1.1 Terminology

Define shortly in your own words the following terms:

- Database System
- Data Model
- Database Schema
- Logical and Physical Schema
- Integrity Constraint
- DDL and DML

1.2 B*-Trees

Let $k=k^*=2$. Given the following B*-Tree:

Perform the listed operations on the B*-Tree and redraw the resulting B*-Tree after each step:

- a) insert 11
- b) insert 52
- c) insert 31
- d) insert 18
- e) insert 25
- f) insert 35
- g) insert 56
- h) insert 29
- i) remove 17
- j) remove 7
- k) remove 29
- l) remove 25
- m) remove 26
- n) remove 18
1.3 System architecture

The 5 layer architecture of a DBMS was introduced in the lecture.

a) Describe the six interfaces and the respective level of abstraction in the architecture by explaining what kind of objects and procedures are used in each one of them.

b) Describe the processing of a simple, fictitious query in the different layers of the DBMS architecture, that retrieves passenger names for a particular flight from an airline company database system. Which operations are performed on which layer? Suppose you are working with a relational DBMS which includes the tables passenger, flight and reservation. The respective indexes are implemented by means of B* trees.

c) Build a second data manipulation operation inserting data into the database which is in conflict with your query in part b) on at least one layer of the DBMS architecture. How can the conflict be solved with transaction management?

d) Discuss the advantages of transaction management spanning over all layers with respect to the scenario described in part c).