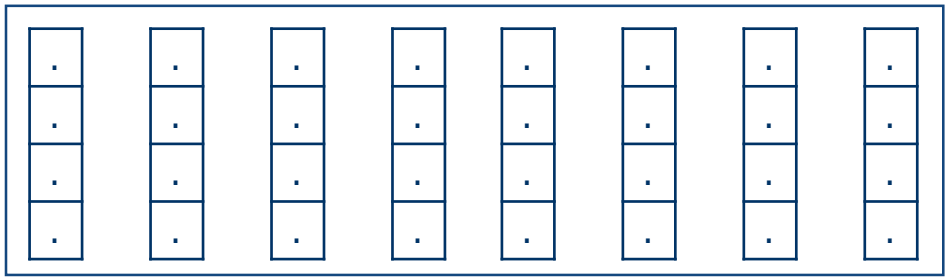
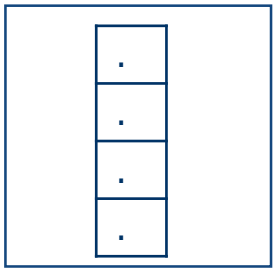

Chap 5. Trees (5)

Contents

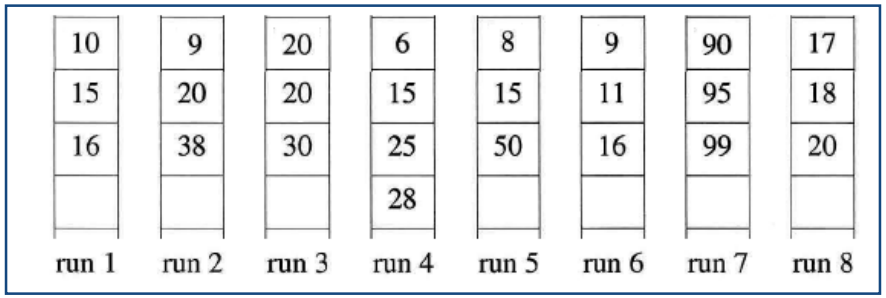
5.8 Selection Trees



External storage



Internal memory



External storage

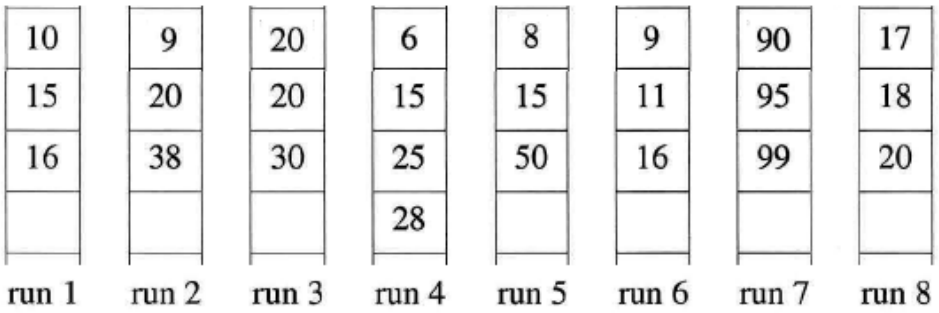
ordered sequences

KNU KYUNGPOOK NATIONAL UNIVERSITY
School of Computer Science and Engineering

5.8 Selection Trees

5.8.1 Introduction

- ***k* ordered sequences**, called ***runs***,
  to be merged into **a single ordered sequence**.



- **The merging task can be accomplished by repeatedly outputting the record with the smallest key.**
- **For *k*>2, we can *reduce the number of comparisons* by using the selection tree; winner trees and loser trees.**

KNU KYUNGPOOK NATIONAL UNIVERSITY
School of Computer Science and Engineering

5.8.2 Winner Trees

- **A *winner tree* is a complete binary tree in which each node represents the smaller of its two children.**

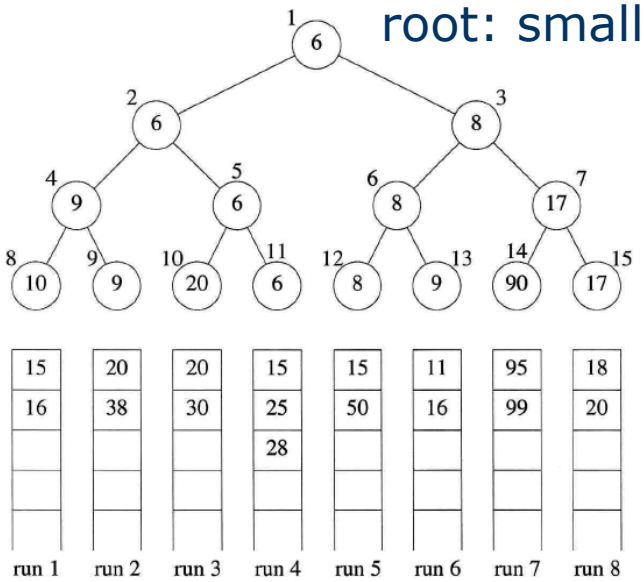


Figure 5.32: Winner tree for $k = 8$, showing the first three keys in each of the eight runs

sequential allocation (complete binary tree)

