handle (yes/no)? yes

What key(s) are in the root node (list comma separated)?

7,22,66,92

Answer 1:

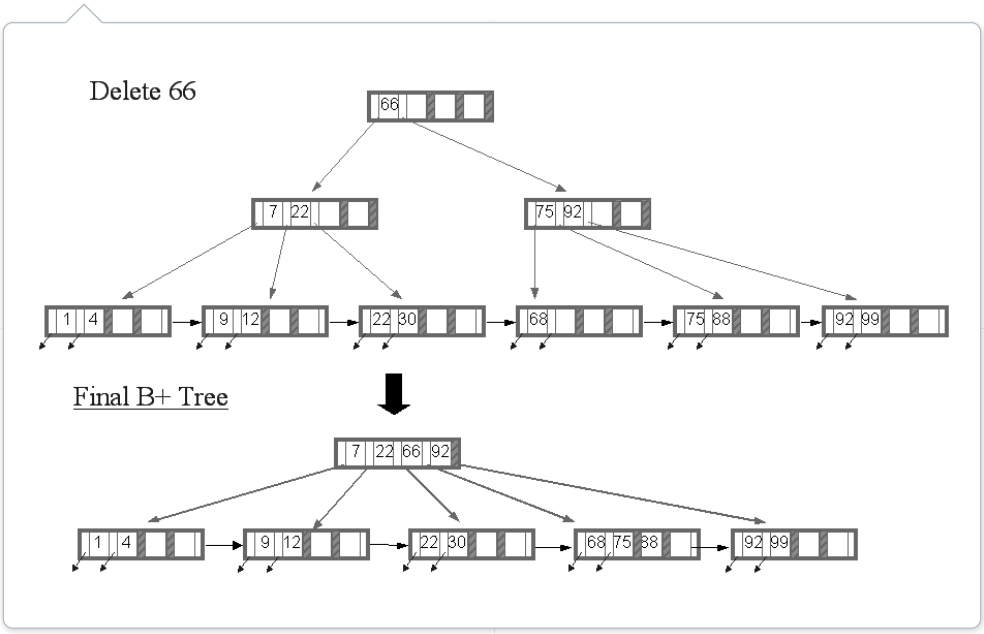
Correct! yes

rrect Answer y

Answer 2:

Correct! 7,22,66,92

rrect Answer 7, 22, 66, 92



Quiz Score: **8** out of 8

4_BTrees

Due Jan 28 at 3pm **Points** 6 **Questions** 6 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 3	less than 1 minute	6 out of 6
LATEST	Attempt 3	less than 1 minute	6 out of 6
	Attempt 2	less than 1 minute	4 out of 6
	Attempt 1	197 minutes	3.33 out of 6

Score for this attempt: **6** out of 6
Submitted Feb 18 at 9:05am
This attempt took less than 1 minute.

Question 1 — 1 / 1 pts

A B-tree has a maximum of 10 keys per node. What is the maximum number of children for a given node?

- 0
- 1
- 10
- 11 **Correct!**
- 20

Question 2

1 / 1 pts

For a B-tree of ***order 1 (max. keys=2)***, insert the following keys in order:

- 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150

What is the key in the root node? 80

How many nodes in total? 15

Answer 1:

80 **Correct!**

Answer 2:

15 **Correct!**

Question 3

1 / 1 pts

Using the previous tree constructed by inserting into a B-tree of ***order 1 (max. keys=2)*** the keys:

- 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150

Delete these keys (in order) and indicate how many nodes changed during the operation:

- 40 - Nodes changed: 7
- 70 - Nodes changed: 1
- 80 - Nodes changed: 6

Answer 1:

Correct!

7

Answer 2:

Correct!

1

Answer 3:

Correct!

6

rrect Answer

5

Question 4

1 / 1 pts

Given a block of 4096 bytes, calculate the maximum number of keys in a node if the key size is 4 bytes, internal B-tree pointers are 8 bytes, and we store the record itself in the B-tree node instead of a pointer. The record size is 100 bytes.

- 18
- 36
- 340
- 680

Correct!

Assuming no header information is stored in the block:

4*k + 100*k +8*(k+1) <= 4096

112k <= 4088

k = 36 (Max. order is 18).

