Task 1 How is the storage managed within a DBMS? Why does a DBMS use an own storage management instead of files and blocks of an operating system?

Task 2 Assigning Records to Pages

In an exemplary DBMS, tuples are stored on pages with a size of 100 Byte.

(a) For the management of meetings with other people, a student uses a DBMS (characterized above). The following table structure is used:

- **Name**:varchar(10)
- **Date**:char(10)
- **Time**:char(5)
- **Venue**:varchar(15)
- **Activity**:varchar(15)
- **Bring along**:varchar(10)

Describe two possible ways to represent a record of this type on a page of a DBMS. Is a record always stored exactly on one page?

(b) Simulate stepwise the insertion of data from the following table:

<table>
<thead>
<tr>
<th>Name</th>
<th>Date</th>
<th>Time</th>
<th>Venue</th>
<th>Activity</th>
<th>Bring along</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kai</td>
<td>16.06.2008</td>
<td>11:00</td>
<td>Layla</td>
<td>Brunch</td>
<td>nichts</td>
</tr>
<tr>
<td>Jana</td>
<td>30.07.2008</td>
<td>15:00</td>
<td>Stadtpark</td>
<td>Skaten</td>
<td>Skates</td>
</tr>
<tr>
<td>Nicole</td>
<td>11.08.2008</td>
<td>18:30</td>
<td>Springbrunnen</td>
<td>Kaffetrinken</td>
<td>Rosen</td>
</tr>
</tbody>
</table>

The following meta information need to be considered:

- Page number
- Previous page
- Succeeding page
- Tuple-Offset
- Tuple-Length
- Attribute-Count
- Attribute-Length
Each meta information needs 1 Bytes to be stored.

(c) Simulate the following updates:

- Last-minute the student detects an embarrassing fault? Skating is not with Jana but with Markus!
- Furthermore, Markus selects a different venue: Einkaufszentrum.

(d) How does the storage of records change considering the Decomposition Storage Model or Partition Attributes Across? Which advantageous regarding processing of operations do the alternative storage model provide?

Task 3 Which aspects of the storage management changes considering main-memory databases?

Good Luck!