CSE 530 – Midterm Exam

Name:

(Print ***CLEARLY***)

Question	Points Possible	Points Earned
1	30	
2	15	
3	15	
4	15	
5	15	
Total	90	

Question 1 – Heap Files

The assignments that we have worked on so far have used fixed length tuples, meaning every tuple is the same length. It is also possible to use variable length tuples with a heap file. As an example, consider that for our database, every string is 129 bytes. So the word "hi" would have one byte for the length, two bytes for the actual characters, and 126 bytes of empty space. Under a variable length record setup, the word "hi" would take up 3 bytes: 1 byte for the length and two bytes for the characters.

These changes necessitate the need to change the structure of our heap file as well. In particular, the header will no longer be used to track which slots are open, rather it will track two values for each tuple: the offset (the location in the file the tuple can be found, in bytes) and the length (also in bytes). You may assume that each value is represented by a four byte integer.

Use this information to answer the following questions:

1.) In general, will this lead to a more efficient use of space? How will this affect the number of tuples in a heap file? Describe a situation where this implementation of a heap file would be a less efficient use of space the heap file structure we used in the assignments. (10 points)

2.) How will this change in heap file structure affect adding and deleting tuples? Describe what changes will need to be made to these algorithms. You do not need to write code for this part, but you may include some code if it will help you answer the question. (10 points)

3.) How will this change in heap file structure affect searching for tuples with a certain value (i.e. a select relational operation)? Describe what changes will need to be made to this algorithm. Will searching for variable length tuples be more or less efficient than the fixed length implementation? Explain how you know. (10 points)

Question 2 – SQL Queries

You are given the following tables. A * next to a column name indicates that it is the primary key:

Customer		
cid*	**Name**	**Address**
1	Doug Shook	123 sycamore lane
2	Trent Reznor	9 nail street

Products	
Itemid*	**Description**
1	Fidget Spinner
2	Salad Spinner

Orders			
oid*	**cid**	**Itemid**	**Quantity**
1	1	2	7
2	1	1	6
3	2	2	9

Write SQL queries to answer the following questions:

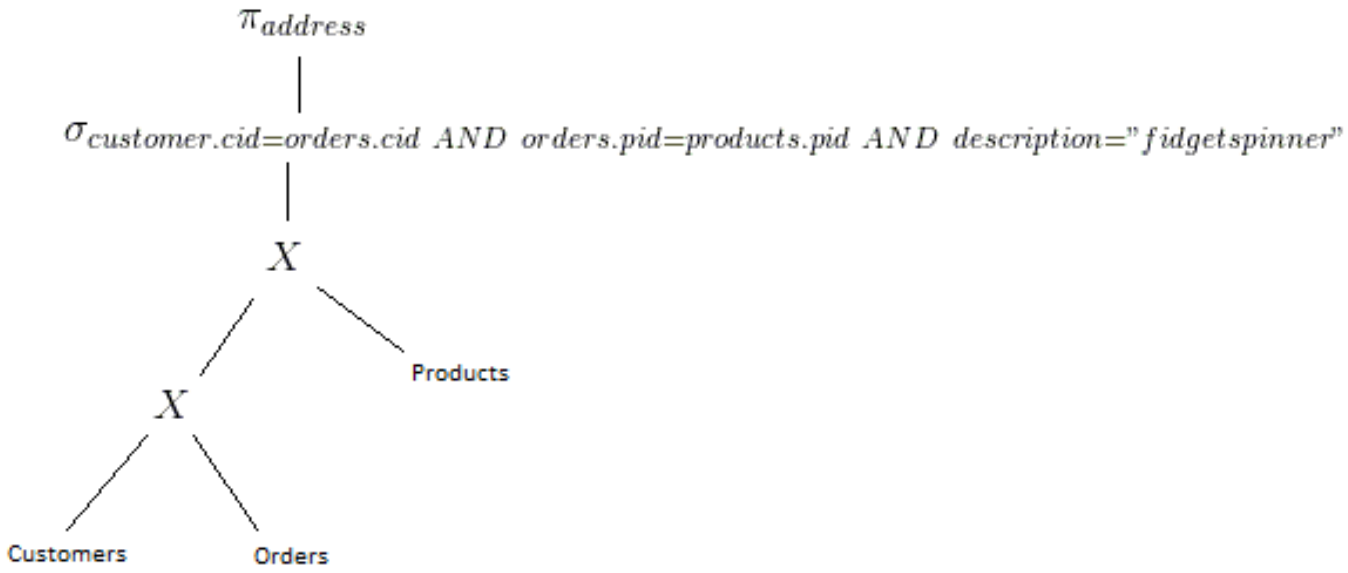
1.) List the descriptions of all items that have been ordered by Doug Shook. (5 points)

2.) List the descriptions of each item in the database along with the total quantity of each item that has been ordered. (5 points)

3.) List the number of orders that have been placed for fidget spinners (note: this is not the same as the total quantity of fidget spinners that have been ordered!). (5 points)

Question 3 – Relational Operations

The following query tree is based on the tables presented in question 2. Construct the optimized version of this query tree. You only need to show the final optimized tree. (10 points)



Write the SQL query that is represented by this query tree. (5 points)

Question 4 – B+ Trees

Construct a B+ tree that inserts the following values in order:

7, 12, 18, 3, 9, 20, 4, 1, 8, 11

Each internal node and leaf node can store up to two values, meaning that internal nodes can have up to three children. (15 points)

Question 5 – Design

Create an ER diagram for a database that tracks playlists. A playlist contains one or more songs. A song can have one or more artist associated with it. A song can appear at most once on a given playlist. (15 points)

Make sure your design includes:

- Entities
- At least two attributes per entity
- Relationships (including participation)