If we assume that we do not have to store the key separately from the record,

100*k +8*(k+1) <= 4096

108k <= 4088

k = 37 (Max. order is also 18 because 19 would require 38 keys.)

Question 5

1 / 1 pts

For a B+-tree of ***order 2 (max. keys=4)***, insert the following keys in order:

- 10, 20, 30, 40, 50, 60, 70, 80, 90

Assuming keys increasing by 10, what is the first key added that causes the B+-tree to grow to height 3?

- 110

- 120
- 130
- 140
- 150

Correct!

Question 6

1 / 1 pts

Assume you start with the tree after inserting 90 above.

Delete the following keys in order. How many changes for each operation?

- a) 70 - Changes: 1
- b) 90 - Changes: 3
- c) 10 - Changes: 3

Answer 1:

Correct!

1

Answer 2:

Correct!

3

Answer 3:

Correct!

3

Quiz Score: **6** out of 6

5_R-Trees

Due Jan 28 at 3pm **Points** 2 **Questions** 2 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	47 minutes	1.5 out of 2

Score for this attempt: **1.5** out of 2
Submitted Jan 28 at 11:11am
This attempt took 47 minutes.

Question 1 — 1 / 1 pts

What type of spatial query is: "Find the city closest to Chicago?"

- ○ Spatial Range Query
- ◉ Nearest Neighbor Query
- ○ Spatial Join Query
- ○ Not a spatial query

Correct!

Question 2 — 0.5 / 1 pts

Select all true statements.

ou Answered

- [x] Searching in a R-tree always follows a single path.

Correct!

- [x] R-tree variants may have different ways for splitting nodes during insertion.

Correct!

- [x] A R+-tree search always follows a single path to a leaf node.
- [ ] None of the above

Quiz Score: **1.5** out of 2

6_Hashing

Due Feb 2 at 3pm **Points** 3 **Questions** 3 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

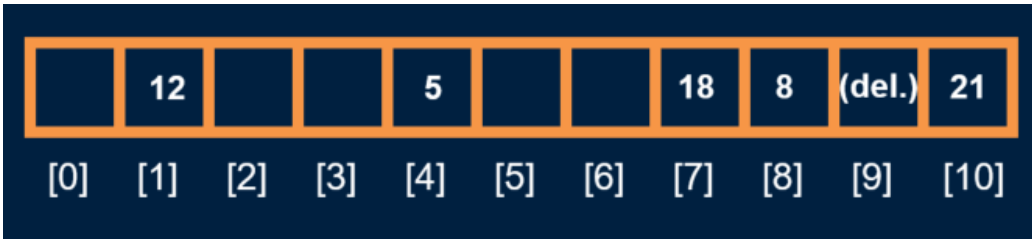
Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	48 minutes	3 out of 3
LATEST	Attempt 2	48 minutes	3 out of 3
	Attempt 1	15 minutes	0 out of 3

Score for this attempt: **3** out of 3
Submitted Feb 2 at 1:27pm
This attempt took 48 minutes.

Question 1 — 1 / 1 pts

What location is **19** inserted at?



- 8
- 9 **Correct!**
- 6

- 0

Question 2

1 / 1 pts

A linear hash table has 5 blocks each with space for 4 records. There are currently 2 records in the hash table. What is its load factor?

Correct!

- (•) 10%
- 40%
- 50%
- 0%

Question 3

1 / 1 pts

Show the resulting hash directory when hashing the keys: 0, 15, 8, 4, 7, 12, 10, 11 using linear hashing.

- Assume a bucket can hold two records (keys).
- Assume 4 bits of hash key.
- Add a new bucket when utilization is >= 85%.

What bucket is 11 in?

- 000
- 001
- 1011

Correct!

- 011

Quiz Score: **3** out of 3

6_Linear_Hashing

Due Feb 9 at 3pm **Points** 5 **Questions** 1 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 3	less than 1 minute	5 out of 5
LATEST	Attempt 3	less than 1 minute	5 out of 5
	Attempt 2	2 minutes	3.57 out of 5
	Attempt 1	104 minutes	2.14 out of 5

Score for this attempt: **5** out of 5
Submitted Feb 18 at 9:14am
This attempt took less than 1 minute.

