COS 226	Algorithms and Data Structures	Fall 2010

Midterm

This test has 8 questions worth a total of 60 points. You have 80 minutes. The exam is closed book, except that you are allowed to use a one page cheatsheet. No calculators or other electronic devices are permitted. Give your answers and show your work in the space provided. **Write out and sign the Honor Code pledge before turning in the test.**

"I pledge my honor that I have not violated the Honor Code during this examination."

Problem	Score
0	
1	
2	
3	
Sub 1	

Problem	Score
4	
5	
6	
7	
Sub 2	

Total	

Name:

Login ID:

Precept:

P01	11	Bob Tarjan
P02	12:30	Yuri Pritykin
P02A	12:30	Bob Tarjan
P03	1:30	Aman Dhesi
P03A	1:30	Siyu Yang

0. **Miscellaneous. (2 point)**

   In the space provided on the front of the exam, write your name and Princeton NetID; circle your precept number; and sign the honor code.

1. **Analysis of algorithms. (14 points)**

   (a) For each expression in the left column, give the best matching description from the right column.

   ____ $1 + 2 + 4 + 8 + 16 + \ldots + N$

   ____ $1 + 2 + 3 + 4 + 5 + \ldots + N$

   ____ $1 + 3 + 5 + 7 + 9 + \ldots + N$

   ____ $\frac{1}{2}N^2$

   ____ $\frac{1}{2}N^2 + 100N \lg N$

   ____ N^2

   ____ N^3

   A. $\sim \frac{1}{2}N^2$.

   B. $O(N^2)$.

   C. Both A and B.

   D. Neither A nor B.

   (b) For each quantity in the left column, give the best matching description from the right column.

   ____ Height of a weighted quick union data structure with N items.

   ____ Height of a binary heap with N keys.

   ____ Height of a left-leaning red-black BST with N keys.

   ____ Maximum function-call stack size when (top-down) mergesorting N keys.

   ____ Maximum function-call stack size when quicksorting N keys.

   ____ Number of compares to binary search in a sorted array of size N.

   A. $\sim \lg N$ in the best case.

   B. $\sim \lg N$ in the worst case.

   C. Both A and B.

   D. Neither A nor B.

(c) Consider the following code fragment.

```
int count = 0;
int N = a.length;
Arrays.sort(a);
for (int i = 0; i < N; i++) {
   for (int j = i+1; j < N; j++) {
       if (Arrays.binarySearch(a, a[i] + a[j]) count++;
    }
}
```

Suppose that it takes 1 second when $N = 3,500$. Approximately how long will it take when $N = 35,000$? Circle the best answer.

10 seconds   20 seconds   1 minute   2 minutes   1 hour   2 hours

(d) Consider the following Java data type definition for a 2-3 tree, where the nested class `Node` represents either a 2-node or a 3-node.

```
public class TwoThreeTree<Key extends Comparable<Key>, Value> {
   private Node root;

   private class Node {
      private int count;                  // subtree count
      private Key key1, key2;             // the one or two keys
      private Value value1, value2;       // the one or two values
      private Node left, middle, right;   // the two or three subtrees
   }
   ...
}
```

How much memory (in bytes) does each `Node` object consume? Circle your answer.

2. **8 sorting algorithms. (8 points)**

The column on the left is the original input of strings to be sorted; the column on the right are the string in sorted order; the other columns are the contents at some intermediate step during one of the 8 sorting algorithms listed below. Match up each algorithm by writing its number under the corresponding column. Use each number exactly once.

john	aviv	alan	ctai	andy	aviv	yort	alan	anna	alan
tzha	azhu	anna	fyau	alan	azhu	tzha	andy	fyau	andy
fyau	ctai	fyau	john	ddix	ctai	vyas	anna	ctai	anna
ctai	ddix	ctai	tzha	azhu	ddix	ravi	aviv	andy	aviv
nbal	fyau	andy	azhu	azhu	fyau	sida	azhu	ddix	azhu
ddix	john	ddix	ddix	anna	john	oleg	azhu	azhu	azhu
sguo	kuan	azhu	nbal	fyau	kuan	sguo	ctai	azhu	ctai
azhu	nbal	azhu	sguo	ctai	nbal	peck	ddix	aviv	ddix
aviv	sguo	aviv	aviv	john	oleg	ctai	fyau	alan	fyau
sida	sida	john	kuan	aviv	sguo	nbal	john	john	john
kuan	tzha	kuan	sida	kuan	sida	kuan	kuan	vyas	kuan
vyas	vyas	vyas	vyas	lily	tzha	lily	kwak	oleg	kwak
oleg	kwak	oleg	kwak	nbal	vyas	fyau	oleg	levy	levy
levy	levy	levy	levy	kwak	levy	levy	levy	kwak	lily
kwak	muir	kwak	muir	sguo	kwak	kwak	vyas	muir	muir
muir	oleg	muir	oleg	muir	muir	muir	muir	peck	nbal
peck	peck	peck	alan	oleg	peck	azhu	peck	ravi	oleg
ravi	ravi	ravi	peck	levy	ravi	aviv	ravi	kuan	peck
alan	alan	sida	ravi	sida	alan	alan	sida	yort	ravi
yort	andy	yort	yort	vyas	yort	john	yort	sida	sguo
azhu	anna	sguo	andy	peck	azhu	azhu	nbal	sguo	sida
andy	azhu	nbal	anna	ravi	andy	andy	tzha	nbal	tzha
anna	lily	tzha	azhu	tzha	anna	anna	sguo	lily	vyas
lily	yort	lily	lily	yort	lily	ddix	lily	tzha	yort
----	----	----	----	----	----	----	----	----	----
0									1

