Outline

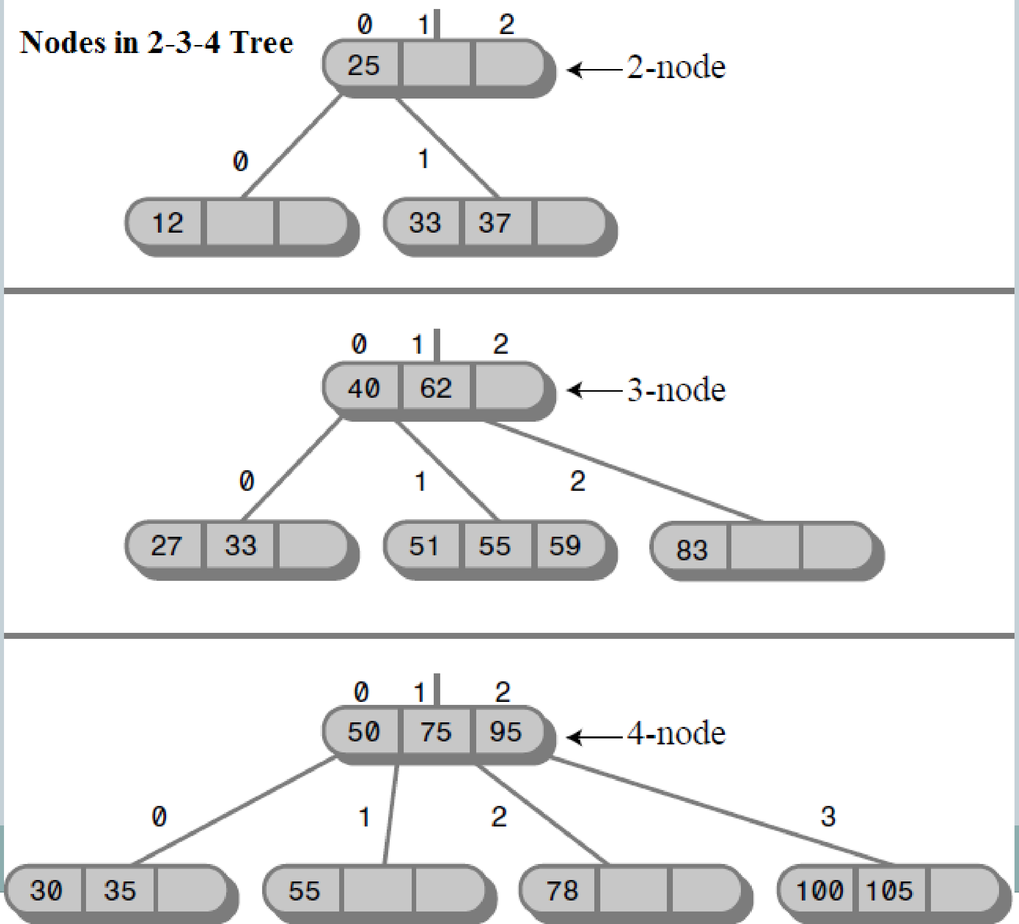
- Binary Trees
- Traversing Binary Tree
- Red-Black Trees
- Red-Black Tree Insertions
- 2-3-4 Trees

Introduction to 2-3-4 Trees

- The 2, 3, and 4 in the name 2-3-4 tree refer to how many links to child nodes can potentially exist in a given node.
- For nonleaf nodes, the following three arrangements are possible:
  - A node with one data item always has two children
  - A node with two data items always has three children
  - A node with three data items always has four children
- Thus, if `L` – number of child links, and `D` – number of data items. `L = D + 1`