Question 1 — 5 / 5 pts

Insert the following keys: 1, 7, 3, 29, 27, 23, 19, 20, 11 into a linear hash table with 2 records per block (split with load factor >= 85%). Assume the hash value contains 5 bits.

How **many** keys in each bucket including overflow blocks?

000: 0

001: 1

010: 0

011: 6

100: 1

101: 1

111: 0

Answer 1:

Correct! 0

Answer 2:

Correct! 1

Answer 3:

Correct! 0

Answer 4:

Correct! 6

Answer 5:

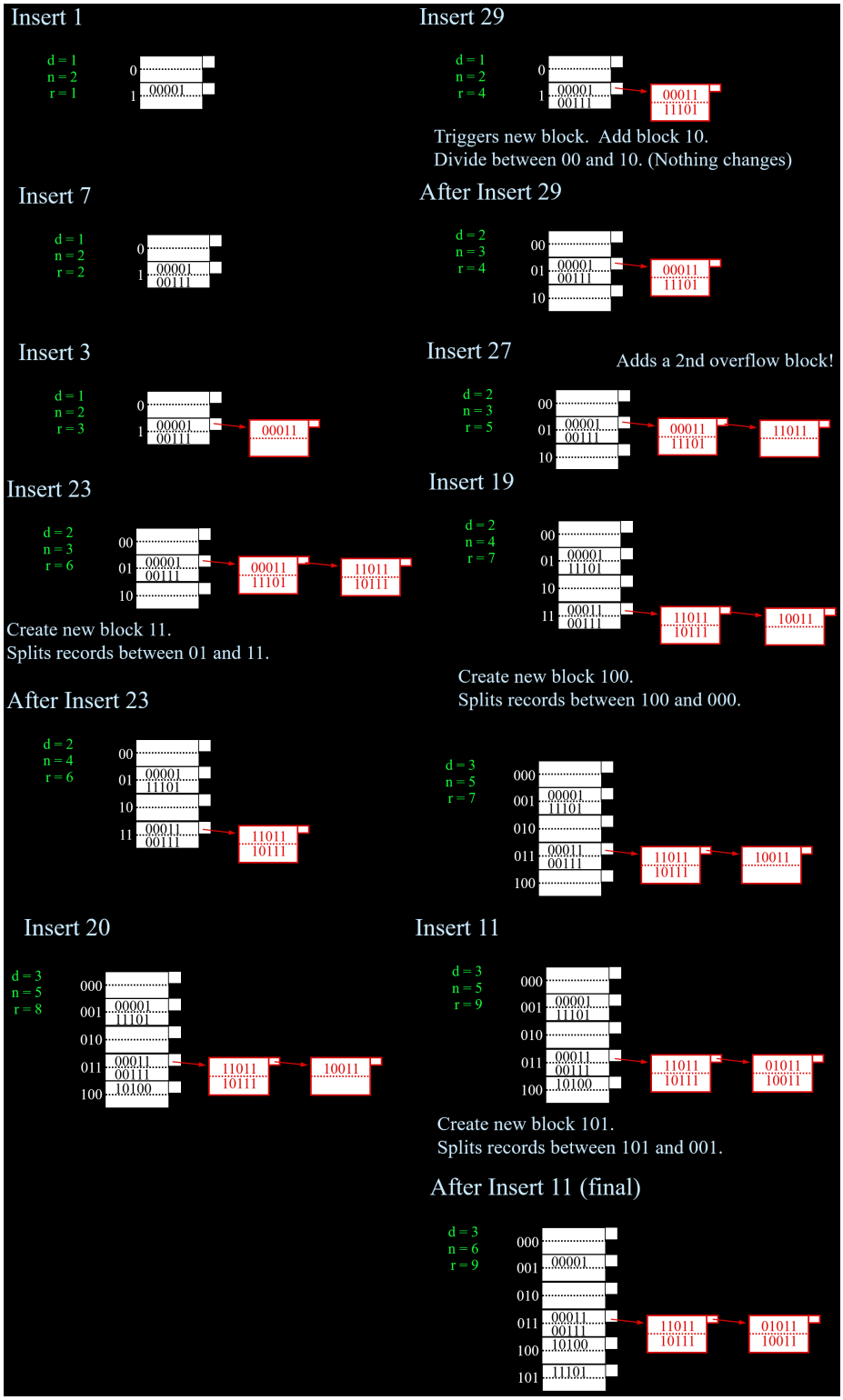
Correct! 1

Answer 6:

Correct! 1

Answer 7:

Correct! 0



Quiz Score: **5** out of 5

7_Multi_Attribute_Indexing

Due Feb 9 at 3pm **Points** 3 **Questions** 2 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 4	less than 1 minute	3 out of 3
LATEST	Attempt 4	less than 1 minute	3 out of 3
	Attempt 3	less than 1 minute	1 out of 3
	Attempt 2	2 minutes	1 out of 3
	Attempt 1	120 minutes	1 out of 3

Score for this attempt: **3** out of 3
Submitted Feb 18 at 9:53am
This attempt took less than 1 minute.

Question 1 — 1 / 1 pts

Consider storing a relation *R(x,y)* in a grid file. Attribute x has a range of values from 0 to 100. Attribute y has a range of values from 200 to 700. The partitions for the grid file are uniformly spaced; for *x* there are partitions every 5 units, at 5, 10, 15, and so on, while for *y* the partitions are every 50 units at 250, 300, 350, and so on. How many buckets do we have to examine to answer the range query:

```
SELECT *
FROM R
WHERE 63 < x AND x < 92 AND 395 < y AND y < 625
```

Correct!

42

rrect Answers

42

42 buckets must be searched.
For *x*, we must search 7 partitions (60-64, 65-69, 70-74, 75-79, 80-84, 85-89, and 90-94). For *y*, we must search 6 partitions (350-399, 400-449, 450-499, 500-549, 550-599, 600-649).
Thus, we search a grid of 7 * 6 = 42 buckets.

Question 2

2 / 2 pts

Suppose we store a relation *R(x,y,z)* in a partitioned hash table with 1024 buckets (i.e. 10-bit bucket addresses). Queries about *R* specify only attribute x half of the time. The remaining queries query either y or z with the queries split evenly between these attributes. Determine the average number of buckets that need to be searched to answer a query for the two following strategies.

The hash function produces 4 bits based only on *x*, 3 bits based only on *y*, and 3 bits based only on *z*. Buckets searched: 96

The hash function produces 6 bits based only on *x*, 2 bits based only on *y*, and 2 bits based only on *z*. Buckets searched: 136

Answer 1:

Correct! 96

Answer 2:

Correct! 136

- If the hash function produces 4 bits based only on *x*, 3 bits based only on *y*, and 3 bits based only on *z*.*(2 marks)*
  For *x* with 4 bits, 6 bits remain of the hash location that are not specified. Thus, 2^6 = 64 buckets to search.
  For *y* with 3 bits, 7 bits remain of the hash location that are not specified. Thus, 2^7 = 128 buckets to search.
  For *z* with 3 bits, 7 bits remain of the hash location that are not specified. Thus, 2^7 = 128 buckets to search.
  On average, 0.5*64 + 0.25*128 + 0.25*128 = **an average of 96 buckets to search.**
- The hash function produces 6 bits based only on *x*, 2 bits based only on *y*, and 2 bits based only on *z*.*(2 marks)* For *x* with 6 bits, 4 bits remain of the hash location that are not specified. Thus, 2^4 = 16 buckets to search.
  For *y* with 2 bits, 8 bits remain of the hash location that are not specified. Thus, 2^8 = 256 buckets to search.
  For *z* with 2 bits, 8 bits remain of the hash location that are not specified. Thus, 2^8 = 256 buckets to search.
  On average, 0.5*16 + 0.25*256 + 0.25*256= **an average of 136 buckets to search.**
- **The first strategy is better and requires a search of around 50% fewer buckets on average.**

Quiz Score: **3** out of 3

7_SQL_Indexing

Due Feb 2 at 3pm **Points** 1 **Questions** 1 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	2 minutes	1 out of 1
LATEST	Attempt 2	2 minutes	1 out of 1
	Attempt 1	81 minutes	0 out of 1

Score for this attempt: **1** out of 1
Submitted Feb 2 at 1:48pm
This attempt took 2 minutes.

Question 1 — 1 / 1 pts

Using partitioned hashing and the following two hash functions, determine how many buckets are searched for the query: **Major='BS' OR Year='1'**

```
Hash Table
h1 is hash function for Major.
h1(BA) = 0
h1(BS)=0
h1(CS)=1
h1(ME)=1
```

```
h2 is hash function for Year.
h2(1) = 00
h2(2) = 01
h2(3) = 10
h2(4) = 11
```

2

4

Correct!