(0) Original input

(1) Sorted

(2) Selection sort

(3) Insertion sort

(4) Shellsort (*13-4-1 increments*)

(5) Mergesort (*top-down*)

(6) Mergesort (*bottom-up*)

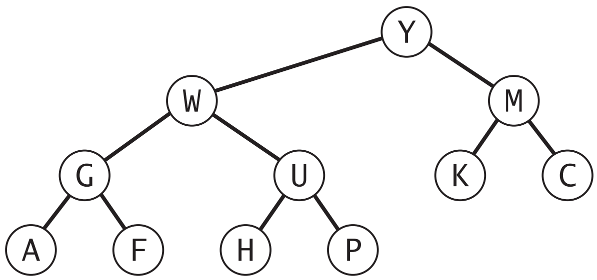
(7) Quicksort (*standard, no shuffle*)

(8) Quicksort (*3-way, no shuffle*)

(9) Heapsort

3. **Binary heaps. (6 points)**

Consider the following binary tree representation of a max-heap.



(a) Give the array representation of the heap.

0	1	2	3	4	5	6	7	8	9	10	11	12
-												-

(b) *Delete* the maximum key. Give the resulting heap, *circling* any entries that changed.

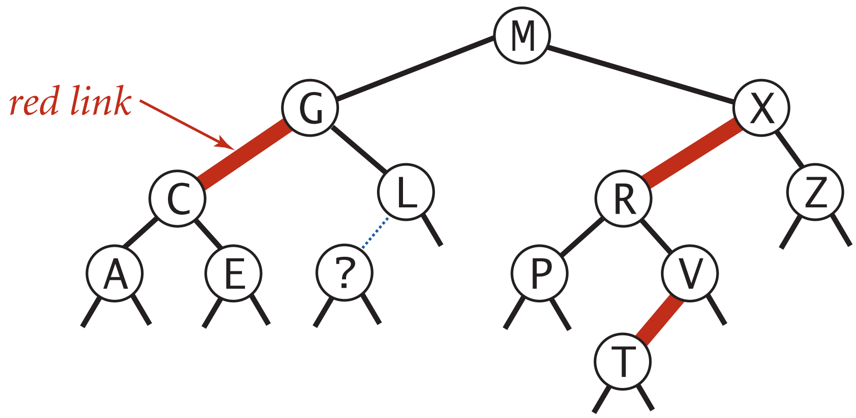
0	1	2	3	4	5	6	7	8	9	10	11	12
-											-	-

(c) *Insert* the key `Q` into the *original* binary heap. *circling* any entries that changed.

0	1	2	3	4	5	6	7	8	9	10	11	12
-												

4. **Red-black BSTs. (8 points)**

   Consider the following left-leaning red-black BST.



(a) Which one or more of the keys below could be in the node labeled with a question mark?

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

(b) What is the color of the link from the node labeled with a question mark to its parent? Circle the best answer.

BLACK RED either RED or BLACK

(c) Add the key `U` and draw the resulting left-leaning red-black BST.

5. **Hashing. (6 points)**

Suppose that the following keys are inserted in the order

A B C D E F G

into an initially empty linear-probing hash table of size 7, using the following hash function:

key	hash(key, 7)
A	3
B	1
C	4
D	1
E	5
F	2
G	5

What is the result of the linear-probing array?

Assume that the array size is fixed and does not double.

0	1	2	3	4	5	6

6. **Bitonic max. (8 points)**

An array is *bitonic* if it consists of a strictly increasing sequence of keys immediately followed by a strictly decreasing sequence of keys. Design an algorithm that determines the maximum key in a bitonic array of size N in time proportional to $\log N$.

(a) *Give a crisp and concise English description of your algorithm—don't write Java code. Your answer will be graded on correctness, efficiency, and clarity.*

(b) To demonstrate your algorithm, list the first four compares that your algorithm performs to find the maximum key in the following bitonic array of 15 elements:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
a[i]	10	34	56	76	87	80	70	66	56	30	28	25	20	15	11

1.

2.

3.

4.

7. **Stable priority queue. (8 points)**

A min-based priority queue is *stable* if `min()` and `deleteMin()` return the minimum key that was least-recently inserted. Describe how to implement a `StableMinPQ` data type such that every operation takes at most (amortized) logarithmic time.

```
public class StableMinPQ<Key extends<Comparable<Key>>
```

	`StableMinPQ()`	*create an empty priority queue*
`Key`	**`min()`**	*return the minimum key (that was least-recently inserted)*
`Key`	**`delMin()`**	*delete and return the minimum key (that was least-recently inserted)*
`void`	**`insert(Key key)`**	*insert the key*
`boolean`	**`isEmpty()`**	*is the priority queue empty?*

Give a crisp and concise English description of your algorithm—don't write Java code. Your answer will be graded on correctness, efficiency, and clarity.