

**CS 14: Data Structures and Algorithms**

**Monday, November 24, 2003**

**Quiz 7, Form: A**

Name: \_\_\_\_\_

Login: \_\_\_\_\_

ID Number: \_\_\_\_\_

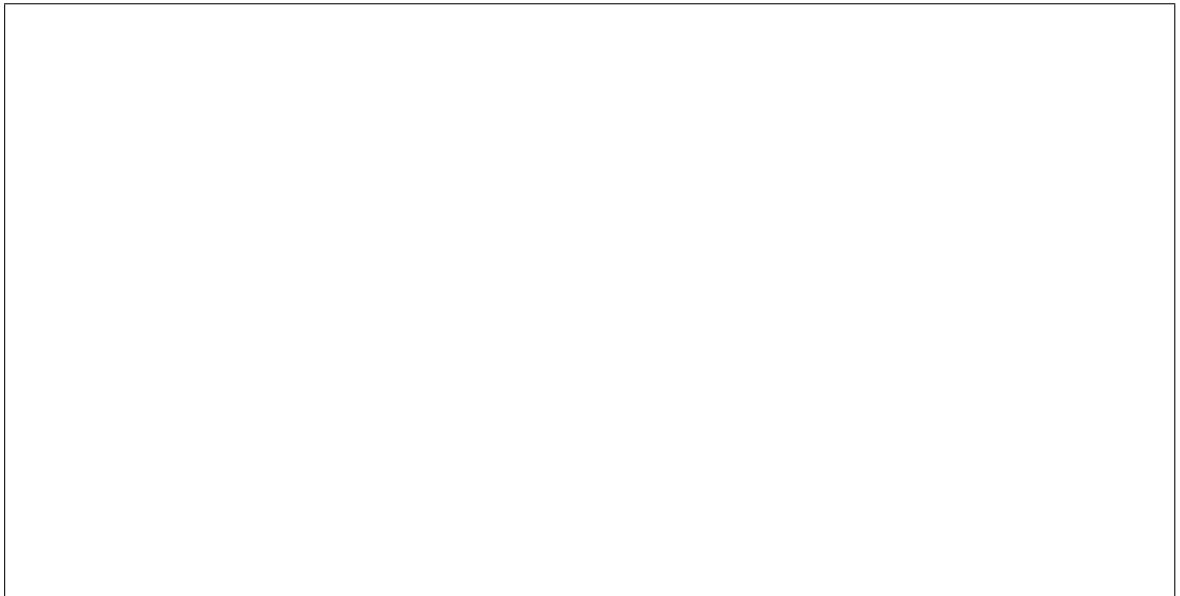
Signature: \_\_\_\_\_

**This quiz is worth 10 points and lasts 12 minutes. Good luck!**

1. [1 point] In the array implementation of a Binary Heap where the root has index 1, the **right** child of the node with index  $i$  is the node with index:
  - (a)  $(i - 1)/2$
  - (b)  $(i + 1)/2$
  - (c)  $2 * i + 1$
  - (d)  $2 * i + 2$
  - (e)  $2 * i$
  - (f)  $i/2$
  
2. [1 point] In the array implementation of a Binary Heap where the root has index 1, the **parent** node of the node with index  $i$  is the node with index:
  - (a)  $(i - 1)/2$
  - (b)  $(i + 1)/2$
  - (c)  $2 * i + 1$
  - (d)  $2 * i + 2$
  - (e)  $2 * i$
  - (f)  $i/2$
  
3. [1 point] In a Maximum Binary Heap, the node with the minimum key is:
  - (a) the root node.
  - (b) one of the leaf nodes.
  - (c) the left-most node.
  - (d) the right-most node.
  - (e) one of the internal nodes.