Each node contains only a pointer to the record

Leaf node: the first record in the corresponding run

Runs : ordered sequences

KNU KYUNGPOOK NATIONAL UNIVERSITY
School of Computer Science and Engineering

5.8.2 Winner Trees

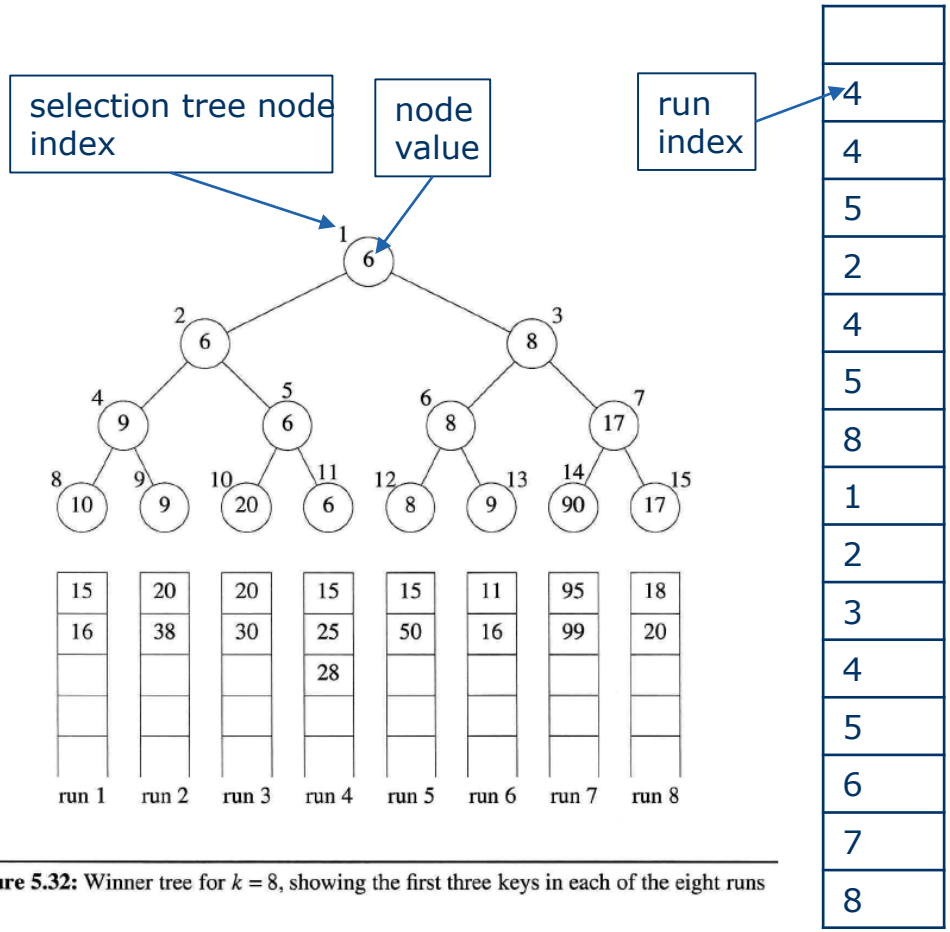
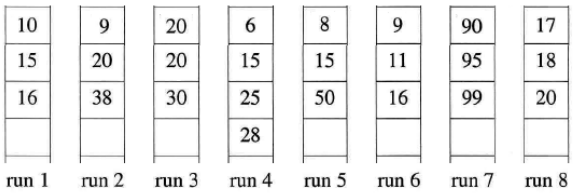


Figure 5.32: Winner tree for $k = 8$, showing the first three keys in each of the eight runs

implemetation of winner tree

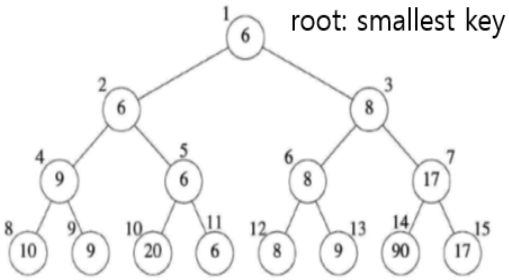
- section tree node arry

Each node contains only a pointer to the record



KNU KYUNGPOOK NATIONAL UNIVERSITY
School of Computer Science and Engineering

5.8.2 Winner Trees



	0	1	2	3	4	5	6	7	8
sortedIdx	-	1	1	1	1	1	1	1	1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
winTree	-	4	4	5	2	4	5	8	1	2	3	4	5	6	7	8

