Cracking the Coding Interview v5 Errata & Changes

All (Sorted by Date) All (Sorted by Problem) Typos Minor Clarification Enhancement Correction HOW TO REPORT ISSUES Sheet11

Type	Date	Reported By	Chapter / Problem	Chapter Name	Page	Description
Spelling / Grammar / Typo	9/29/2011	Itai	2.6	Linked Lists	194	should say B instead of b in: "when A moves q steps away from B, it is also moving q steps closer to b"
Minor	9/29/2011	Itai	9	Recursion and Dynamic Programming	109	Should say "All recursive code can be implemented iteratively" instead of "All recursive code can be implemented re
Clarification	9/29/2011	Itai	2.2	Linked Lists	185	Added note at beginning of problem for definition of k: "Note that for this solution, we have defined k such that passi would return the last element, k = 2 would return to the second to last element, and so on. It is equally acceptable to that k = 0 would return the last element."
Enhancement	9/29/2011	Itai	2.2	Linked Lists	187	At beginning of function, added bounded check on k: if (k <= 0) return null;

Enhancement	9/29/2011	Itai	3	Stacks and Queues	79	Line 19: for consistency, changed to Object peek() {
						return top.data;
						}

Minor	9/29/2011	Itai	11	Sorting and Searching	121	final block should say solution = doc5 instead of solution = doc
Minor	9/29/2011	Itai	2.7	Linked Lists	200	Lines 2, 5, and 7 should say Result instead of Question.Result
Minor	9/29/2011	Itai	4	Trees and Graphs	85	should say "the key thing to remember is the use of the queue" instead of "the key thing to remember is the use of t
Minor	9/29/2011	Momchil	5.7	Bit Manipulation	255	should say "in lines 24 and 27" instead of "in lines 23 and 25"
Spelling / Grammar / Typo	9/29/2011	Momchil	7	Mathematics and Probability	264	P(missing 1, and making 1 and 3) should say " P(missing 1, and making 2 and 3)
Spelling / Grammar / Typo	9/29/2011	Momchil	7	Mathematics and Probability	264	3 should be p in "p * p * (1 - p) + p * (1 - p) * 3 + (1 - p) * p * p
Spelling / Grammar / Typo	9/29/2011	Gayle	7	Mathematics and Probability	320	Removed "the" from "Since A[5] = 2, we know that [the] A[4]"
Spelling / Grammar / Typo	9/29/2011	Momchil	7	Mathematics and Probability	320	Array diagram should have A[5] equal to 3 instead of 2
Spelling / Grammar / Typo	9/29/2011	Gayle	7	Mathematics and Probability	321	should say "subsets" in "Write a method to return all subset of a set"
Spelling / Grammar / Typo	9/29/2011	Momchil	7	Mathematics and Probability	322	should say "or just P(n)" instead of "or just P(3)"
Enhancement	9/29/2011	Momchil	9	Recursion and Dynamic Programming	332	Removed line 12 which cleared queen. Line wasn't strictly necessary, as noted in the code attachment. It's somewha up the board when you're done, but this line probably added more confusion than it was worth.
Enhancement	9/29/2011	Momchil	10	Scalability and Memory Limits	356	Added additional explanation before tree: "In the below example, the value in parentheses indicates the number of no subtree (or, in other words, the rank of the node relative to its subtree)."
Minor	9/29/2011	Momchil	10.6	Scalability and Memory Limits	353	Changed from "If each URL is an average of 100 characters, and each character is 4 bytes, then this list of 100 billic take up about 400 gigabytes" to "If each URL is an average of 100 characters, and each character is 4 bytes, then tl billion URLs will take up about 4 terabytes."
Spelling / Grammar / Typo	9/29/2011	Momchil	16.2	Threads and Locks	417	Corrected - "accomplish" should be "accomplished"
Spelling / Grammar / Typo	9/29/2011	Andreas	1.5	Arrays and Strings	73	In problem description "a2b1c8a3" should say "a2b1c5a3". (Solution description is correct.)
Spelling / Grammar / Typo	9/29/2011	Vinit	1.7	Arrays and Strings	181	Extra set of parens on code line 18. Should be "if (row[i] || column[j])"
Clarification	9/29/2011	Mark	Technical Questions	Technical Questions	52	In example for Approach II: Pattern Matching, added note that the array has all unique elements.
Spelling / Grammar / Typo	9/29/2011	Createspace	4.7	Trees and Graphs	232	Tree is somewhat cut off on page
Correction	9/29/2011	Vikas	5	Bit Manipulation	89	Answer of row 4, column 3 should read 1000 instead of 0011. (Solution described in below bullets is still correct.)
Clarification	9/29/2011	Curious Cat	10.3	Scalability and Memory Limits	115	Removed constraint to implement code in O(log n) time, since that's not possible in certain cases. Also added clarifi code about runtime. "This code will run in O(log n) if all the elements are unique. However, with many duplicates, the actually O(n). This is because with many duplicates, we will often have to search both the left and right sides of the subarrays)."
Minor	9/29/2011	Gayle	17.6	Moderate	441	Code crashed if array was already sorted. Added line after line 39 to check for this condition.: if (min_index >= array. Already sorted
Enhancement	10/2/2011	Gayle	7.7	Mathematics and Probability	275	Changed code in first solution for consistency with second solution. (1) Line 21 changed to "if k < 0". (2) for loop in li to start from 0 instead of 1 (3) comment in line 25 removed.
Minor	10/5/2011	Gayle	9	Recursion and Dynamic Programming	108	2nd line after "Simple Example of Dynamic Programming: Fibonacci Numbers" says "prime" instead of "Fibonacci"

