Q1 (25). Consider the instance of the Students relation shown in the following figure, sorted by age: For the purposes of this question, assume that these tuples are stored in a sorted file in the order shown; the first tuple is on page 1 the second tuple is also on page 1; and so on. Each page can store up to three data records; so the fourth tuple is on page 2.

<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>login</th>
<th>age</th>
<th>gpa</th>
</tr>
</thead>
<tbody>
<tr>
<td>53831</td>
<td>Madayan</td>
<td>madayan@music</td>
<td>11</td>
<td>1.8</td>
</tr>
<tr>
<td>53832</td>
<td>Guldn</td>
<td>guldn@music</td>
<td>12</td>
<td>2.0</td>
</tr>
<tr>
<td>53866</td>
<td>Jones</td>
<td>jones@cs</td>
<td>18</td>
<td>3.4</td>
</tr>
<tr>
<td>53888</td>
<td>Smith</td>
<td>smith@ee</td>
<td>19</td>
<td>3.2</td>
</tr>
<tr>
<td>53850</td>
<td>Smith</td>
<td>smith@math</td>
<td>19</td>
<td>3.8</td>
</tr>
</tbody>
</table>

Show what the data entries in each of the following indexes contain. If the order of entries is significant, say so and explain why. If such an index cannot be constructed, say so and explain why.

1. An unclustered index on age using Alternative (1).
2. An unclustered index on age using Alternative (2).
3. An unclustered index on age using Alternative (3).
4. A clustered index on age using Alternative (1).
5. A clustered index on age using Alternative (2).
6. A clustered index on age using Alternative (3).
7. An unclustered index on gpa using Alternative (1).
8. An unclustered index on gpa using Alternative (2).
9. An unclustered index on gpa using Alternative (3).
10. A clustered index on gpa using Alternative (1).
11. A clustered index on gpa using Alternative (2).
12. A clustered index on gpa using Alternative (3).

Use notation < A, (B,C) > in your answer for Alternative (2), where A is the search key for the entry and (B,C) is the rid for data entry, with B being the page number of the entry and C being the location on page B of the entry. For Alternative (3), the notation is the same, but with the possibility of additional rid’s listed.
Q2 (25). Suppose that a page can contain at most four data values and that all data values are integers. Using only B+ trees of order 2, give examples of each of the following:

1. A B+ tree whose height changes from 2 to 3 when the value 25 is inserted. Show your structure before and after the insertion.

2. A B+ tree in which the deletion of the value 25 leads to a redistribution. Show your structure before and after the deletion.

3. A B+ tree in which the deletion of the value 25 causes a merge of two nodes but without altering the height of the tree.

4. An ISAM structure with four buckets, none of which has an overflow page. Further, every bucket has space for exactly one more entry. Show your structure before and after inserting two additional values, chosen so that an overflow page is created.

Q3 (25). Consider the following actions taken by transaction T1 on database objects X and Y:

\[ \text{R}(X), \text{W}(X), \text{R}(Y), \text{W}(Y) \]

1. Give an example of another transaction T2 that, if run concurrently to transaction T without some form of concurrency control, could interfere with T1.

2. Explain how the use of Strict 2PL would prevent interference between the two transactions.

3. Strict 2PL is used in many database systems. Give two reasons for its popularity.

Q4 (25). Consider a database with objects X and Y and assume that there are two transactions T1 and T2. Transaction T1 reads objects X and Y and then writes object X. Transaction T2 reads objects X and Y and then writes objects X and Y.

1. Give an example schedule with actions of transactions T1 and T2 on objects X and Y that results in a write-read conflict.

2. Give an example schedule with actions of transactions T1 and T2 on objects X and Y that results in a read-write conflict.

3. Give an example schedule with actions of transactions T1 and T2 on objects X and Y that results in a write-write conflict.

4. For each of the three schedules, show that Strict 2PL disallows the schedule.