PRINCETON UNIVERSITY

Algorithms, Part I

Robert Sedgewick
Kevin Wayne

Feedback — Final Exam

You have submitted this quiz on **Sat 29 Sep 2012 2:00:12 AM PDT**. You achieved a score of **17.90** out of **20.00**.

```
To specify an array or sequence of values in an answer, you must separate the values by a
single space character (with no punctuation and with no leading or trailing whitespace).
For example, if the question asks for the first ten powers of two (starting at 1), the only
accepted answer is:

1 2 4 8 16 32 64 128 256 512

If you wish to discuss a particular question and answer in the forums, please post the
entire question and answer, including the seed (which is used by the course staff to
uniquely identify the question) and the explanation (which contains the correct answer).
```

Question 1

```
(seed = 775685)
Give the id[] array that results from the following sequence of union operations
on a set of 10 items using the weighted quick-union algorithm from lecture.

    7-0 0-6 2-1 3-4 0-5 4-2 5-3 8-5 5-9

Recall: when joining two trees of equal size, our weighted quick union convention
is to make the root of the second tree point to the root of the first tree.
```

7 2 3 7 3 7 7 7 7 7

Your Answer		Score	Explanation
7 2 3 7 3 7 7 7 7 7	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
7 2 3 7 3 7 7 7 7 7
```

```
Here is the id[] array after each union operation:

      0 1 2 3 4 5 6 7 8 9
7-0:  7 1 2 3 4 5 6 7 8 9
0-6:  7 1 2 3 4 5 7 7 8 9
2-1:  7 2 2 3 4 5 7 7 8 9
3-4:  7 2 2 3 3 5 7 7 8 9
0-5:  7 2 2 3 3 7 7 7 8 9
```

```
4-2:  7 2 3 3 3 7 7 7 8 9
5-3:  7 2 3 7 3 7 7 7 8 9
8-5:  7 2 3 7 3 7 7 7 7 9
5-9:  7 2 3 7 3 7 7 7 7 7
```

Question 2

```
(seed = 69715)
Suppose that you binary search for the key 27 in the following sorted array of size 15:

    13 19 24 30 43 48 49 53 63 72 76 78 82 84 92

Give the sequence of keys in the array that are compared with 27.
```

53 30 19 24

Your Answer		Score	Explanation
53 30 19 24	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
53 30 19 24
```

```
Here is the array to be searched after each compare:

     13 19 24 30 43 48 49 53 63 72 76 78 82 84 92
53:  13 19 24 30 43 48 49  -  -  -  -  -  -  -  -
30:  13 19 24  -  -  -  -  -  -  -  -  -  -  -  -
19:   -  - 24  -  -  -  -  -  -  -  -  -  -  -  -
24:   -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
```

Question 3

```
(seed = 369735)
Suppose that you time a program as a function of N and produce the following table.

        N   seconds
-------------------
       81      0.00
      243      0.01
      729      0.39
     2187      8.86
     6561    222.59
    19683   5579.55


Estimate the order of growth of the running time as a function of N. Assume that the
running
time obeys a power law T(N) ~ a N^b. For your answer, enter the constant b. Your answer
will
be marked as correct if it is within 1% of the target answer - we recommend using
```

```
two digits after the decimal separator, e.g., 2.34.
```

2.93

Your Answer		Score	Explanation
2.93	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

```
The theoretical order-of-growth is N ^ (44/15) = 2.93

The empirical order-of-growth is N ^ (log_3 ratio)

                                  log_3
        N   seconds     ratio     ratio
---------------------------------------
       81      0.00     25.30      2.94
      243      0.01     24.09      2.90
      729      0.39     27.16      3.01
     2187      8.86     23.00      2.85
     6561    222.59     25.13      2.93
    19683   5579.55     25.07      2.93
```

Question 4

```
(seed = 273152)
What is the order of growth of the worst case running time of the following code
fragment
as a function of N?

int sum = 0;
for (int i = 1; i <= N; i++)
    for (int j = 1; j <= i*N; j++)
        sum++;
```

Your Answer		Score	Explanation
● N^3	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

```
N^3
```

```
The body of the innermost loop executes N + 2N + 3N + 4N + ... + N^2 ~ 1/2 N^3 times.
```

Question 5

(seed = 406108)
Given the following definition of a MysteryBox object:

```
public class MysteryBox {
    private boolean x0, x1;
    private long y0;
    private int z0, z1;
    private double[] a = new double[192];

    ...
}
```

Using the 64-bit memory cost model from lecture, how many bytes does each object of type MysteryBox use?