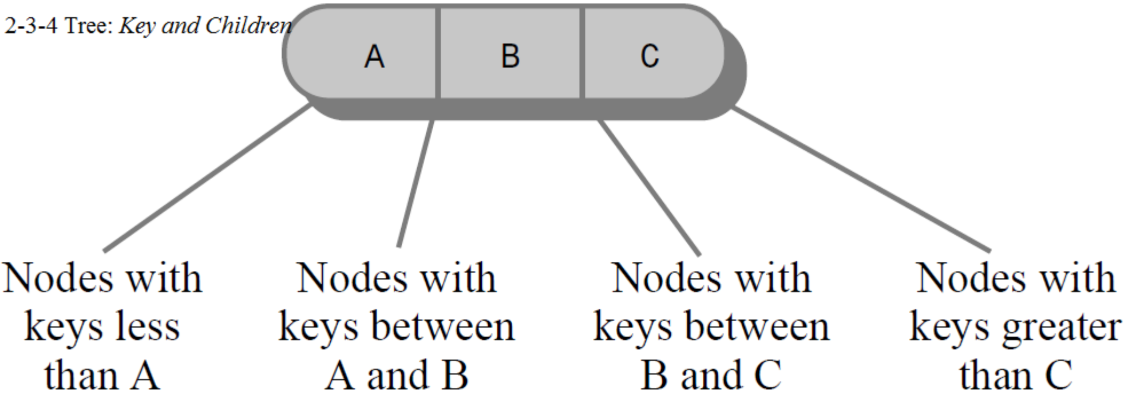
Introduction to 2-3-4 Trees (Cont.)

- Why a 2-3-4 tree is not called a 1-2-3-4 tree? Because a node cannot have only one child.



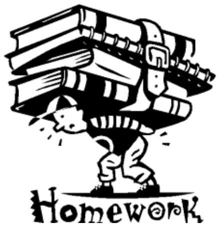
2-3-4 Tree Organization

- Data items in a link were numbered from 0 to 2, and the child links from 0 to 3.
- The data items in a node are arranged in ascending key order from left to right.
- 2-3-4 tree characteristics:
  - All children in the subtree rooted at child 0 have key values less than key 0.
  - All children in the subtree rooted at child 1 have key values greater than key 0 but less than key 1.
  - All children in the subtree rooted at child 2 have key values greater than key 1 but less than key 2.
  - All children in the subtree rooted at child 3 have key values greater than key 2.


2-3-4 Tree: *Key and Children*
A
B
C
Nodes with keys less than A
Nodes with keys between A and B
Nodes with keys between B and C
Nodes with keys greater than C

Searching for a Data Item in 2-3-4 Tree

- Finding a data item with a particular key is similar to the search routine in a binary tree.
- You start at the root, and, unless the search key is found there, select the link that leads to the subtree with the appropriate range of values.



Homework 23 submit to:

