CPS216 Advanced Database Systems - Fall 2008
Assignment 1

- Due date: Thursday, Sept. 25, 2008, in class (2.50 PM). Late submissions will not be accepted.
- Submission: In class, or email solution in pdf or plain text to shivnath@cs.duke.edu.
- Do not forget to indicate your name on your submission.
- State all assumptions. For questions where descriptive solutions are required, you will be graded both on the correctness and clarity of your reasoning.
- Email questions to shivnath@cs.duke.edu.

Question 1 **Points 20**

Consider the following SQL query over tables $R(A)$, $S(A)$, and $T(A)$. Note that "Select Distinct" in SQL represents a duplicate-eliminating projection.

```
Select Distinct R.A
From   R, S, T
Where R.A = S.A and R.A = T.A
```

Figures 1(a)-(e) show five logical plans. The logical operator π in Figure 1 represents a duplicate-eliminating projection. For example, $\pi_{R.A}$ represents a duplicate-eliminating projection of attribute $R.A$. On the other hand, the logical operator τ represents a duplicate-preserving projection. The logical operator $\bowtie$ represents a natural join. Assume that, in the general case, R, S, and T can contain duplicate tuples.

1. Is the logical plan in Figure 1(a) equivalent to the logical plan in Figure 1(b)? (Equivalent means that they produce the same query result.) If not, what is the minimal set of constraints on the tables such that these plans are equivalent?
2. Is the logical plan in Figure 1(a) equivalent to the logical plan in Figure 1(c)? If not, what is the minimal set of constraints on the tables such that these plans are equivalent?
3. Is the logical plan in Figure 1(a) equivalent to the logical plan in Figure 1(d)? If not, what is the minimal set of constraints on the tables such that these plans are equivalent?
4. Is the logical plan in Figure 1(a) equivalent to the logical plan in Figure 1(e)? If not, what is the minimal set of constraints on the tables such that these plans are equivalent?

Question 2 **Points 20**

In this question we will find the cost of physical plans in terms of the number of getNext() calls. Figure 2 shows the physical plan generated from each logical plan in Figure 1. We make the following assumptions:

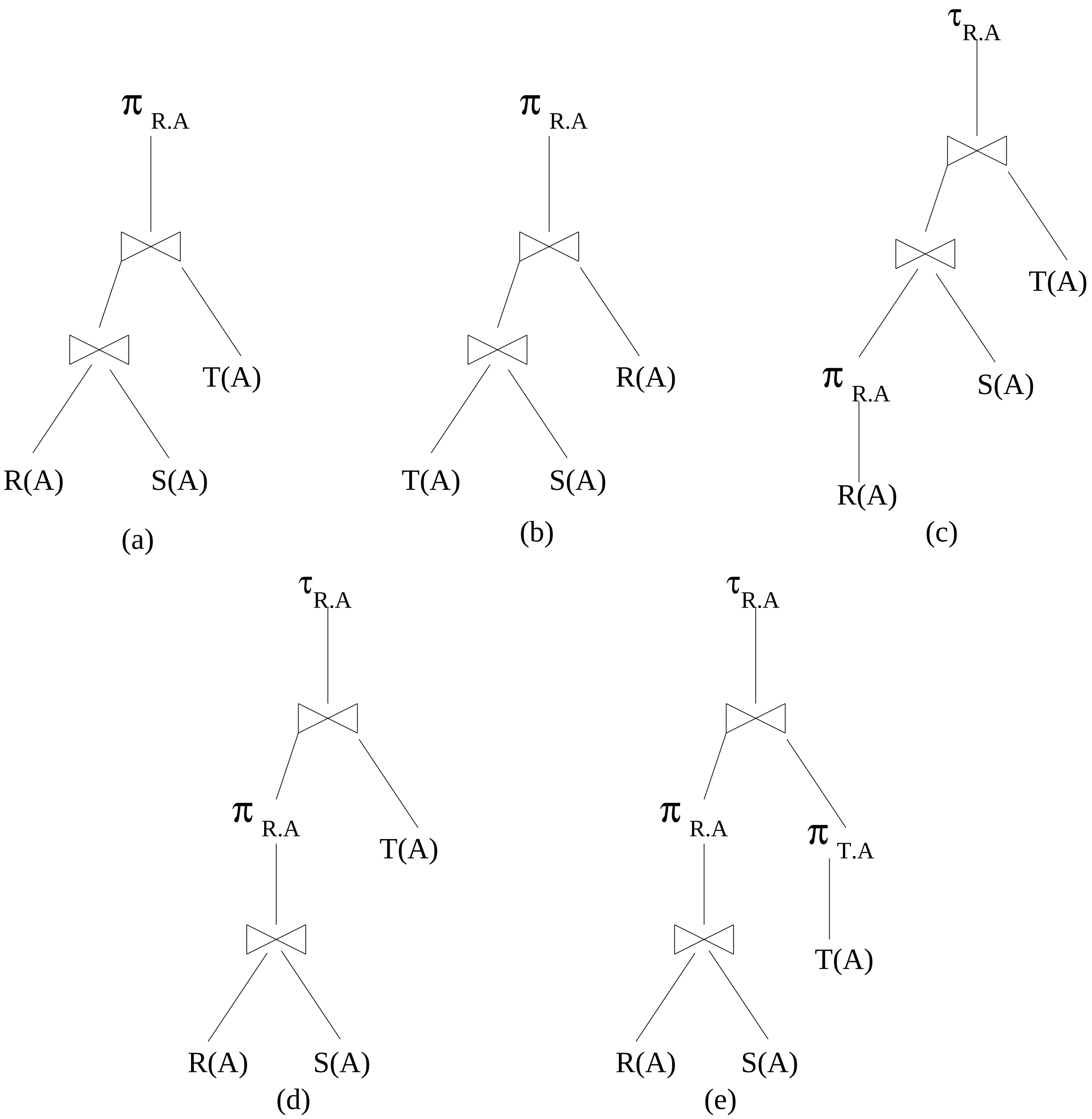

Figure 1: Logical plans

A1 Data from each table is read through a table scan which is represented as *Scan*. We do not consider index scans.

A2 Each join is implemented using a tuple nested loop join, or *TNLJ*. TNLJ is the only physical join operator available.

A3 A duplicate-eliminating projection is always implemented using the operator π. Note that π is a physical operator here.

A4 A duplicate-preserving projection is always implemented using the operator τ. Note that τ is a physical operator here.

The records in the tables are as follows.

1. $R(A)$ contains $\{1, 1, 2, 2, 3, 3\}$.