1608

Your Answer		Score	Explanation
1608	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

1608

```
public class MysteryBox {                       //   16 (object overhead)
    private boolean x0, x1;                     //    2 (2 boolean)
    private long y0;                            //    8 (1 long)
    private int z0, z1;                         //    8 (2 int)
    private double[] a = new double[192];       //    8 (reference to array)
                                                // 1560 (double array of size 192)
    ...                                                6 (padding)
}                                                   ----
                                                    1608
```

Question 6

(seed = 141307)
Suppose that you have a data type for a sequence of N items and that it is represented internally using a doubly-linked list Assume that the data type is implemented in an efficient and natural manner given the specified representation.

Match up each of the following operations with their worst-case running time.
You may use each number once, more than once, or not at all.

```
___ is an item in the sequence?                                     0. 1

___ remove and return the last item in the sequence                 1. log N

___ replace the ith item in the sequence with a different item      2. N

___ insert an item at the beginning of the sequence                 3. N log
```

```
N

___ remove and return the ith item in the sequence                    4. N^2

___ insert an item immedidately after the ith item in the sequence
```

2 0 2 0 2 2

Your Answer		Score	Explanation
2	✓	0.17	
0	✓	0.17	
2	✓	0.17	
0	✓	0.17	
2	✓	0.17	
2	✓	0.17	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
2 0 2 0 2 2
```

Question 7

```
(seed = 815003)
Give the array that results after the first 4 exchanges when selection sorting the
following array:

    44 12 14 71 50 51 47 40 25 79
```

12 14 25 40 50 51 47 71 44 79

Your Answer		Score	Explanation
12 14 25 40 50 51 47 71 44 79	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
12 14 25 40 50 51 47 71 44 79
```

```
Here is the array after each exchange:

   44 12 14 71 50 51 47 40 25 79
1: 12 44 14 71 50 51 47 40 25 79
2: 12 14 44 71 50 51 47 40 25 79
3: 12 14 25 71 50 51 47 40 44 79
4: 12 14 25 40 50 51 47 71 44 79
```

Question 8

```
(seed = 397433)
Give the array that results immediately after the 7th calls to merge()
when bottom-up mergesorting the following array:

31 41 28 84 63 48 27 75 50 86
```

28 31 41 63 84 27 48 75 50 86

Your Answer		Score	Explanation
28 31 41 63 84 27 48 75 50 86	✗	0.00	
Total		0.00 / 1.00	

Question Explanation

The correct answer is:

```
28 31 41 84 27 48 63 75 50 86
```

```
Here is the array immediately after each call to merge():

                 31 41 28 84 63 48 27 75 50 86
merge(0, 0, 1):  31 41 28 84 63 48 27 75 50 86
merge(2, 2, 3):  31 41 28 84 63 48 27 75 50 86
merge(4, 4, 5):  31 41 28 84 48 63 27 75 50 86
merge(6, 6, 7):  31 41 28 84 48 63 27 75 50 86
merge(8, 8, 9):  31 41 28 84 48 63 27 75 50 86
merge(0, 1, 3):  28 31 41 84 48 63 27 75 50 86
merge(4, 5, 7):  28 31 41 84 27 48 63 75 50 86
```

Question 9

```
(seed = 925507)
Give the array that results after applying quicksort partitioning to the following
array:

    71 52 44 55 49 85 92 51 10 45 13 24

Use the standard partitioning algorithm, in which the leftmost entry is the partitioning
item.
```