fe.assignment@gmail.com

Mon., 7-March-201
@ 15:00

1. Create a 2-3-4 Tree.
2. Create a function for searching a data item in 2-3-4 Tree

Late submission: the score will be minus 10% for every hour

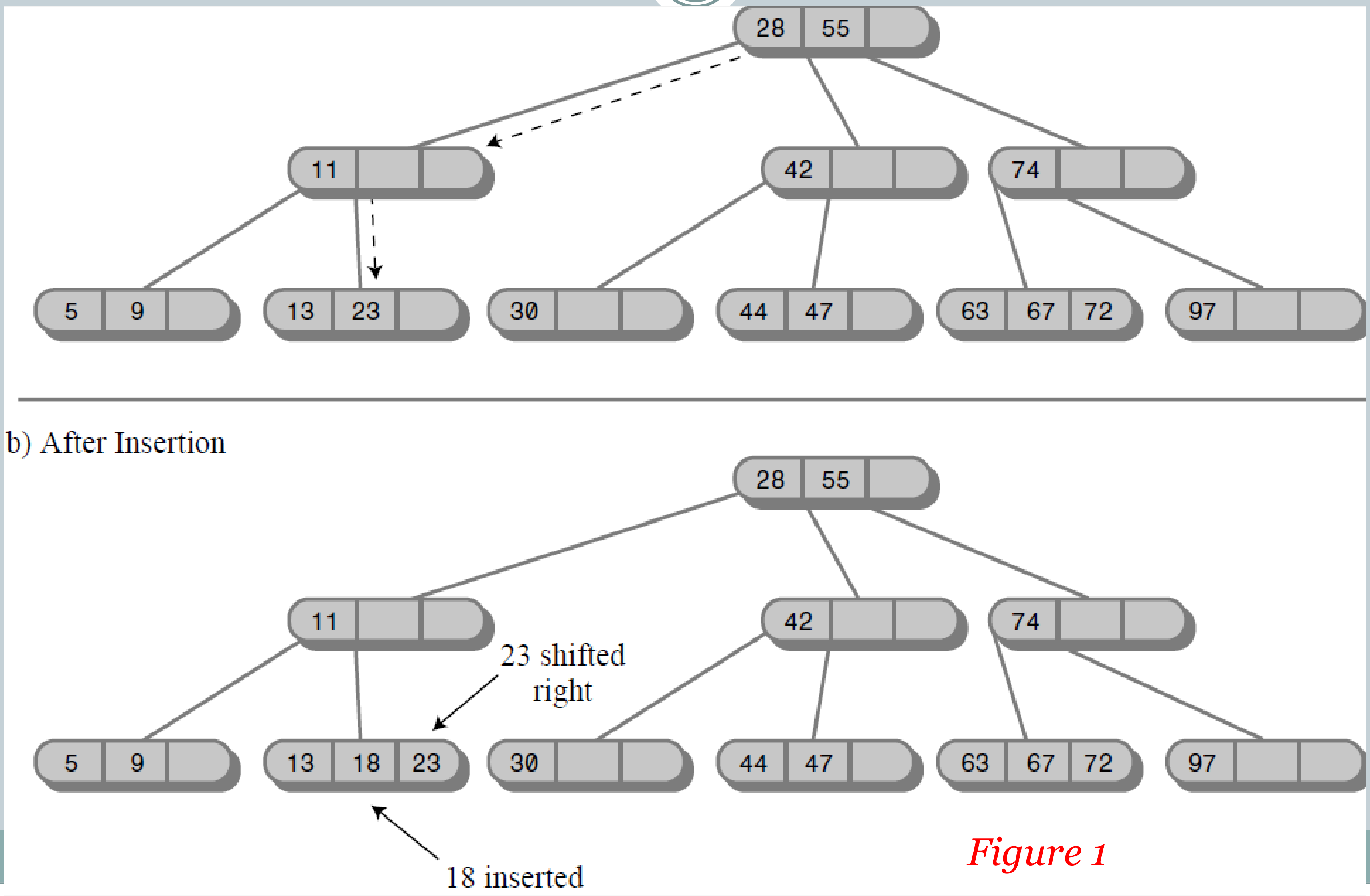
Outline

- Binary Trees
- Traversing Binary Tree
- Red-Black Trees
- Red-Black Tree Insertions
- 2-3-4 Trees
- 2-3-4 Trees Insertions

2-3-4 Trees Insertion

- New data items are always inserted in leaves, which are on the bottom row of the tree.
- Insertion into a 2-3-4 tree is sometimes quite easy and sometimes rather complicated.
  - If no full nodes are encountered during the search, insertion is easy. When the appropriate leaf node is reached, the new data item is simply inserted into it *(Figure 1)*.
  - Insertion becomes more complicated if a full node is encountered on the path down to the insertion point. When this happens, the node must be *split*. The splitting process keeps the tree balanced *(Figure 2)*.

