Indexes

Storage Hierarchy, ISAM, B-Trees and Hashing
Storage Hierarchy

- Storage hierarchy is important for run-time
  - keep as much as possible in main memory
- Read data from disk:
  - Store things you do not need outside on disk
  - Seek time: time until track is found
  - Latency: Rotation until head starts to write sector
  - Transfer time: time to transfer full sector
Hierarchical Indexes: ISAM and B-Trees

• Data transfer can be very time-expensive → keep it as little as possible
• Use index structures: only needed parts are transferred
• **ISAM**: Index Sequential Access Method
  • Predecessor of B-Trees
  • Main Idea: sort tuples on indexes attribute
  • Similar to thumb index in a book
  • Problem: Maintenance of index is expensive
    → Index pages for index pages (B-Trees)
B-Tree

Properties

• Balanced and sorted
• Degree $i$: between $i$ and $2i$ entries (exception: root)
• Every node with $n$ entries has $n+1$ children (exception: leaves)

Improvement: B+-Tree: reference keys in inner nodes, data in leaf nodes → better run-time
Hashing

• Storing tuples in a defined memory area
• Hash function: mapping tuples to a fixed set of function values
• Optimal hash function: injective and surjective
• Typical hash function \( h(x) = x \mod N \rightarrow \) set of function values thereby \( \{0, \ldots, N-1\} \)
Hashing – advantages and disadvantages

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Few accesses to external storage</td>
<td>• Collision handling necessary</td>
</tr>
<tr>
<td>• Simple implementation</td>
<td>• Pre-allocation of memory area</td>
</tr>
<tr>
<td></td>
<td>• Not dynamic,</td>
</tr>
<tr>
<td></td>
<td>• no range queries</td>
</tr>
</tbody>
</table>
Exam Exercises

• Insert following numbers into an empty B-Tree (degree 2)

  7, 21, 5, 3, 17, 90, 4, 34, 24, 32, 13

• What does degree 2 mean for this tree?
Exam Exercises – Solution (1)

1) Insert 7, 21, 5 and 3 in order:

```
    7
   / | \
  7, 21
 /    \
5, 7, 21
```

2) Insert 17: take median as root

```
    7
   / | \
  3, 5
 /    \
17, 21
```

3) Insert 90, 4 and 34

```
    7
   / | \
  3, 4, 5
 /       \
17, 21, 34, 90
```

4) Insert 24: take the median to root

```
   7, 24
  /   \
3, 4, 5
   /    \
17, 21
 /    \
34, 90
```

5) Insert 32 and 13

```
   7, 24
  /   \
3, 4, 5
   /    \
13, 17, 21
 /       \
32, 34, 90
```

Exercise:
Insert 7, 21, 5, 3, 17, 90, 4, 34, 24, 32, 13
Exam Exercises – Solution (2)

Exercise: What does degree 2 mean for this tree?

• B-Tree has at least 2 and at most $2 \times 2 = 4$ entries for every node (except the root)