5

Answer: 5 buckets. All 4 buckets that being with 0 plus 000 and 100. Since 000 is already done, only search 5.

6

8

Quiz Score: **1** out of 1

8_Query_Processing_1

Due Feb 4 at 3pm **Points** 8 **Questions** 8 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	59 minutes	8 out of 8
LATEST	Attempt 2	59 minutes	8 out of 8
	Attempt 1	3 minutes	3 out of 8

Score for this attempt: **8** out of 8
Submitted Feb 4 at 1:41pm
This attempt took 59 minutes.

Question 1 — 1 / 1 pts

Given this table and the query:

eno	ename	title	salary
E1	J. Doe	EE	30000
E2	M. Smith	SA	50000
E3	A. Lee	ME	40000
E4	J. Miller	PR	20000
E5	B. Casey	SA	50000
E6	L. Chu	EE	30000

E7	R. Davis	ME	40000
E8	J. Jones	SA	50000

```
SELECT eno, salary
FROM   emp
WHERE  salary >= 40000
```

How many rows in the result?

- 2
- 3
- 4
- 5

Correct!

Question 2

1 / 1 pts

Given these tables and the query:

Emp

eno	ename	title	salary	dno
E1	J. Doe	EE	30000	null
E2	M. Smith	SA	50000	D3
E3	A. Lee	ME	40000	D2
E4	J. Miller	PR	20000	D3
E5	B. Casey	SA	50000	D3
E6	L. Chu	EE	30000	D2
E7	R. Davis	ME	40000	D1

Dept

dno	dname	mgreno
D1	Management	E8
D2	Consulting	E7
D3	Accounting	E5
D4	Development	null

E8	J. Jones	SA	50000	D1		

$\Pi_{eno,\ ename}\ (\sigma_{title='EE'}\ (Emp \bowtie_{dno=dno} Dept)\ )$

How many rows in the result?

- 0
- 1

Correct!

- 2
- 8

Question 3

1 / 1 pts

What is the symbol for duplicate elimination?

- σ
- X
- π
- ⋈
- δ

Correct!

Question 4

1 / 1 pts

If T(R)=100 and V(R,a)=1 and we perform **$\delta(\Pi_a(R))$**, select a true statement.

- The maximum memory size used is 100 tuples (not counting input tuple).
- The size of the result is 100 tuples.
- The size of the result is unknown.

Correct!

- (selected) The maximum memory size used is 1 tuple (not counting input tuple).

Question 5

1 / 1 pts

How many buffers are required to perform a selection operation on a relation that has size 10,000 blocks?

Correct!

1

Only 1 block is required as selection is a tuple-at-a-time operation.

Correct Answers

1

Question 6

1 / 1 pts

Assume the number of buffers M=100. Let B(R)=10,000 and B(S)=90. How many block reads are performed for R ∪ S?

Correct!

10090

# of blocks = B(R)+B(S) = 10,090

rrect Answers

10090

Question 7

1 / 1 pts

If M=100, B(R)=5,000 and B(S)=1,000, how many block reads are performed for R - S using a one-pass algorithm? Provide a number or answer **N/A** if not possible.

Correct!

N/A

Not possible using a one-pass algorithm as neither relation fits entirely into memory buffers.

rrect Answers

N/A

no

Question 8

1 / 1 pts

Select a true statement.

Correct!

- (•) BNLJ buffers the smaller relation in memory.
- ( ) BNLJ buffers the larger relation in memory.

Quiz Score: **8** out of 8

8_Query_Processing_Operator_Costs

Due Feb 16 at 5pm **Points** 9 **Questions** 7 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 3	7 minutes	8 out of 9
LATEST	Attempt 3	7 minutes	8 out of 9
	Attempt 2	2,516 minutes	0 out of 9
	Attempt 1	163 minutes	4 out of 9

Score for this attempt: **8** out of 9
Submitted Feb 18 at 9:25am
This attempt took 7 minutes.

Question 1 — 1 / 1 pts

Relation *R1* has 5,000,000 tuples and relation *R2* has 5,000,000 tuples. How many disk I/Os in the *worst-case* does a tuple based Nested-Loop Join perform?

