CPS196.03 Information Management and Mining - Spring 2009
Assignment 3

- Due date: Tuesday, March 24, 2009, 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.
- Total points = 100.
- **In all questions, assume that storing an integer or a floating point number takes 4 bytes, both in memory and on disk.**
- Most questions in this assignment are based on the fact table on Slide number 27 in Notes 8. (Notes 8 is available on the Course schedule and notes page on the course Web site.) This fact table has three dimension attributes and 15 records only. Input datasets provided in Programming Project II will have more dimension attributes and many more records.

Question 1 **Points 10**

Consider the following fact table in a data warehouse. The table contains 3 records. Each record has 10 dimension attributes and 1 measure attribute.

A	B	C	D	E	F	G	H	I	J	Measure
a1	d2	d3	d4	d5	d6	d7	d8	d9	d10	1
d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	1
d1	d2	c3	d4	d5	d6	d7	d8	d9	d10	1

The following questions are based on the Cube Aggregates Lattice for the above fact table. Explain your answers.

1. Compute the total number of records over all the nodes in the Lattice. For example, the node for A,B,C,D,E,F,G,H,I,J in the Lattice contains the 3 records shown above. The node for *all* contains 1 record. You have to add up the number of records over all the nodes in the Lattice.
2. How many nodes in the Lattice have two or more records?

Question 2 **Points 10**

Consider the fact table on Slide number 27 in Notes 8. Attributes D1, D2, and D3 are dimension attributes. M is the measure attribute. The following questions are based on the Cube Aggregates Lattice for this fact table. The aggregate of interest is SUM. Explain your answers.

1. Write down the aggregate computed for node D1,D2.
2. Write down the aggregate computed for node D2.
3. Which nodes in the Lattice have more than 5 records?
4. Compute the total number of records over all the nodes in the Lattice.
5. Compute the total amount of memory (in number of bytes) to store the full cube for this fact table.

Question 3 **Points 10**

Consider the fact table on Slide number 27 in Notes 8. The aggregate of interest is SUM. Dimension attribute D1 takes one of 8 distinct values: A1, A2, A3, A6, A7, A10, A13, and A15. Dimension attribute D2 takes one of 4 distinct values: B1, B2, B7, and B11. Dimension attribute D3 takes one of two distinct values: C1 and C12.

We want to store this data as a multidimensional array. Each chunk of this array contains four distinct values of D1, two distinct values of D2, and one distinct value of D3. Assume that:

- A1, A2, A3, and A6 go to the same chunk
- A7, A10, A13, and A15 go to the same chunk
- B1 and B2 go to the same chunk
- B7 and B11 go to the same chunk
- C1 and C12 go to different chunks

1. How many chunks are there in the multidimensional array? Let N be the total number of chunks.
2. If the dimension order is D1D2D3. That is, we number chunks along D1 first, then along D2, and finally along D3. For this dimension order, give the chunk number for each of the 15 records shown on Slide number 27 in Notes 8. Chunk numbers start at 1 and go up to N. (See Figure 1 in the Zhao, Deshpande, and Naughton paper for an example of chunk numbering.)
3. We want to store each chunk in a separate file. The file for chunk i is named i.TXT. Give an algorithm that does one scan over the fact table, and outputs each chunk in its corresponding file.
4. How is your algorithm different from the algorithm given in Section 2.3 (Loading Arrays from Tables) of the Zhao, Deshpande, and Naughton paper?

Question 4 **Points 10**

Consider the fact table on Slide number 27 in Notes 8. This question assumes that you have generated chunks from this fact table as per Question 3 and based on the dimension order D1D2D3. You have stored each chunk in a separate file.

1. Give an algorithm that computes the cube by doing one scan through all files in the order 1.TXT, 2.TXT, up to N.TXT. One scan means that you are allowed to read each file only once. Recall that N is the number of chunks.
2. Estimate the amount of memory that your above algorithm uses.
3. Suppose you are allowed the flexibility to read the chunks in any order. However, you can scan through the files only once, i.e., you can read each file only once. In what order will you read the files to minimize the amount of memory that your algorithm uses to compute the cube. Explain your answer.

Question 5 **Points 20**

Consider the fact table on Slide number 27 in Notes 8. This question assumes that you have generated chunks from this fact table as per Question 3 and based on the dimension order D1D2D3. You have stored each chunk in a separate file.

1. Suppose we want to compute the D1D2 aggregate by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute the D1D2 aggregate?
2. Suppose we want to compute the D2D3 aggregate by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute the D2D3 aggregate?
3. Suppose we want to compute the D1D3 aggregate by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute the D1D3 aggregate?
4. Suppose we want to compute both the D1D2 and the D2D3 aggregates together by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute both aggregates together?
5. Suppose we want to compute both the D1D2 and the D1 aggregates together by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute both aggregates together?

Question 6 **Points 10**

Consider the fact table on Slide number 27 in Notes 8. The aggregate of interest is SUM. We want to store this data as a multidimensional array. Each chunk of this array contains two distinct values of D1, two distinct values of D2, and two distinct values of D3. Assume that:

- A1 and A2 go to the same chunk
- A3 and A6 go to the same chunk
- A7 and A10 go to the same chunk
- A13 and A15 go to the same chunk
- B1 and B2 go to the same chunk
- B7 and B11 go to the same chunk

- C1 and C12 go to the same chunk

1. How many chunks are there in the multidimensional array? Let N be the total number of chunks.
2. If the dimension order is D1D2D3. That is, we number chunks along D1 first, then along D2, and finally along D3. For this dimension order, give the chunk number for each of the 15 records shown on Slide number 27 in Notes 8. Chunk numbers start at 1 and go up to N.
3. If the dimension order is D3D2D1. That is, we number chunks along D3 first, then along D2, and finally along D1. For this dimension order, give the chunk number for each of the 15 records shown on Slide number 27 in Notes 8. Chunk numbers start at 1 and go up to N.

Question 7 **Points 10**

Consider the fact table on Slide number 27 in Notes 8. This question is based on the algorithm that you gave in Question 4.1 to compute the cube by doing one scan through all files in the order 1.TXT, 2.TXT, up to N.TXT.

1. This question assumes that you have generated chunks from this fact table as per Question 6 and based on the dimension order D1D2D3. You have stored each chunk in a separate file. Estimate the amount of memory that your algorithm from Question 4.1 uses to compute the cube.
2. This question assumes that you have generated chunks from this fact table as per Question 6 and based on the dimension order D3D2D1. You have stored each chunk in a separate file. Estimate the amount of memory that your algorithm from Question 4.1 uses to compute the cube.

Question 8 **Points 20**

Consider the fact table on Slide number 27 in Notes 8. This question assumes that you have generated chunks from this fact table as per Question 6 and based on the dimension order D3D2D1. You have stored each chunk in a separate file.

1. Suppose we want to compute the D1D2 aggregate by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute the D1D2 aggregate?
2. Suppose we want to compute the D2D3 aggregate by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute the D2D3 aggregate?
3. Suppose we want to compute the D1D3 aggregate by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute the D1D3 aggregate?
4. Suppose we want to compute both the D1D2 and the D2D3 aggregates together by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute both aggregates together?
5. Suppose we want to compute both the D1D2 and the D1 aggregates together by reading each file at most once. Which files should we read, and in what order so that we need the minimum amount of memory to compute both aggregates together?