Mathwork

by Jk J

Submission date: 09-Apr-2021 09:04PM (UTC-0500)
Submission ID: 1555082297
File name: Reworked.docx (237.66K)
Word count: 625
Character count: 2685

WORKED OUT ANSWERS

Part 1

Problem 1

1. Suppose B(R) = B(S) = 10,000. For what value of M would we need to compute R !" S using the nested-loop join algorithm with no more than the following number of I/Os? (8 points, 4 points each)

Using the equation given in Section 15.3.4 of the textbook, solve for M:

I/O = B(S) + $\frac{(B(S)B(R))}{(M-1)}$

(a) 100,000
$100{,}000 = 10{,}000 + \frac{(10{,}000 \times 10{,}000)}{(M-1)}$
M = 1,112.1 or ceil(M) = 1,113

(b) 25,000
$25{,}000 = 10{,}000 + \frac{(10{,}000 \times 10{,}000)}{(M-1)}$
M = 6,667.7 or ceil(M) = 6,668

2. If two relations R and S are both unclustered, it seems that the nested-loop join algorithmrequires about T(R)T(S)/M disk I/Os. How can you do signicantly better than this cost? Describe your modied version of the nested-loop algorithm and give the number of disk I/Os required for your algorithm. We assume that M is large enough such that M ? 1 ! M , and that B(R) ! T (R) and B(S) ! T (S); that is, the number of tuples of a relation is much greater than that of blocks of the relation. (8 points)

Note that the cost of algorithm given in the question is T(R)T(S)/M, which means it is using tuple-based nested-loop join. In order to improve the disk I/O cost of nested-loop join algorithm, we need to use block-based nested-loop join. In order to carry out block-based nested loop join efficiently, we need the inner relation clustered, and search structure built on the common attributes of R and S.

Let R be the inner relation (assuming S is smaller):

•• Cost of reading all tuples of R, cluster them, and write them back: T(R) + [redacted]) [redacted]
memory:

T(S) + $\frac{B(S)B(R)}{M}$

Therefore the total cost is T(R) + B(R) + T(S) + $\frac{B(S)B(R)}{M}$.

Problem 2.

We have two relations R and S where B(R) = B(S) = 10, 000. Give an approximate size of main memory M required and the number of disk I/Os in order to perform the two-pass algorithms for the following operations: (12 points, 4 points each)

(a) set union
Using the equation given in Section 15.4.9 of the textbook:

I/O of set union operation = 3= 3 ×(10,000 + 10,000) ×(B(S) + B(R))

= 60,000

Approximate M requirement for set union

operation is !(*B*)*R* + *B*(*S*) = √20,000 = 141.42, which the given M satisfies.

(b) simple sort-join
I/O of set union operation = 5 ×(B(S) + B(R))

= 5 ×(10,000 + 10,000)

= 100,000

Note that given figures satisfy the required M, which is B(S) and B(R) ≤ M^2,

i.e. 10,000 ≤ 10,00,000

(c) the more efficient sort-join described in Section 15.4.8 [redacted]
[redacted] of set union operation = 3 ×(B(S) + B(R))

$= 3 \times (10{,}000 + 10{,}000)$

$= 60{,}000$

Requirement for M in the e1,000,000 which is satisfied here.fficient sort-join algorithm is $B(R) + B(S) \leq M^2 = 20{,}000$

$\leq$

Problem 3

Two-pass Algorithms Based on Sorting (20 points)

1. Suppose we have a relation with 1,000,000 records and each records requires 10 bytes. Let the disk-block size be 4,096 bytes. (8 points, 4 points each)

(a) What is the minimum number of blocks in main memory required for using TPMMS(Two-Phase Multiway Merge-Sort) to sort these records?
The size of the relation in bytes is $1{,}000{,}000 \times 10 = 10{,}000{,}000$ bytes, and each disk-block is 4,096 bytes. The minimum number of blocks to hold the relation is ceil($\frac{10{,}000{,}000}{4{,}096}$) = 2,442$\sqrt{2{,}442}$) =. The minimum M requirement for TPMMS is Bceil(49.4) = 50. $\leq M^2$, M must

at least be ceil(

(b) Following (a), how many disk I/Os are needed to sort all the records?Number of disk I/O for TPMMS is 3B, which is $3 \times 2{,}443 = 7{,}329$

Problem 5

(10 points, 5 points each)

1. We consider two relations R(A, B, C) and S (C, D, E). Convert the following expressions in relational algebra by applying algebraic laws so that we can perform selections and projections as early as possible.

(a) $\sigma_{B=3 AND E=4}(R \text{ !" } \sigma_{C>10}(S))$

$\sigma_{(B=3) AND (C>10)}(R) \text{ !" } \sigma_{C>10 AND E=4}(S)$

(b) $\pi_{A,D}(R \text{ !" } S)$ $\pi_{A,D}(\pi_{A,C}(R) \text{ !" } \pi_{C,D}(S))$

Problem 6

Dynamic Programming (15 points) Compute the optimal plan for R [redacted] technique of dynamic programming. We make the [redacted] assumptions (as we did in the class):

- [redacted]) = 400, [redacted]) = 800, [redacted], and B(U) = 700.

- B(R1)The size of a join for two relations R1 and R2 is estimated as: B(R1×

  B(R2). If a subplan is a single relation and does not involve any join, the size!"

  R2) = 0.01 × of its intermediate result is zero.

- [redacted] theintermediate results.

- [redacted] cost of a scan is zero.

Draw the table for dynamic programming, to show how you compute the optimal plan for all possible join orders allowing all trees.

Subquery	Size	Lowest cost	Plan
RS	3200	0	RS
RT	2400	0	RT
RU	2800	0	RU
ST	4800	0	ST
SU	5600	0	SU
TU	4200	0	TU
RST	19200	2400	(RT)S
RSU	22400	2800	(RU)S
RTU	16800	2400	(RT)U
STU	33600	4200	(TU)S
RSTU	134400	7400	(RS)(TU)

Figure 1: dynamic programming plan

Mathwork

ORIGINALITY REPORT

13%	8%	3%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	7%
2	www.cs.uiuc.edu Internet Source	6%
3	www.cs.washington.edu Internet Source	4%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		