CS245 Database System Principles - Winter 2001

- Due in class on Thursday, Jan 25.
- State all assumptions.
- Subscribe to **cs245-win01** to receive clarifications and changes.
- Email questions to **cs245ta-win01@lists.stanford.edu**

Assignment 2

A relational database system holds three relations; *C* (companies), *P* (products) and *M* (models) with the following characteristics:

Relation *C* (company):

- Tuples are stored as fixed length, fixed format records, length *1200 bytes*.
- There are *10,000 C* tuples.
- Tuples contain key attribute *C.N* (company number), length *20 bytes*; other fields and record header make up rest.
- There is an index on attribute *C.N*.

Relation *P* (product):

- Tuples are stored as fixed length, fixed format records, length *300 bytes*.
- There are *30,000 P* tuples.
- Tuples contain attribute *P.N* (the company number who makes the product), length *20 bytes*; other fields and record header make up rest.
- Tuples also contain attribute *P.I* (product identifier), length *20 bytes*.
- There is an index on attribute *P.N*.

Relation *M* (model):

- Tuples are stored as fixed length, fixed format records, length 100 bytes.
- There are *150,000 M* tuples.
- Tuples contain attribute *M.I* (the identifier of the product involved), length *20 bytes*; other fields and record header make up rest.
- There is an index on attribute *M.I*.

While the number of products associated with each company varies, for evaluation purposes we may assume that each company has *3* products, and each product has *5* model records associated with it.

The records are to be placed in a collection of *4 Kilobytes* (*4096 bytes*) disk blocks that have been reserved to exclusively hold *C*, *P*, or *M* records, or combinations of those records. (That is, there are no other types of records in the blocks we are discussing in this problem.) Each block uses *50 bytes* for its header; records are not spanned.)

Three disk organization strategies are being considered:

1. **Sequential:**
   - All the company (*C*) records are placed sequentially (ordered by company number) in one subset of blocks.
   - Product (*P*) records are separate in another set of blocks, Products are not ordered by company number.
   - Finally, model (*M*) records are in a third set of blocks, not ordered by product identifier.

2. **Clustered:**
   - For each company (*C*) record, the *3* products for that company (*C.N* = *P.N*) reside in the same block.
   - Similarly, the *15* model records for those companies are in the same block.
   - The company records are not sequenced in any way.

3. **Random:**
   - The records are placed as follows, without regard to *C.N*, *P.N*, *P.I*, *M.I* values, as follows: Each block contains one random *C* record, *3* random *P* records, and *15* random *M* records.

We are also told there are four types of queries that constitute the vast majority of the workload:

1. **Probe**: Given a company number, get the company record.
2. **Ordered scan**: For all companies, in increasing company number order, get each company record.
3. **Plain scan:** For all companies, in any order, get all company records.
4. **Join:** For a given company number *C.N*, get the company record followed by all its model records. (That is, get all transaction records with *M.I* = *P.I*, for any account with *P.N* = *C.N*)

Problem 1

For each of the storage strategies, compute how many total disk blocks are needed for holding relations *C*, *P* and *M*. Briefly explain your answers. Display your final results as explained below.

Problem 2

For each query type and for each storage strategy, estimate the number of disk blocks that must be transferred from disk to execute the query. Briefly explain each answer. Display your final results as explained below.

Assume you only have one buffer page (4 Kilobytes) in memory; thus, each time you need a block it counts as one IO (unless the request is for exactly the same block you already have in the buffer). You also have enough memory to hold a single working copy of a *C* record, of an *P* record, and of a *M* record. So, for example, to do a *C*, *P* join, you first read in a block containing a *C* record, and copy the record to the working *C* area. Then you read in the block containing a *P* record into your 4 Kilobytes buffer, and join the two records.

Also, ignore any IOs due to index accesses. That is, you may assume the indexes reside in memory.

Displaying Final Answers:
In Problems 1 and 2 you are to compute a total of 15 values. Please display them in a table like this:

	Storage cost (blocks)	IOs for Probe	IOs for Ordered Scan	IOs for Plain Scan	IOs for Join
Sequential					
Clustered					
Random					

(You also need to explain the derivation of the numbers, this is just a summary.)

Hint: Do not concern yourself with events that are very unlikely. For example, say you want to get the three product records for a particular company, and that these records are randomly placed on a large number of disk blocks. There is some chance that the account records you want happen to be in the same block (just by luck). However, for this problem you may assume this probability is negligible and it will take you exactly three IOs to get those three records.