

Computer Science COMP3420 in 2010 – Assignment One

Due: 5pm Thursday, April 22
Late Penalty: 20% per day

*No programming is needed for this assignment. Before you hand in your assignment, make sure you have put **your name, student ID and your tut/lab group** into the front page of your assignment. Put your work into the COMP3420 box on the ground floor in CSIT Building. The total mark for this assignment are 20 points (20% of the final marks). In addition, there are extra 3 points (3% of the final marks) for bonus questions. Notice that for all calculation questions, show all your major working steps. In other words, if you just write down the final result of each question, you would not receive any marks!*

Question 1 (2/20) What are the major differences between an enterprise data warehouse and an operational database? Can you describe the main components of a data warehouse and their functions?

Question 2 (3/20) Suppose a group of 21 *sensory reading* values has been sorted as follows.

4, 6, 8, 9, 9, 11, 14, 14, 16, 21, 22, 30, 35, 40, 45, 50, 51, 59, 70, 92, 115.

Partition the data into **four** bins using the following four methods.

- (a) equal-width partitioning
- (b) equal-depth partitioning
- (c) smooth the data in bins by bin means
- (d) smooth the data in bins by bin boundaries

Question 3 (4/20) Suppose a hospital tested the age and body fat data for 18 randomly selected adults with the following results:

age	21	23	23	27	39	44	47	48	51
%fat	8.5	28.5	9.8	19.8	35.4	27.9	26.4	27.2	29.2

age	52	54	54	57	57	57	58	59	61
%fat	34.6	43.5	29.8	38.4	32.2	34.1	31.9	41.9	34.7

- (a) Calculate the mean, median, and standard deviation of *age* and *%fat*.
- (b) Draw the boxplots for *age* and *%fat*.
- (c) Normalise the two variables *age* and *%fat* using *min-max normalization* where $min = 0$ and $max = 1$.
- (d) Calculate the *correlation coefficient* (Pearson's product moment coefficient). Are these two variables positively or negatively correlated?

Question 4 (4/20)

Suppose that a data warehouse consists of the four dimensions *hospital*, *time*, *doctor*, and *patient*, and the two measures *count* and *charge*, where *count* is the number of visits of a patients to a doctor and *charge* is the fee that a doctor charges a patient for a visit.

- (a) Draw a *star* and a *snowflake* schema diagrams for the above data warehouse.
- (b) Starting with the base cuboid [**hospital**, **day**, **doctor**, **patient**], what specific *OLAP operations* (e.g. roll-up) should be performed in order to list the total fee collected by each doctor in 2009?
- (c) Starting with the base cuboid [**hospital**, **day**, **doctor**, **patient**], what specific *OLAP operations* should be performed in order to list the total fee collected by Dr John Rudd in Canberra hospital from 2002 to 2009?
- (d) To obtain the same list as (c), write an SQL query assuming the data is stored in a relational database with the schema *fee(day, month, year, doctor, hospital, patient, count, charge)*.

Question 5 (3/20)

Assume that a data warehouse contains 100 dimensions, each with ten levels of granularity.

- (a) Users are mainly interested in 10 particular dimensions, each having four frequently accessed levels for rolling up and drilling down. Design a data cube structure to efficiently support this preference, and justify your design.
- (b) A user wants to drill through the data cube, down to the raw data for one or two particular dimensions. How would you support this feature?

Question 6 (4/20) There are several cube computation methods including *multiway array computation*, *bottom-up computation*, and *star-cubing*. Briefly describe these three methods (i.e., use one or two lines to outline the key points), and compare their feasibility and performance under the following conditions.

- (a) Computing a dense full cube of low dimensionality (e.g., less than 8 dimensions)
- (b) Computing a sparse iceberg cube of high dimensionality (e.g., over 100 dimensions)

- (c) Computing an iceberg cube of around 10 dimensions with a highly skewed data distribution

Bonus Questions (3 points) Warning: *The following question is designed for people who are capable to do some extra research-related work. Only you have finished all basic questions **correctly** and are willing to challenge yourself, you may proceed the questions.*

Question b1 (3 points) A data warehouse normally turns itself off from customer queries when it proceeds the data maintenance to keep the data within the data warehouse consistent. However, for a data warehouse owned by a global company, it is desirable that the data warehouse should be available to its customers for 24 hours and 7 days per week by responding their queries. Assume that all tables in the data warehouse are the relational tables and there is no limitation on the storage space in the data warehouse. To design a data warehouse that is always able to handle customer queries, and at the same time can carry out its data maintenance, (a) can you briefly propose a solution for such a data warehouse design? (b) If the maintenance time is upper bounded by a given value T , (i) can you propose an algorithm for a materialized view maintenance in the data warehouse under the maintenance time constraint? (*Hints: you may adopt a timestamp for each tuple. The queries are time-sensitive queries.*).