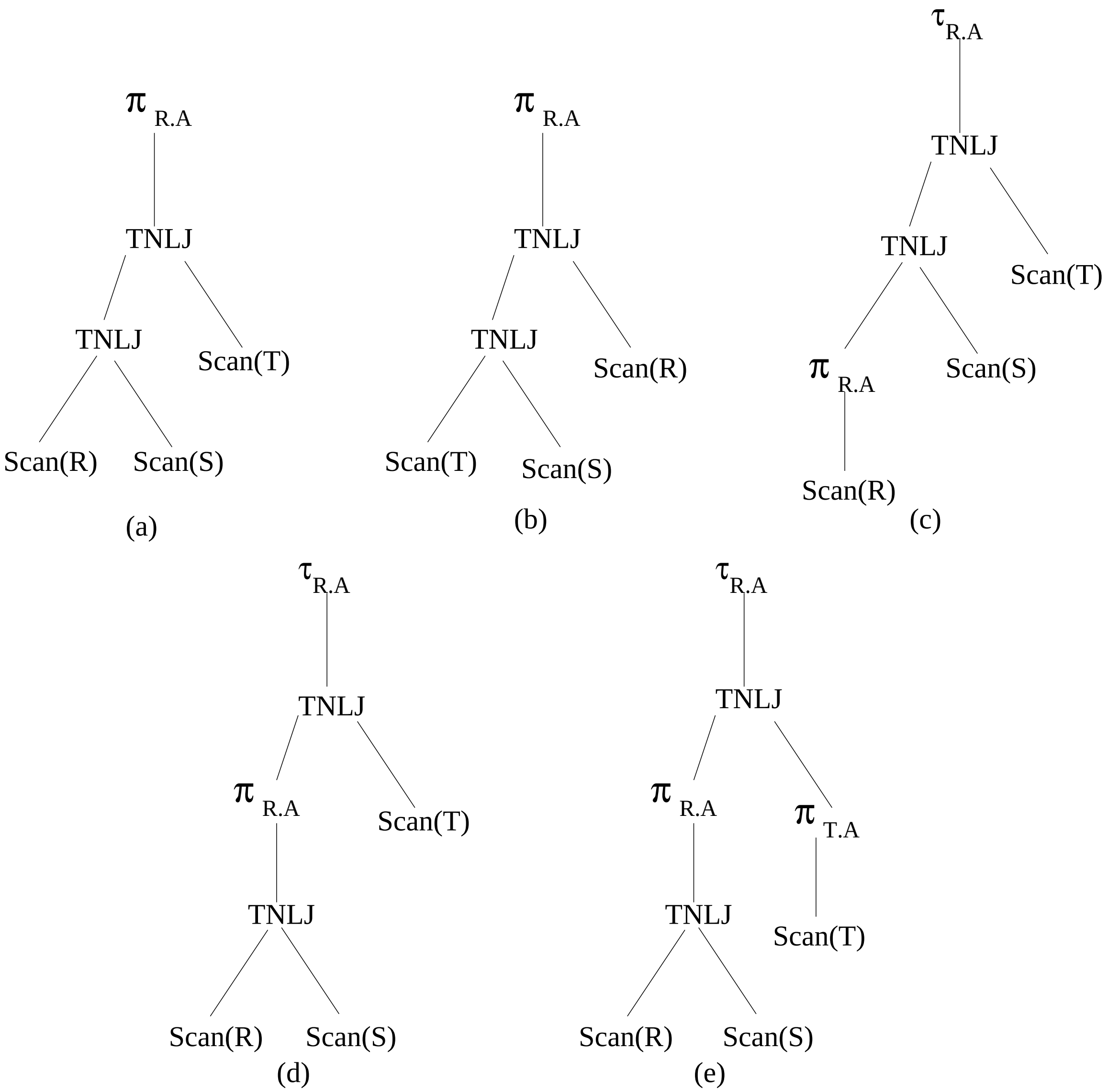

Figure 2: Physical plans

2. $S(A)$ contains $\{1, 2\}$.
3. $T(A)$ contains $\{1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4\}$.

Answer the following five questions based on the above information. Give explanations for partial credit.

1. How many getNext() calls are made by the plan in Figure 2(a)?
2. How many getNext() calls are made by the plan in Figure 2(b)?
3. How many getNext() calls are made by the plan in Figure 2(c)?
4. How many getNext() calls are made by the plan in Figure 2(d)?
5. How many getNext() calls are made by the plan in Figure 2(e)?

Question 3 **Points 15**

Given the assumptions A1 to A4 in Question 2 about physical plans, find the number of possible physical plans for the SQL query in Question 1. Assume that R, S, and T contain duplicate tuples. Give explanations for partial credit.

Question 4 **Points 15**

Given the assumptions A1 to A4 in Question 2 about physical plans, find the physical plan that does the minimum number of getNext() calls for the SQL query in Question 1 and the records in tables $R(A)$, $S(A)$, and $T(A)$ given in Question 2. Give explanations for partial credit.

Hint: The physical plan that does the minimum number of getNext() calls may not be one of the plans in Figure 2.

Question 5 **Points 15**

Recall the problem that we worked out in class where:

- We have two select operators σ_1 and σ_2.
- For each record processed by σ_1, there is a *cost* of c_1. Similarly, for each record processed by σ_2, there is a cost of c_2.
- α_1 and α_2 are the respective *selectivities* of σ_1 and σ_2. Let n be a positive integer. If σ_1 processes n records, then it will let $n\alpha_1$ records pass through, and drop the remaining $n(1-\alpha_1)$ records. Similarly, if σ_2 processes n records, then it will let $n\alpha_2$ records pass through, and drop the remaining $n(1-\alpha_2)$ records.

