Question 1: Modern OLTP Indexes (Part I) . . . . . [160 points]

(i) **[10 points] T-Tree:**
Distinguish between a B+tree and a T-Tree.

(ii) **[10 points] T-Tree:**
List an advantage and a disadvantage of T-Tree.

(iii) **[10 points] Versioned Latch Coupling:**
Illustrate an advantage of versioned latch coupling using an example.

(iv) **[10 points] Versioned Latch Coupling:**
Illustrate a disadvantage of versioned latch coupling using an example.

(v) **[10 points] Versioned Latch Coupling:**
Illustrate an advantage of versioned latch coupling using an example.

(vi) **[10 points] Versioned Latch Coupling:**
Distinguish between optimistic and pessimistic latch coupling schemes.

(vii) **[10 points] Versioned Latch Coupling:**
Distinguish between Test-and-Set and Compare-and-Swap primitives.

(viii) **[10 points] Versioned Latch Coupling:**
Can you implement Test-and-Set using Compare-and-Swap primitive? Justify your answer.

(ix) **[10 points] Bw-Tree:**
List the key ideas of a Bw-Tree. Distinguish between a B+tree and a Bw-Tree.

(x) **[10 points] Bw-Tree:**
How do delta updates improve caching behavior?

(xi) **[10 points] Bw-Tree:**
Define these terms:

- Garbage Collection
- Delta Updates
- Mapping Table

(xii) **[10 points] Bw-Tree:**
How are conflicting updates to the same node resolved?

(xiii) **[10 points] Bw-Tree:**
Define these terms:

- Node Consolidation
- Node Split
- Node Merge

(xiv) **[10 points] Bw-Tree:**
Distinguish between epoch-based garbage collection and reference counting.

Question 1 continues…

(xv) **[10 points] Bw-Tree:**
Explain how versioning helps increase concurrency.

(xvi) **[10 points] Bw-Tree:**
Explain how BwTree manages to handle complex tree operations with only the compare-and-swap instruction.