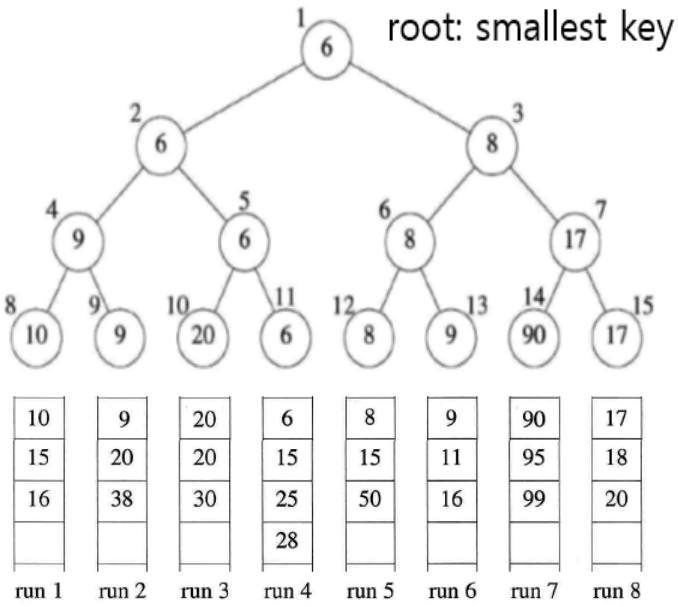
nums

각 레코드의 **1**번째 원소들을 정렬해야 함

10	9	20	6	8	9	90	17
11	10	21	7	9	10	91	18
12	11	22	8	10	11	92	19
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
19	18	29	15	17	18	99	26

5.8.2 Winner Trees

4(5)
4(2)
5
2
4
5
8
1
2
3
4
5
6
7
8



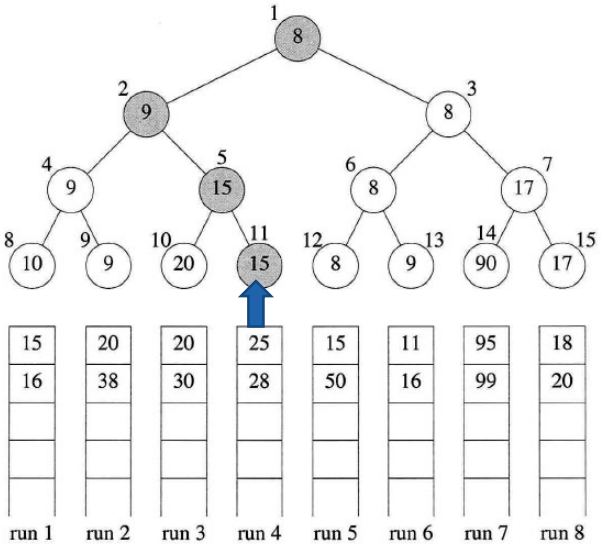
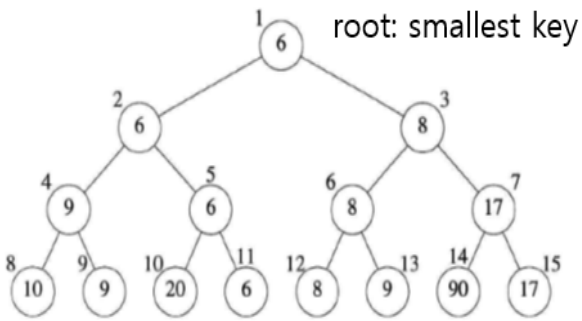


Figure 5.33: Winner tree of Figure 5.32 after one record has been output and the tree restructured (nodes that were changed are shaded)