Correct! 25,000,000,000,000

Correct Answers 25,000,000,000,000 (with margin: 0)

Tuple Nested Loop Join = 5,000,000 * 5,000,000
= **25,000,000,000,000**.

Question 2

1 / 1 pts

Relation *R1* has 5,000,000 tuples and relation *R2* has 5,000,000 tuples. How many disk I/Os does a Block Nested-Loop Join perform? Assume that *R1* is stored in blocks with 50 tuples per block and *R2* is stored with 250 tuples per block, and you are allocated 1000 blocks of memory.

Correct!

2,120,000

rrect Answers

2,120,000 (with margin: 0)

Block Nested Loop Join = ceiling(B(R2)/(M-1)) * B(R1) + B(R2) = 2,100,000 + 20000 = **2,120,000**

Question 3

1 / 1 pts

Relation *R1* has 5,000,000 tuples and relation *R2* has 5,000,000 tuples. How many disk I/Os does a hybird hash join perform? Assume that *R1* is stored in blocks with 50 tuples per block and *R2* is stored with 250 tuples per block, and you are allocated 1000 blocks of memory.

Correct!

348,000

rrect Answers 348,000 (with margin: 0)

Hybrid Hash Join = (3 - 2M/B(R2)) * (B(R1) + B(R2)) = (3 - 0.1) * 120,000 = **348,000**

Question 4

2 / 3 pts

The following two relations *R* and *S* have one attribute storing integers. Assume that a join is performed on this attribute.

R = { 4, 3, 1, 6, 2, 9, 0, 5, 8, 7 }
S = { 3, 4, 5, 8, 0, 2, 1, 6, 7, 9 }

Show the order of tuples output for the following join algorithms: (for example Sort Join would give (0,0), (1,1),...,(9,9))

1. One pass join where *M*=12 and *R* is put in memory: (3,3),(4,4),(5,5),(8,8
2. Merge-sort-join with *M*=5: (0,0),(1,1),(2,2),(3,3
3. Hash Join with three buckets and *M*=4. Assume *R* is the build relation and *S* is the probe relation. Also assume bucket 0 is output first. Output: r (3,3),(0,0),(6,6),(9

Answer 1:

Correct! (3,3),(4,4),(5,5),(8,8),(0,0),(2,2),(1,1),(6,6),(7,7),(9,9)

rrect Answer (3,3), (4,4), (5,5), (8,8), (0,0), (2,2), (1,1), (6,6), (7,7), (9,9)

Answer 2:

Correct! (0,0),(1,1),(2,2),(3,3),(4,4),(5,5),(6,6),(7,7),(8,8),(9,9)

rrect Answer (0,0), (1,1), (2,2), (3,3), (4,4), (5,5), (6,6), (7,7), (8,8), (9,9)

Answer 3:

ou Answered r (3,3),(0,0),(6,6),(9,9),(4,4),(1,1),(7,7),(5,5),(8,8),(2,2)

rrect Answer (3,3),(0,0),(6,6),(9,9),(4,4),(1,1),(7,7),(5,5),(8,8),(2,2)

rrect Answer (3,3), (0,0), (6,6), (9,9), (4,4), (1,1), (7,7), (5,5), (8,8), (2,2)

Example expect syntax: (3,3),(4,4),(5,5)

Note that no spaces is preferred. System will accept space between tuples such as (3,3), (4,4), (5,5)

Question 5

1 / 1 pts

Given the following relations and query, calculate the cost of the query in terms of block I/Os of *Band* and *PlaysIn*. Your formula should contain variables like T(Band) (tuples of Band), B(Band) (blocks of Band), T(PlaysIn), and B(PlaysIn).

```
Band(groupName, country, genre)
PlaysIn(musicianName, bandName, instrument)

SELECT groupName, genre
FROM Band, PlaysIn
WHERE musicianName = "Eric Clapton" AND groupName = bandName;
```

Calculate the cost of the query using Sort Join for the query.

○ 3 * (B(PlaysIn) + B(Band))

Correct! ◉ 5 * (B(PlaysIn) + B(Band))

B(PlaysIn) + B(Band)

3 * B(PlaysIn) + B(Band)

Question 6

1 / 1 pts

Using the previous information, calculate the cost of the query using Sort-Join if *Band* is physically ordered by *groupName*.