Insertion with no Splits



b) After Insertion

23 shifted right

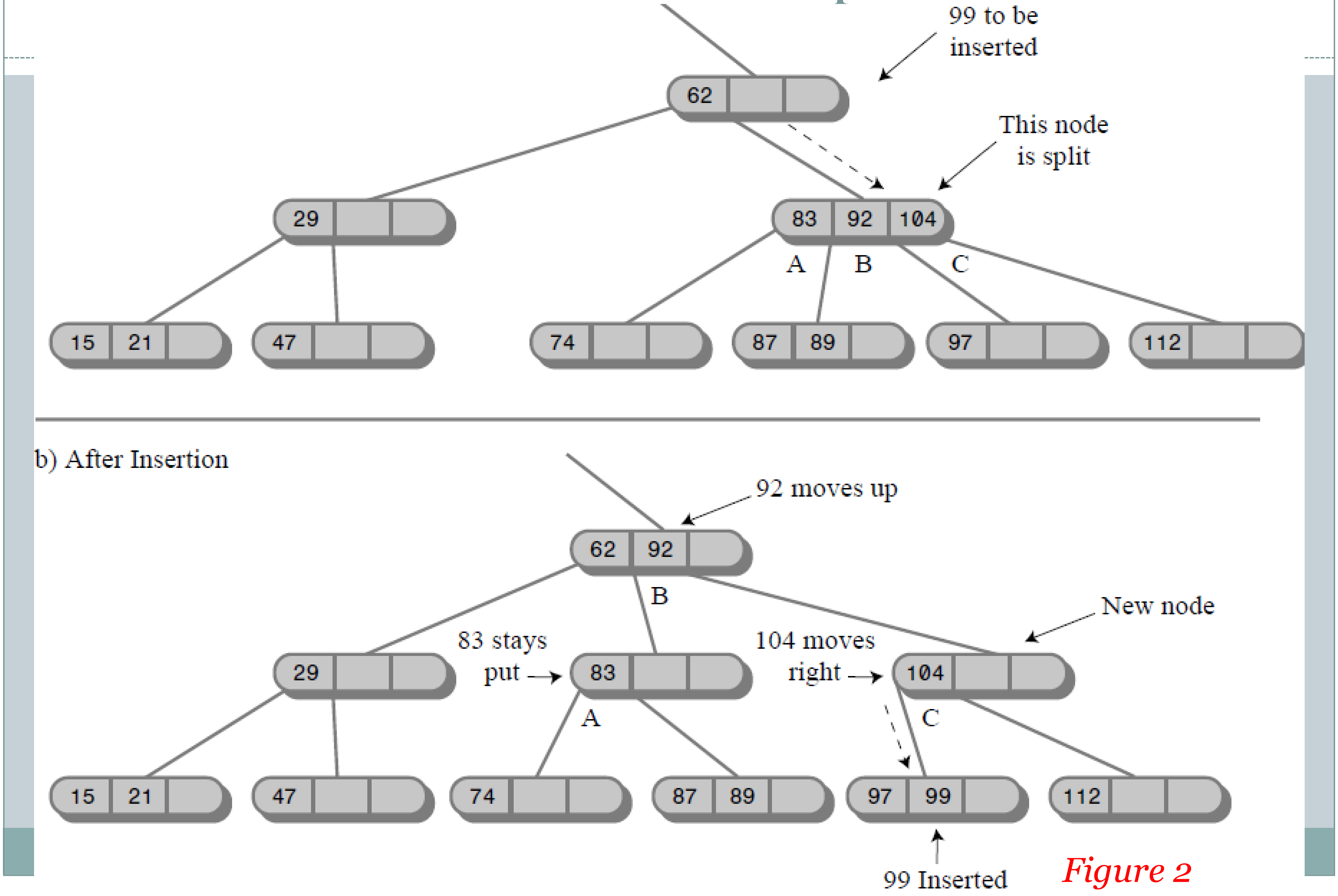
18 inserted

Figure 1

Splits Procedure

- Let’s name the data items in the node that’s about to be split **A**, **B**, and **C**.
- Splits Procedure:
  - A new, empty node is created. It’s a sibling of the node being split, and is placed to its right.
  - Data item **C** is moved into the new node.
  - Data item **B** is moved into the parent of the node being split.
  - Data item **A** remains where it is.
  - The rightmost two children are disconnected from the node being split and connected to the new node.

Insertion with Splits



99 to be inserted

This node is split

b) After Insertion

92 moves up

New node

83 stays put

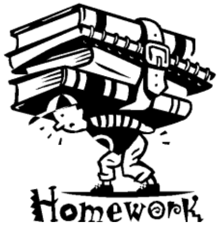
104 moves right

99 Inserted

Figure 2

Item Insertion in 2-3-4 Trees Pseudo Code

```
void Insert( int Key ){
  PCurrent = Root;
  While( true ){
   if PCurrent is full {
      Splits(PCurrent);
      PCurrent = PCurrent->Parent()
      PCurrent goes to next child base on value of Key
   }
   else if PCurrent is Leaf {break;}
   else PCurrent goes to next child base on value of Key
  }
  Insert item Key to PCurrent node
}
```



Homework 24
submit to:
fe.assignment@gmail.com

18-March-2016
@ 15:00

Base on Pseudo Code, write a function to insert an item to 2-3-4 Trees

Late submission: the score will be minus 10% for every hour