In class we worked out the condition that $c_1, \alpha_1, c_2, \alpha_2$ should satisfy so that it is better to run σ_1 before σ_2 in a plan. (That is, we worked out the condition that makes the plan in Figure 3(a) better than the plan in Figure 3(b).)

In this question, we are given N select operators σ_1-σ_N with respective cost and selectivity values $c_1, \alpha_1, c_2, \alpha_2, \ldots, c_N, \alpha_N$. We can run these operators in any of the possible $N!$ (N factorial) permutations. Give the condition that makes it best to run these operators in the order σ_1 first, then σ_2, then σ_3, and so on, and finally σ_N. (That is, find the condition that makes the plan in Figure 3(c) better than all other possible plans for the N select operators.)

Hint: The answer to this question is a very elegant condition. Work out on paper with 2 select operators, and then with 3 select operators to see whether you can spot an interesting pattern.

Question 6 **Points 5**

Consider a 3.5 inch disk with 2 magnetic surfaces with 64 tracks per surface, rotating at 3600 rpm. It has a usable capacity of 2 megabytes (2×2^{20} bytes). Assume 20% of each track is used as overhead (gaps). Also, assume that the usable capacity is equally distributed among the tracks.

a. What is the burst bandwidth this disk can support?

b. What is the sustained bandwidth this disk can support?

c. What is the average rotational latency?

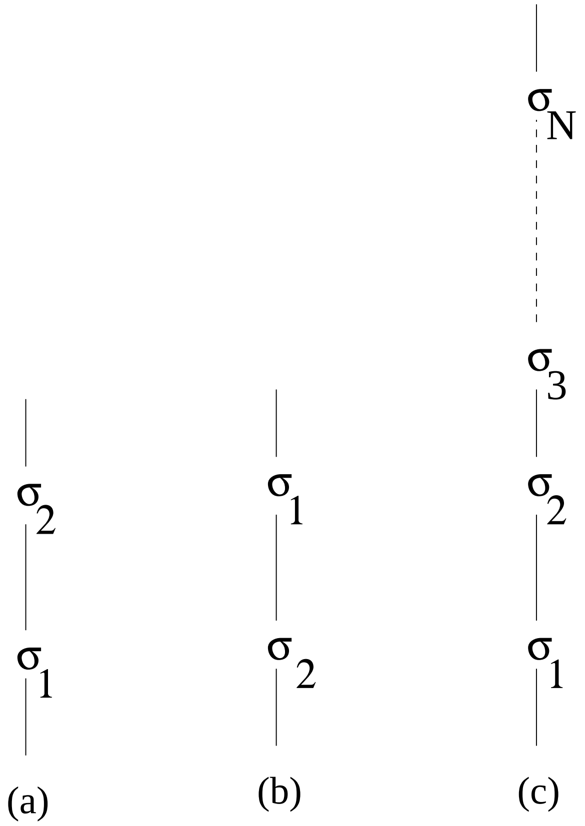

Figure 3: Plan with select operators

d. Assuming the average seek time is 16 ms, what is the average time to fetch a 2-kilobyte (2×2^{10} bytes) sector?

Question 7 **Points 10**

Consider a disk with the following properties:

- There are four platters providing eight surfaces.
- There are $2^{13} = 8192$ tracks per surface.
- There are (on average) $2^8 = 256$ sectors per track.
- There are $2^9 = 512$ bytes per sector.
- The disk rotates at 3840 rpm.
- The block size is $2^{12} = 4096$ bytes.
- Assume 10% of each track is used as overhead.
- The time it takes the head to move n tracks is 1 + n/500 milliseconds.

Suppose that we know that the last I/O request accessed cylinder 3000. (Cylinders are numbered sequentially: $1, 2, \ldots, 8192$.)

a. What is the expected (average) number of cylinders that will be traveled due to the very next I/O request to this disk?

b. What is the expected block access time for the next I/O, again given that the head is on cylinder 3000 initially?