Question 1: Relational Model ..... [245 points]

(i) **[20 points] Toolchain:**
Provide brief definitions of these tools/services.

- Compiler
- Build System
- Continuous Integration
- Memory Error Detector

(ii) **[10 points] External Sort:**
Distinguish between external and internal sort algorithms. How can the former algorithm leverage the latter?

(iii) **[10 points] External Sort:**
If there are n values to be sorted and only m values fit in memory. How many runs are created initially in an external sort algorithm?

(iv) **[10 points] Iterative 2-way Merge:**
Consider the task of sorting k runs using an external sort algorithm. How many iterations are needed to sort these runs using an iterative 2-way merge algorithm? Give an asymptotic bound along with a justification.

(v) **[10 points] Iterative 2-way Merge:**
What is the total time to run an iterative 2-way merge algorithm. Give an asymptotic bound in terms of n and k along with a justification.

(vi) **[10 points] k-way Merge:**
What is the total time to run a k-way merge algorithm: (1) without a heap and (2) with a heap. Give asymptotic bounds in terms of n and k along with a justification.

(vii) **[10 points] k-way Merge:**
Consider a heap with k elements. What is the time taken to find the maximum element?

(viii) **[10 points] k-way Merge:**
If k values do not fit in memory (only contains m slots and $m < k$), how would you sort the data?

(ix) **[15 points] Flat Files vs Relational DBMSs:**
List three limitations of using flat files as opposed to a relational DBMS.

(x) **[10 points] Data Integrity:**
Provide a brief definition for data integrity. Give an example of ensuring data integrity in a database application.

(xi) **[10 points] Relational DBMSs:**
Explain the tight coupling of logical and physical layers in early, non-relational DBMSs. How does a relational DBMS circumvent this limitation?

(xii) **[10 points] Non-Relational Data Models:**
Give an example of a non-relational data model.

Question 1 continues...

(xiii) **[10 points] Relation:**
Define a relation. Is it ordered or unordered? Can it contain duplicates?

(xiv) **[10 points] Primary vs Foreign Key:**
Distinguish between primary and foriegn keys with an example application.

(xv) **[10 points] Many-to-Many Relationship:**
How do you capture a many-to-many relationship with the relational model? Give an example.

(xvi) **[10 points] Relational Algebra:**
Define relational algebra. Is it procedural or non-procedural?

(xvii) **[10 points] Relational Operator:**
Define a relational operator. What are its inputs and output? Give three examples of relational operators.

(xviii) **[10 points] Predicate:**
Distinguish between conjunction and disjunction of two predicates. Which combination is likely to return more tuples?

(xix) **[20 points] Set vs Bag Semantics:**
Distinguish between set and bag semantics. Which one is used in practice? Why?

(xx) **[10 points] Product vs Join Operators:**
Distinguish between product and join operators. Which operator is used more often in practice?

(xxi) **[10 points] Product vs Join Operators:**
Consider two tables with m and n tuples respectively. Is it possible to evaluate the join operator faster than the product operator? Give asymptotic bounds along with a justification.

(xxii) **[10 points] Set-Oriented Processing:**
Distinguish between tuple- vs set-oriented processing. Which one is faster? Why?