1. (20 points) Database Concepts (choose one):
   1.1 Explain the three-layer database architecture.
   1.2 Discuss the six relational integrity constraints.

2. (20 points) Database Design:
   Show how the below company data are modeled into (1) an ER schema and (2) create relational tables in DDL/SQL.
   ‘In a university, a student belongs to a department and may participate in up to two research projects based on different work hours (such as half_time, full_time, etc.). Projects are conducted by a department, and a project must have a minimum of four students.’ (You may design the entities’ attributes by yourself.)

3. (20 points) Query Processing:
   From the database above, a database application issues a query below:
   "Find female students who work for the CS department, and give their SIDs, names, and locations of their department."
   Show (1) the query in DML/SQL, and (2) query plan in the form of a query tree (or step-wise algebra operations).

4. (20 points) Transaction Management (choose one):
   4.1 Transaction Recovery: (1) interpret the transaction log below onto a time chart; (2) show transaction recovery process for Deferred Update algorithm.

**TRANSACTION LOG:**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
<td>□□□□□□□□□</td>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
<tr>
<td>220</td>
<td>79</td>
<td>Null</td>
<td>228</td>
<td>START</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>221</td>
<td>77</td>
<td>119</td>
<td>249</td>
<td>INSERT</td>
<td>COURSE</td>
<td>606</td>
<td>-</td>
<td>-</td>
<td>606, DB, …</td>
</tr>
<tr>
<td>228</td>
<td>79</td>
<td>220</td>
<td>230</td>
<td>INSERT</td>
<td>STUDENT</td>
<td>20072678</td>
<td>-</td>
<td>-</td>
<td>20072678, KIM, F, …</td>
</tr>
<tr>
<td>230</td>
<td>79</td>
<td>228</td>
<td>Null</td>
<td>COMMIT</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>231</td>
<td>89</td>
<td>Null</td>
<td>240</td>
<td>START</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>240</td>
<td>89</td>
<td>231</td>
<td>262</td>
<td>UPDATE</td>
<td>STUDENT</td>
<td>20071398</td>
<td>GPA</td>
<td>3.0</td>
<td>3.5</td>
</tr>
<tr>
<td>249</td>
<td>77</td>
<td>221</td>
<td>260</td>
<td>DELETE</td>
<td>COURSE</td>
<td>600</td>
<td>-</td>
<td>600, OS, …</td>
<td>-</td>
</tr>
<tr>
<td>255</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>checkpoint</td>
</tr>
<tr>
<td>260</td>
<td>77</td>
<td>249</td>
<td>266</td>
<td>UPDATE</td>
<td>COURSE</td>
<td>606</td>
<td>ROOM</td>
<td>L305</td>
<td>L405</td>
</tr>
<tr>
<td>262</td>
<td>89</td>
<td>240</td>
<td>Null</td>
<td>COMMIT</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>266</td>
<td>77</td>
<td>260</td>
<td></td>
<td>UPDATE</td>
<td>COURSE</td>
<td>500</td>
<td>ROOM</td>
<td>L405</td>
<td>L305</td>
</tr>
</tbody>
</table>

!! Crash !!
4.2 Transaction Concurrency: (1) Explain transaction serializability and (2) show a serializable schedule for transaction <Ti> using Two-phase Locking Protocol.

```
<Ti>

BEGIN_TR
  request a read-lock on (A)
  read_lock (A) acquired;
  read (A)
  unlock (A)
  request a write-lock on (B)
  write_lock (B) acquired;
  read_ (B);
  B:=A * B;
  write (B)
  unlock (B);
  (COMMIT)
```

(Hint) Two-phase Locking Protocol: All locking operations (read_lock, write_lock) precede the first unlock operation in the transaction. After releasing a lock, a transaction must never go on to acquire any more locks. (growing/shrinking phase)

5. (20 points) Short-answer questions. (choose five questions only)
   (a) Base data, auxiliary data, metadata
   (b) View relation
   (c) FD
   (d) Normalization
   (e) Primitive algebra operations
   (f) JOIN operation
   (g) Database transaction
   (h) Schema