45 52 44 55 49 24 13 51 10 71 92 85

Your Answer		Score	Explanation
45 52 44 55 49 24 13 51 10 71 92 85	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
45 52 44 55 49 24 13 51 10 71 92 85
```

```
Here is the array before and after each exchange:

  i   j    0   1   2   3   4   5   6   7   8   9  10  11
--------------------------------------------------------
  0  12   71  52  44  55  49  85  92  51  10  45  13  24
  5  11   71  52  44  55  49  85  92  51  10  45  13  24
  5  11   71  52  44  55  49  24  92  51  10  45  13  85
  6  10   71  52  44  55  49  24  92  51  10  45  13  85
  6  10   71  52  44  55  49  24  13  51  10  45  92  85
 10   9   45  52  44  55  49  24  13  51  10  71  92  85
      9   45  52  44  55  49  24  13  51  10  71  92  85
```

Question 10

```
(seed = 990625)
Give the sequence of the 7 keys in the array that results after performing 3 successive
delete-the-max
operations on the following maximum-oriented binary heap of size 10:

    92 89 85 61 55 31 41 43 53 48
```

61 55 48 53 43 31 41

Your Answer		Score	Explanation

61 55 48 53 43 31 41	✓	1.00	Here is the sequence of keys in the array after each deletion:
			
			`                92 89 85 61 55 31 41 43 53 48`
			`[ 92 deleted ]  89 61 85 53 55 31 41 43 48`
			`[ 89 deleted ]  85 61 48 53 55 31 41 43`
			`[ 85 deleted ]  61 55 48 53 43 31 41`

Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
61 55 48 53 43 31 41
```

```
Here is the sequence of keys in the array after each deletion:
```

```
                 92 89 85 61 55 31 41 43 53 48
[ 92 deleted ]  89 61 85 53 55 31 41 43 48
[ 89 deleted ]  85 61 48 53 55 31 41 43
[ 85 deleted ]  61 55 48 53 43 31 41
```

Question 11

```
(seed = 295488)
Suppose that you have a priority queue containing N keys that is represented internally
using an unsorted array. Assume that the data type is implemented in an efficient and
natural manner given the specified representation.

Match up each of the following operations with their amortized running times.
You may use each number once, more than once, or not at all.

___ does the priority queue contain a given key?              0. 1

___ delete and return a minimum key                           1. log N

___ insert N keys                                             2. N

___ return a minimum key                                      3. N log N

___ insert a key                                              4. N^2

___ delete and return a maximum key
```

