HW #6

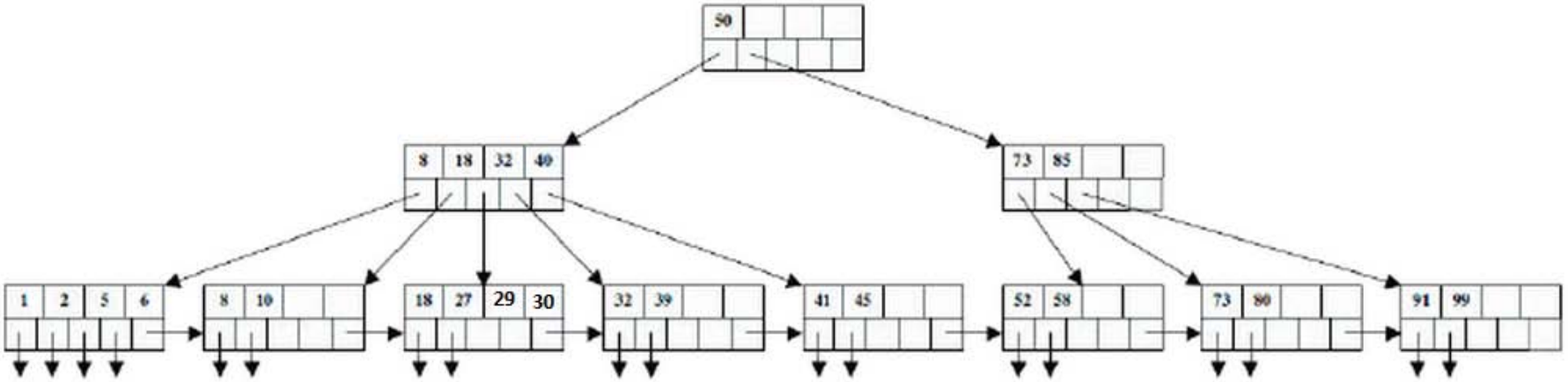
Part A. Consider the B+ tree index shown in the figure. For each problem below, you only need to show the final resulting B+ Tree.

- If you wish, you can just redraw the changed part of the tree. So, e.g., if only the values in 3 of the leaves need to change, note which part of the tree you are redrawing, redraw those 3 leaves, and note that the rest of the tree is unchanged.
- Please use **rotation** correctly when inserting / deleting, as discussed in class.

HW #6 (2)

A1. Show the B+ tree that would result from inserting a data entry with key 28 into the original tree.

A2. Show the B+ tree that would result from deleting a data entry with key 27 from the original tree.



HW #6 (3)

Part B

- Consider a database schema with three relations:

  Students (sid:integer, sname:string, age:integer)

  Enrolled (sid:integer, cid:integer, grade:integer)

  Courses(cid:integer, cname:string, credits:integer)

- The keys are underlined in each relation. Students are identified uniquely by **sid**, and courses by **cid**. Students enroll to take courses, and for each course they obtain a **grade** which is an integer. **sname** is the student name (string), **age** represents the student age and is an integer. **cname** is the course name (string), and **credits** is the number of credits for a particular course (integer).

HW #6 (4)

- Write **relational algebra** expressions for the following queries:

B1. Find the names of students who got grade 10 in some course.

B2. Find the ages of students who take some course with 3 credits.

B3. Find the names of students who take a course named ‘Calculus’.

B4. Find the names of students who obtained grade at least 8 in some course that has less than 4 credits.

B5. Find the names of students who obtained only grades of 10 (implies that they took at least one course).

HW #6 (5)

B6. Find the names of students who took a course with three credits or who obtained grade 10 in some course.

B7. Find the ages of students who attend ‘Calculus’ but never took any 4-credit course (assume there is a course ‘Calculus’ with 3 credits).

B8. Find the names of students who have the lowest grades.

B9. Find the names of students who are enrolled in a **single** course.

B10. Find the grades of students who are enrolled in course(s) with the highest number of credits.

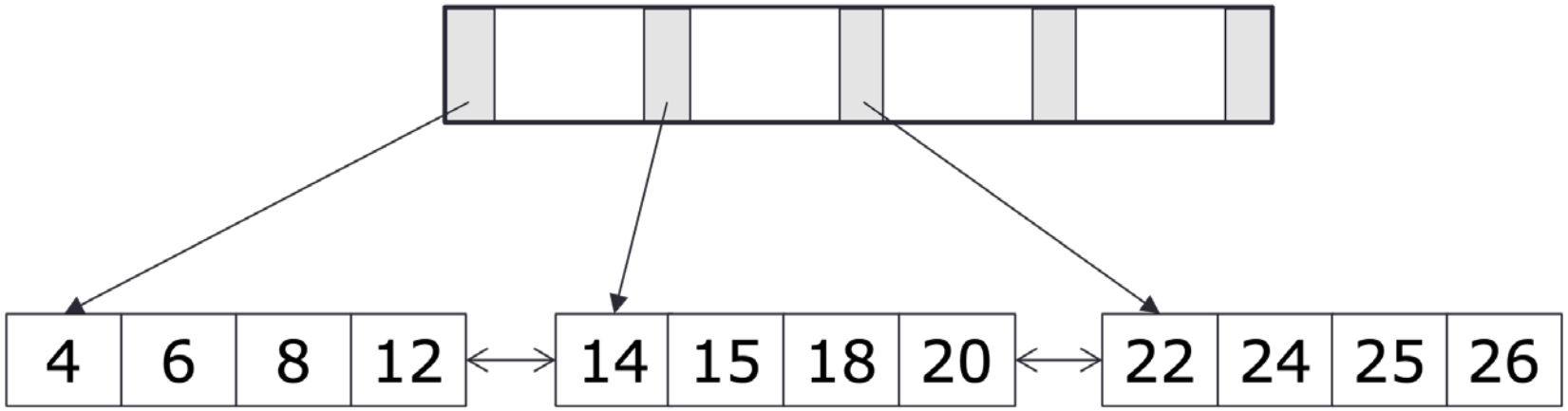
HW #6 (6)

Part C

- Consider a partially filled B+ tree — root value(s) missing.

C1. Fill in the value(s) present in the root of the tree.

C2. There are many sequences of exactly 3 insertions that will cause the root node in the given B+ tree to split. List one such sequence of three insertions, and briefly justify your choice.



HW #6 (7)

Part D

- Consider an empty B+ tree with order d=2, i.e., there are at most 4 keys per node, and at most 5 pointers to children. Bulk load the B+ tree with data entries with odd numbers from 1 to 50 (i.e., 1, 3, 5,…,49) so that each leaf is **at least** half full.

D1. Sketch the final B+ tree after bulk loading.

D2. What is the height of the tree after inserting all the above keys?

D3. What is the minimum number of keys that must be deleted so that the height of the tree decreases by 1? List these keys.

HW #6 (8)

Part E

- Consider an extendible hash structure where buckets can hold up to three records. Initially the structure is empty and global depth is 2.

E1. Sketch the extendible hash structure after the records with the given hash values in Table 1 (in the same order shown) have been inserted. Assume that the directory doubles in size at each overflow.

E2. How many buckets have the same local depth as the global depth of the structure?

Record	Hash Value
a	001110
b	011010
c	011010
d	101001
e	010111
f	101001
g	010000
h	010111
i	111100
j	000110
k	101011
l	100101
m	010000
n	000100

Table 1