KNU KYUNGPOOK NATIONAL UNIVERSITY
School of Computer Science and Engineering

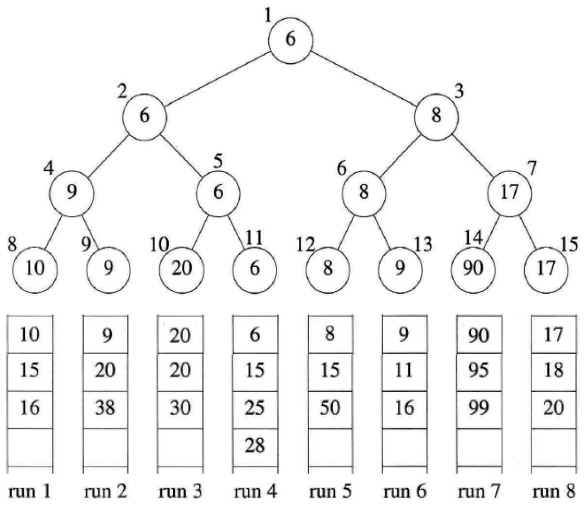
5.8.2 Winner Trees

- **Analysis of merging runs using winner trees**
  - *Let n be the number of records in all k* runs.
  - The number of levels in the tree is $[\log_2 k+1]$
  - The time to restructure the tree is $O(\log_2 k)$.
  - The time required to merge all n records is $O(n\log_2 k)$.
  - The time required to set up the selection tree the first time is $O(k)$.
  - The total time needed to merge the k runs is $O(n\log_2 k)$.



5.8.3 Loser Trees

- **A selection tree in which each nonleaf node retains a pointer to the loser is called a *loser tree*.**



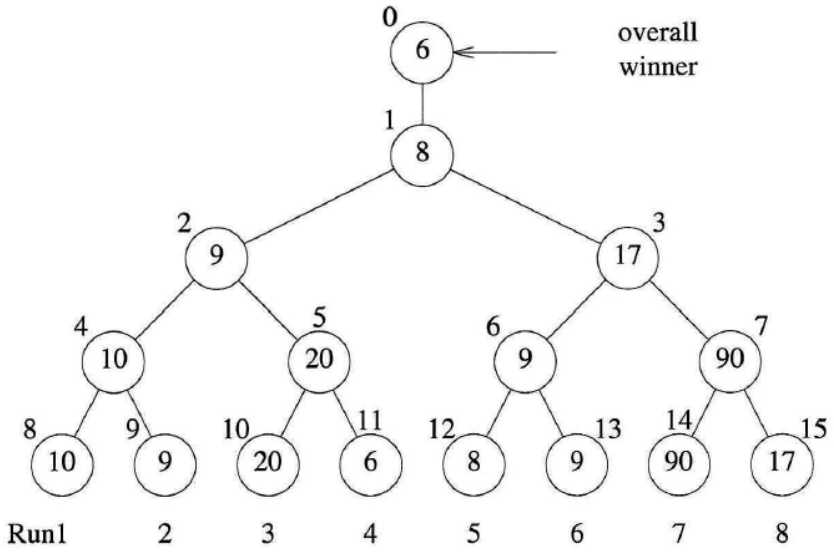
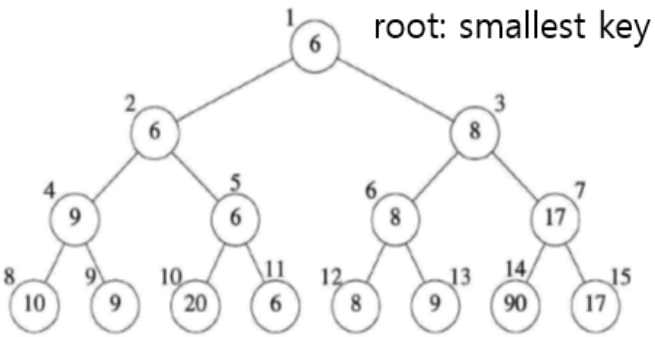


Figure 5.34: Loser tree corresponding to winner tree of Figure 5.32

KNU KYUNGPOOK NATIONAL UNIVERSITY
School of Computer Science and Engineering

5.8.3 Loser Trees

- **In winner tree, following the output of the overall winner, the tree is restructured by playing tournaments along the path from node 11 to node 1.**
- ***In loser, the records with which the tournaments are to be played* are readily available *from the parent nodes*.**
  - As a result, *sibling nodes along the path from 11 to l are not accessed*.



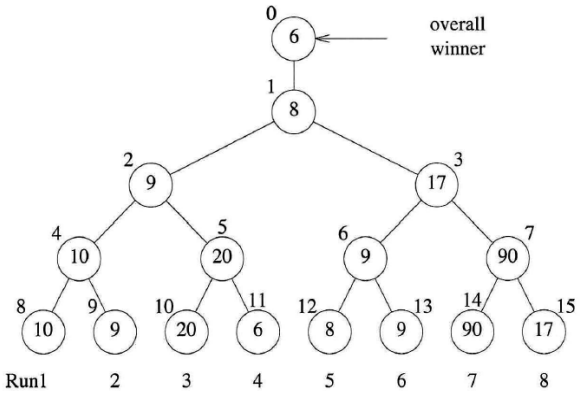


Figure 5.34: Loser tree corresponding to winner tree of Figure 5.32