2 2 2 2 0 2

Your Answer		Score	Explanation
2	✓	0.17	
2	✓	0.17	
2	✓	0.17	
2	✓	0.17	
0	✓	0.17	
2	✓	0.17	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
2 2 2 2 0 2
```

Question 12

(seed = 543481)
The column on the left contains the original input of 24 strings to be sorted or shuffled;
the column on the right contains the strings in sorted order; the other columns contain the
contents at some intermediate step during one of the 9 algorithms listed below.

Match up each column with the corresponding sorting or shuffling algorithm from the list given below.
You should use each algorithm exactly once. That is, your answer should be a permutation of the 11
integers 0 to 10, starting with 0, ending with 10, and with each integer separated by a single space.

0	mint	ecru	bole	lava	aqua	ecru	aqua	pine	flax	silk	aqua
1	ecru	iris	buff	ecru	bole	mint	blue	buff	aqua	sand	blue
2	ruby	bole	drab	iris	buff	bole	bole	mint	buff	ruby	bole
3	bole	jade	ecru	bole	drab	ruby	buff	jade	blue	rose	buff
4	jade	blue	jade	jade	ecru	jade	drab	drab	jade	rust	drab
5	pine	buff	lust	blue	flax	pine	ecru	mist	lime	pine	ecru
6	buff	lime	mint	buff	jade	buff	flax	lust	ecru	flax	flax
7	pink	aqua	mist	lime	lava	pink	iris	silk	bole	pink	iris
8	silk	flax	pine	aqua	lust	mist	jade	ecru	lava	puce	jade
9	mist	lust	pink	flax	mint	silk	lava	pink	mint	plum	lava
10	lust	drab	ruby	lust	mist	drab	lime	bole	iris	mint	lime
11	drab	lava	silk	drab	pine	lust	lust	ruby	drab	iris	lust
12	lava	mint	aqua	mint	pink	flax	mist	lava	palm	lava	mint
13	flax	mist	flax	mist	rose	lava	ruby	flax	mist	buff	mist
14	aqua	rose	lava	silk	ruby	aqua	mint	aqua	pine	aqua	palm
15	rose	palm	palm	rose	silk	rose	rose	rose	lust	ecru	pine
16	palm	puce	puce	palm	palm	palm	palm	palm	pink	palm	pink
17	puce	silk	rose	puce	puce	puce	puce	puce	puce	bole	plum
18	lime	plum	blue	pink	lime	lime	silk	lime	rose	lime	puce
19	plum	pink	iris	plum	plum	plum	plum	plum	plum	mist	rose
20	blue	rust	lime	pine	blue	blue	pine	blue	rust	blue	ruby
21	rust	sand	plum	rust	rust	rust	rust	rust	silk	jade	rust
22	sand	pine	rust	sand	sand	iris	sand	sand	sand	lust	sand
23	iris	ruby	sand	ruby	iris	sand	pink	iris	ruby	drab	silk

```
      ___   ___   ___   ___   ___   ___   ___   ___   ___   ___
```

```
---

 0.  Original input
 1.  Selection sort
 2.  Insertion sort
 3.  Shellsort (with 3x + 1 increment sequence)
 4.  Mergesort (top-down)
 5.  Mergesort (bottom-up)
 6.  Quicksort (standard, no shuffle)
 7.  Quicksort (3-way, no shuffle)
 8.  Heapsort
 9.  Knuth shuffle
10.  Sorted
```

0 7 4 6 2 5 1 8 3 9 10

Your Answer		Score	Explanation
0	✓	0.09	
7	✓	0.09	
4	✓	0.09	
6	✓	0.09	
2	✓	0.09	
5	✓	0.09	
1	✓	0.09	
8	✗	0.00	
3	✓	0.09	
9	✗	0.00	
10	✓	0.09	
Total		0.82 / 1.00	

Question Explanation

The correct answer is:

```
0 7 4 6 2 5 1 9 3 8 10
```

```
 0: Original input
 7: Quicksort (3-way, no shuffle) after first partitioning step
 4: Mergesort (top-down) just before second-to-last call to merge
 6: Quicksort (standard, no shuffle) after first partitioning step
 2: Insertion sort after 16 iterations
 5: Mergesort (bottom-up) after merging sorted subarrays of size 1 to form sorted
subarrays of size 2
 1: Selection sort after 12 iterations
 9: Knuth shuffle after 12 iterations
 3: Shellsort (with 3x + 1 increment sequence) after 3-sorting phase
 8: Heapsort after heap construction phase
10: Sorted
```

Question 13

```
(seed = 231636)
What is the order of growth of the expected running time when running the sorting
algorithms
in the left-hand column on an input of size N whose keys are all equal?

You may use each number once, more than once, or not at all.

___ selection sort                  0. 1

___ mergesort (bottom-up)           1. log N

___ mergesort (top-down)            2. N

___ heapsort                        3. N log N

                                    4. N^2
```

4 3 3 3

Your Answer		Score	Explanation
4	✓	0.25	
3	✓	0.25	
3	✓	0.25	
3	✓	0.25	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
4 3 3 3
```

Question 14

```
(seed = 63145)
Give the level order traversal of the BST that results after inserting
the following sequence of keys into an initially empty BST:

    55 84 64 65 33 10 59 11 70 43
```