5 * (B(PlaysIn) + B(Band))

3 * (B(PlaysIn) + B(Band))

Correct!

5 * B(PlaysIn) + B(Band)

3 * B(PlaysIn) + B(Band)

The Sort-join cost with Band sorted by groupName = **5 * B(PlaysIn) + B(Band)** as Band is sorted already.

Question 7

1 / 1 pts

Suppose relation *Band* has an index on *groupName*, calculate the cost of an index join for the query. Assume Eric Clapton PlaysIn five bandNames and each index search on *groupName* cost 4 disk operations to search for and retrieve a data record.

5 * (B(PlaysIn) + B(Band))

Correct!

- (●) B(PlaysIn) + 20
- ( ) B(PlaysIn) + 4
- ( ) B(Band) + 20
- ( ) B(Band) + 5

The cost of an index join is **B(PlaysIn) + numberOfTuples_from_PlaysIn * 4**, as we must scan all of `PlaysIn` and then each tuple that matches the WHERE criteria will be joined to `Band` using the index.
The index cost is based on the scan on `PlaysIn` to get the 5 "Eric Clapton" bands (`bandName`). The table scan costs `B(PlaysIn)`. For each of the five bands, use the index on `bandName` for `Band` to perform "join" by looking up matching tuples. The cost of using the index is 5 index searches each of which costs 4 block reads. Thus, the cost is **B(PlaysIn) + 20**.

Quiz Score: **8** out of 9

8_Query_Processing_Set_Operations

Due Feb 16 at 3pm **Points** 4 **Questions** 4 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	less than 1 minute	4 out of 4
LATEST	Attempt 2	less than 1 minute	4 out of 4
	Attempt 1	162 minutes	2 out of 4

Score for this attempt: **4** out of 4
Submitted Feb 18 at 9:16am
This attempt took less than 1 minute.

Given the two relations *R* and *S* below, perform the following set operations and give the result.

R

3
8
9
2
4
4
6
4

S

4
1
7
3
5
8
8
5

Question 1

1 / 1 pts

One pass bag intersection.

- {3, 8, 8, 4, 4}
- {3, 8, 4}
- {3, 4, 4, 4, 8, 8}
- {3, 4, 8, 5}

Correct!

Question 2

1 / 1 pts

One pass bag difference R - S.

- {1, 7, 5, 5, 8}
- {1, 7, 5, 5, 8}
- {9, 2, 4, 6}
- {9, 2, 4, 6, 4}

Correct!

Question 3

1 / 1 pts

Two pass set union using sorting with M=5 and bfr=1.

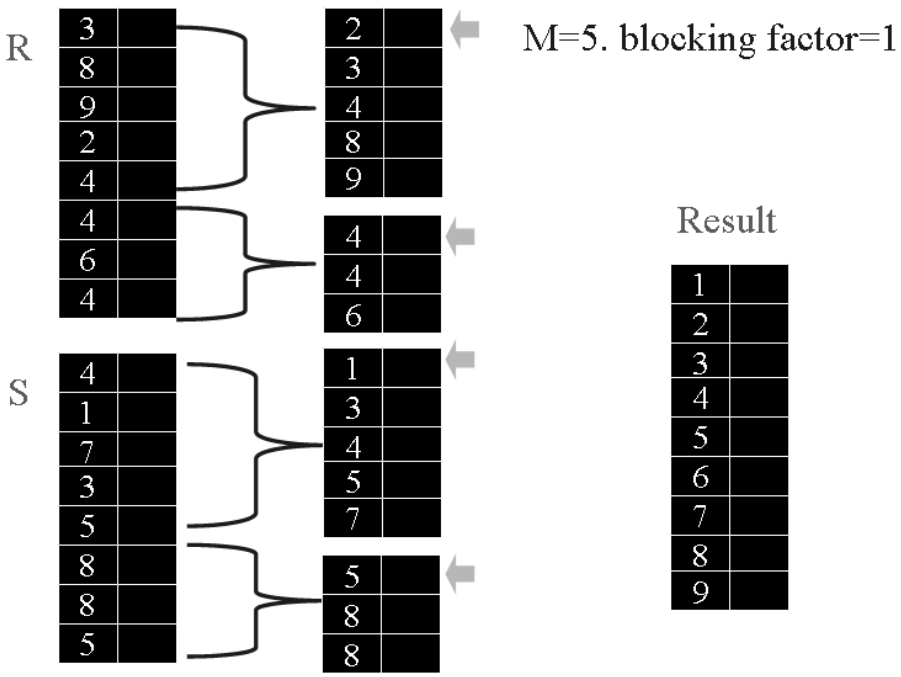
- {2, 3, 4, 6, 8, 9}

{1, 3, 4, 5, 7, 8}

Correct!