4. [1 point] The runtime efficiency of a single percolate-down operation in a Binary Heap with  $n$  nodes is:
- (a)  $\mathcal{O}[\log_2(n)]$ .
  - (b)  $\mathcal{O}[n \log_2(n)]$ .
  - (c)  $\mathcal{O}[1]$ .
  - (d)  $\mathcal{O}[n]$ .
  - (e)  $\mathcal{O}[n^2]$ .
5. [1 point] The fastest implementation of a regular Queue is:
- (a) an array implementation.
  - (b) a singly-linked list implementation, without tail.
  - (c) a doubly-linked list implementation, without tail.
  - (d) a singly-linked list implementation, with tail.
  - (e) a doubly-linked list implementation, with tail.
6. [1 point] A complete binary tree with 5 levels:
- (a) has at least 15 nodes and at most 31 nodes.
  - (b) has at least 15 nodes and at most 32 nodes.
  - (c) has at least 16 nodes and at most 31 nodes.
  - (d) has at least 16 nodes and at most 32 nodes.
  - (e) None of the other alternatives is correct.
7. [1 point] The runtime efficiency of creating a Binary Heap from  $n$  keys:
- (a) is always  $\mathcal{O}[n]$ .
  - (b) is  $\mathcal{O}[n^2]$ .
  - (c) depends on whether we're building a minimum or a maximum binary heap.
  - (d) is always  $\mathcal{O}[n \log_2(n)]$ .
  - (e) can be made to be  $\mathcal{O}[n]$ .
  - (f) None of the other alternatives is correct.

8. [3 points] Draw the Maximum Binary Heap that results from the insertion, one at a time, of nodes with keys 27, 59, 21, 38, 54, 63, 8, 70, 15, in that order, maintaining the heap order after every insertion. **Use the scratch pages for your intermediate work and draw only the final result in the box below.**



# Answer Key for Exam A

This quiz is worth 10 points and lasts 12 minutes. Good luck!

1. [1 point] In the array implementation of a Binary Heap where the root has index 1, the **right** child of the node with index  $i$  is the node with index:
  - (a)  $(i - 1)/2$
  - (b)  $(i + 1)/2$
  - (c)  $2 * i + 1$**
  - (d)  $2 * i + 2$
  - (e)  $2 * i$
  - (f)  $i/2$
  
2. [1 point] In the array implementation of a Binary Heap where the root has index 1, the **parent** node of the node with index  $i$  is the node with index:
  - (a)  $(i - 1)/2$
  - (b)  $(i + 1)/2$
  - (c)  $2 * i + 1$
  - (d)  $2 * i + 2$
  - (e)  $2 * i$
  - (f)  $i/2$**
  
3. [1 point] In a Maximum Binary Heap, the node with the minimum key is:
  - (a) the root node.
  - (b) one of the leaf nodes.**
  - (c) the left-most node.
  - (d) the right-most node.
  - (e) one of the internal nodes.

4. [1 point] The runtime efficiency of a single percolate-down operation in a Binary Heap with  $n$  nodes is:
- (a)  $\mathcal{O}[\log_2(n)]$ .
  - (b)  $\mathcal{O}[n \log_2(n)]$ .
  - (c)  $\mathcal{O}[1]$ .
  - (d)  $\mathcal{O}[n]$ .
  - (e)  $\mathcal{O}[n^2]$ .
5. [1 point] The fastest implementation of a regular Queue is:
- (a) an array implementation.
  - (b) a singly-linked list implementation, without tail.
  - (c) a doubly-linked list implementation, without tail.
  - (d) a singly-linked list implementation, with tail.
  - (e) a doubly-linked list implementation, with tail.
6. [1 point] A complete binary tree with 5 levels:
- (a) has at least 15 nodes and at most 31 nodes.
  - (b) has at least 15 nodes and at most 32 nodes.
  - (c) has at least 16 nodes and at most 31 nodes.
  - (d) has at least 16 nodes and at most 32 nodes.
  - (e) None of the other alternatives is correct.
7. [1 point] The runtime efficiency of creating a Binary Heap from  $n$  keys:
- (a) is always  $\mathcal{O}[n]$ .
  - (b) is  $\mathcal{O}[n^2]$ .
  - (c) depends on whether we're building a minimum or a maximum binary heap.
  - (d) is always  $\mathcal{O}[n \log_2(n)]$ .
  - (e) can be made to be  $\mathcal{O}[n]$ .
  - (f) None of the other alternatives is correct.

8. [3 points] Draw the Maximum Binary Heap that results from the insertion, one at a time, of nodes with keys 27, 59, 21, 38, 54, 63, 8, 70, 15, in that order, maintaining the heap order after every insertion. **Use the scratch pages for your intermediate work and draw only the final result in the box below.**

**Answer:**