55 33 84 10 43 64 11 59 65 70

Your Answer	Score	Explanation

55 33 84 10 43 64 11 59 65 70	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
55 33 84 10 43 64 11 59 65 70
```

```
Here is the level order traversal of the BST after each insertion:

55:  55
84:  55 84
64:  55 84 64
65:  55 84 64 65
33:  55 33 84 64 65
10:  55 33 84 10 64 65
59:  55 33 84 10 64 59 65
11:  55 33 84 10 64 11 59 65
70:  55 33 84 10 64 11 59 65 70
43:  55 33 84 10 43 64 11 59 65 70
```

Question 15

```
(seed = 119534)
Consider the left-leaning red-black BST whose level order traversal is

    37 21 88 13 36 59 89 29 41 87          ( red links = 59 29 )

What is the level order traversal of the red-black BST that results after
inserting the following sequence of keys:

    72 94 79
```

```
79 37 88 21 59 87 94 13 36 41 72 89 29
```

Your Answer		Score	Explanation
79 37 88 21 59 87 94 13 36 41 72 89 29	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
79 37 88 21 59 87 94 13 36 41 72 89 29
```

```
Here is the level order traversal of the red-black BST after each insertion:

    37 21 88 13 36 59 89 29 41 87          ( red links = 59 29 )
72: 37 21 88 13 36 59 89 29 41 87 72       ( red links = 59 29 72 )
94: 37 21 88 13 36 59 94 29 41 87 89 72    ( red links = 59 29 89 72 )
79: 79 37 88 21 59 87 94 13 36 41 72 89 29 ( red links = 37 89 29 )
```

Question 16

```
(seed = 81364)
Suppose that you insert the following sequence of points into an initially empty kd-
tree.
Give the level order traversal of the resulting kd-tree.

A (0.83, 0.21)
B (0.35, 0.25)
C (0.14, 0.53)
D (0.73, 0.05)
E (0.24, 0.16)
F (0.57, 0.32)
G (0.89, 0.77)
H (0.81, 0.59)

Recall that our convention is to subdivide the region using the x-coordinate at even
levels
(including the root) and using the y-coordinate at odd levels. Also, we use the left
subtree
for points with smaller x- or y-coordinates.
```

A B G D C E F H

Your Answer		Score	Explanation
A B G D C E F H	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
A B G D C E F H
```

```
Here is the level order traversal of the kd-tree after each insertion:

A:  A
B:  A B
C:  A B C
D:  A B D C
E:  A B D C E
F:  A B D C E F
G:  A B G D C E F
H:  A B G D C E F H
```

Question 17

```
(seed = 315015)
For each quantity in the left column, give the best-matching description from the right
column.
You may use each number once, more than once, or not at all.

___ Max height of a BST with N keys                                  0. 0

___ Max function-call stack size when (top-down) mergesorting N keys    1. ~ 1
___ Min height of an (unweighted) quick union tree with N items          2. ~
1/2 lg N

___ Max height of a weighted quick union tree with N items               3. ~
log_3 N

___ Max height of an (unweighted) quick union tree with N items          4. ~ ln
N

___ Max height of a left-leaning red-black BST with N keys               5. ~ lg
N

                                                                         6. ~ 2
lg N

                                                                         7. ~ 2
ln N

                                                                         8. ~ N
```

8 5 1 5 8 5

Your Answer		Score	Explanation
8	✓	0.17	
5	✓	0.17	
1	✗	0.00	
5	✓	0.17	
8	✓	0.17	
5	✗	0.00	
Total		0.67 / 1.00	

Question Explanation

The correct answer is:

```
8 5 0 5 8 6
```

Question 18

```
(seed = 964548)
Give the array that results after inserting the following sequence of 10 keys
into an initially empty linear probing hash table.

key  hash
---  ----
 R    2
 Q    1
 A    5
 O    9
```

```
X    8
F    0
G    1
C    7
E    9
L    6

Assume that the size of the hash table is 10 and that it does not grow or shrink.
```

F Q R G E A L C X O

Your Answer		Score	Explanation
F Q R G E A L C X O	✓	1.00	
Total		1.00 / 1.00	

Question Explanation

The correct answer is:

```
F Q R G E A L C X O
```

```
Here is the array immediately after each insertion:

R:  - - R - - - - - - -
Q:  - Q R - - - - - - -
A:  - Q R - - A - - - -
O:  - Q R - - A - - - O
X:  - Q R - - A - - X O
F:  F Q R - - A - - X O
G:  F Q R G - A - - X O
C:  F Q R G - A - C X O
E:  F Q R G E A - C X O
L:  F Q R G E A L C X O
```

Question 19

```
(seed = 508757)
Suppose that you have a data type for a set of N items (no duplicate keys) and that it
is represented internally using an unordered array. Assume that the data type is
implemented in an efficient and natural manner given the specified representation.

Match up each of the following operations with their amortized running times.
You may use each number once, more than once, or not at all.

___ insert a key into the set                  0. 1

___ number of keys <= a specified key          1. log N

___ smallest key >= a specified key            2. N

___ number of keys between lo and hi           3. N log N

___ is a key in the set?                       4. N^2

___ delete the minimum key
```

0 0 2 2 2 2

Your Answer		Score	Explanation
0	✗	0.00	
0	✗	0.00	
2	✓	0.17	
2	✓	0.17	
2	✓	0.17	
2	✓	0.17	
Total		0.67 / 1.00	

Question Explanation

The correct answer is:

2 2 2 2 2 2

Question 20

(seed = 211332)
You are applying for a job at a new software technology company. Your interviewer asks you
to identify the following tasks as either possible (with algorithms and data structures learned
in this course), impossible, or an open research problem.

You may use each number once, more than once, or not at all.

0. Possible

1. Impossible

2. Open

___ Implement a union-find data type so that all operations (except construction) take constant time in the worst case.

___ Implement a union-find data type so that all operations (except construction) take logarithmic time in the worst case.

___ Build a left-leaning red-black BST containing N keys using ~ 8N compares (where the array of keys are given to you in ascending order).

___ Design a compare-based sorting algorithm that guarantees to sort any array of N items using ~ 9/10 N lg N compares in the worst case.

___ Design a compare-based sorting algorithm that guarantees to sort any of N items with

5 distinct keys in linear time.

___ Output the keys in an (unbalanced) BST in sorted order in linear time.

___ Find the smallest key greater than or equal to a given key in a left-leaning red-black BST in logarithmic time.

___ Perform a left rotation in a BST in constant time.

0 0 0 1 1 0 0 0

Your Answer		Score	Explanation
0	✗	0.00	
0	✓	0.12	
0	✓	0.12	
1	✓	0.12	
1	✗	0.00	
0	✓	0.12	
0	✓	0.12	
0	✓	0.12	
Total		0.75 / 1.00	

Question Explanation

The correct answer is:

1 0 0 1 0 0 0 0

<pre style="background-color:transparent;">Implement a union-find data type so that all operations (except construction) take constant time in the worst case.
Impossible: inverse Ackermann lower bound in cell-probe model of computation

Implement a union-find data type so that all operations (except construction) take logarithmic time in the worst case.
Possible: weighted quick union

Build a left-leaning red-black BST containing N keys using ~ 8N compares (where the array of keys are given to you in ascending order).
Possible: can do without any compares

Design a compare-based sorting algorithm that guarantees to sort any array of N items using ~ 9/10 N lg N compares in the worst case.
Impossible: sorting lower bound

Design a compare-based sorting algorithm that guarantees to sort any of N items with 5 distinct keys in linear time.
Possible: 3-way quicksort

Output the keys in an (unbalanced) BST in sorted order in linear time.
Possible: inorder traversal

```
Find the smallest key greater than or equal to a given key in a left-leaning red-black
BST in logarithmic time.
Possible: ceiling algorithm discussed in BST lecture

Perform a left rotation in a BST in constant time.
Possible: it changes only a constant number of pointers

</pre>
```