{1, 2, 3, 4, 5, 6, 7, 8, 9}

{1, 2, 3, 3, 4, 4, 4, 4, 5, 5, 6, 7, 8, 8, 8, 9}



Question 4

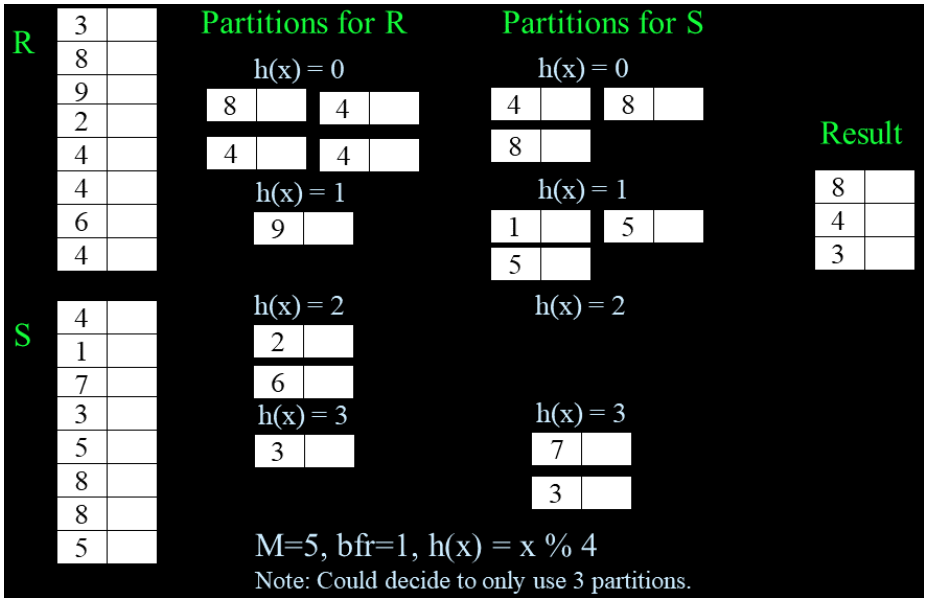
1 / 1 pts

Two pass set intersection using hashing with M=5 and bfr=1.

- {4, 4, 3, 3, 5, 8}
- {3, 5, 8}
- {3, 4, 8, 9}

Correct!

- {8, 4, 3}



Quiz Score: **4** out of 4

8_QueryProcessing_Hashing

Due Feb 11 at 3pm **Points** 3 **Questions** 3 **Time Limit** None
Allowed Attempts 10

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 3	less than 1 minute	3 out of 3
LATEST	Attempt 3	less than 1 minute	3 out of 3
	Attempt 2	2 minutes	2 out of 3
	Attempt 1	less than 1 minute	1 out of 3

Score for this attempt: **3** out of 3
Submitted Feb 11 at 2:11pm
This attempt took less than 1 minute.

Question 1 — 1 / 1 pts

Given ***M*** memory buffers, how many hash buckets are used when hash partitioning?

- 1
- **Correct!** M-1
- M
- M+1

Question 2

1 / 1 pts

Select a true statement.

- The probe relation is the smallest relation.
- The probe relation has an in-memory hash table built on its tuples.
- The build relation is the smallest relation.
- The probe relation is buffered in memory.

Correct!

Question 3

1 / 1 pts

Assume the percentage of join memory available compared to the smaller relation size is M / B(S). Select a percentage where block nested-loop join is faster than hybrid hash join in terms of disk I/Os.

- 10
- 25
- 70
- 100

Correct!

Answer: C 25 to 49% or D 50 to 100%

Quiz Score: **3** out of 3