Minor	10/10/2011	V.	3	Stacks and Queues	206	In example, 5 should be pushed first, then 6. It should say:
						push(5); // stack is {5}, min is 5
						push(6); // stack is {6, 5}, min is 5
						push(3); // stack is {3, 6, 5}, min is 3

Enhancement	10/19/2011	Gayle	9.1	Recursion and Dynamic Programming	109	Code can be cleaned up slightly to use just an int array instead of a hash table.
						public static int countWaysDP(int n, int[] map) {
						if (n < 0) {
						return 0;
						} else if (n == 0) {
						return 1;
						} else if (map[n] > -1) {
						return map[n];
						} else {
						map[n] = countWaysDP(n - 1, map) +
						countWaysDP(n - 2, map) +
						countWaysDP(n - 3, map);
						return map[n];
						}
						}

Spelling / Grammar / Typo	10/21/2011	Gayle	Acknowledgements	Acknowledgements	491	"write" to "writing"
Spelling / Grammar / Typo	10/21/2011	Gayle	Acknowledgements	Acknowledgements	491	"exception" to "exceptional"
Spelling / Grammar / Typo	10/21/2011	Gayle	Acknowledgements	Acknowledgements	491	removed "... especially considering you had none to me..."
Minor	10/25/2011	Gayle	9.9	Recursion and Dynamic Programming	331	Diagram depicted incorrect solution. Queen at (7, 4) should be at (7, 5).
Minor	10/28/2011	Richard	5	Bit Manipulation	90	Description should read "An operation like x & (~0 << n) clears the n rightmost" not "An operation like x && (~0 << n rightmost"
Minor	10/31/2011	I-Gene	4	Trees and Graphs	83	Should say "a tree must have exactly 2^n - 1 nodes to meet this condition" not "a tree must have exactly 2^n nodes condition"
Minor	11/2/2011	Cihat	9.1	Recursion and Dynamic Programming	316	It should say "the runtime of this algorithm is exponential (specifically, O(3^N))" not "the runtime of this algorithm is (specifically, O(N^3))"
Minor	11/8/2011	Fin	5.3	Bit Manipulation	249	the equations after 'This math reduces' should have 2^{c0} and 2^{c1-1} instead of 2c0 and 2c1-1.
Spelling / Grammar / Typo	11/10/2011	John	Introduction	Introduction	3	"...of these questions. My first to show you what they're like, but I have chosen to allocate space where there's more should be "...The book will briefly touch on some of these questions to show you what they're like, but I have choser space where there's more to learn."
Correction	11/19/2011	KW	1.5	Arrays and Strings	178	In line 42 at the TOP of the page, "count = 0" should read "count = 1".
Minor	11/19/2011	KW	2.2	Linked Lists	186	In line 10 on Approach C, the line should say nthToLastR2(head.next, k, i) instead of nthToLastR2(head.next, n, i)
Minor	11/23/2011	KW	2.7	Linked Lists	197	In page 197, the last paragraph of Solution #1 should say "If the first half of normal list matches the FIRST half of th then the second half of the normal list must match the SECOND half of the reversed list." instead of "If the first half matches the second half of the reversed list, then the second half of the normal list must match the first half of the r
Minor	11/25/2011	KW	3	Stacks and Queues	80	In page 80, line 5 of the class Queue should say "if (first == null) {" instead of "if (!first) {"
Correction	11/25/2011	KW	3.1	Stacks and Queues	202	Line 3 should be "static int [] stackPointer = {-1, -1, -1}" instead of "static int [] stackPointer = {0, 0, 0}". This is beca stackPointer represents the current top element, not the next element to be added.
Correction	11/25/2011	KW	3.1	Stacks and Queues	203	should say "if (start <= index && index < start + capacity) {" instead of "if (start <= index && index <= start + capac

Enhancement	11/25/2011	KW	3.1	Stacks and Queues	204	added in code for numberOfElements(): public static int numberOfElements() {
						return stacks[0].size + stacks[1].size + stacks[2].size;
						}

Enhancement	12/3/2011	Gayle	3.1	Stacks and Queues	202	created method absTopOfStack to be called by push, pop, and peek

Enhancement	12/3/2011	Gayle	3.1	Stacks and Queues	202	reorganized end of push: /* Increment stack pointer and then update top value*/
						stackPointer[stackNum]++;
						buffer[absTopOfStack(stackNum)] = value;

Enhancement	12/3/2011	Gayle	3.1	Stacks and Queues	202	reorganized end of pop: int value = buffer[absTopOfStack(stackNum)]; // Get top
						buffer[absTopOfStack(stackNum)] = 0; // Clear index
						stackPointer[stackNum]--; // Decrement pointer
						return value;

Enhancement	12/3/2011	Gayle	3.1	Stacks and Queues	204	In Approach 2, changed lines 15 and 21 to "non-wrapping case" and "wrapping case"

Correction	12/3/2011	KW	3.1	Stacks and Queues	204	Bug-fix for wrapping case, and code clean-up: public boolean isWithinStack(int index, int total_size) {
						// Note: if stack wraps, the head (right side) wraps around to the left.
						if (start <= index && index < start + capacity) {
						// non-wrapping, or "head" (right side) of wrapping case
						return true;
						} else if (start + capacity > total_size &&
						index < (start + capacity) % total_size) {
						// tail (left side) of wrapping case
						return true;
						}
						return false;
						}

						In the middle of page 253, right above the second table, the equations should be 'count_2(0s)=count_2(1s) OR

Minor	12/3/2011	LY	5.7	Bit Manipulation	253	count_2(0s)=1+count_2(1s)' instead of 'count_2(0s)=1+count_2(1s) OR count_2(0s)=1+count_2(1s)'.
Spelling / Grammar / Typo	12/3/2011	LY	16.5	Threads and Locks	426	On line 37 of page 426, the comment should be 'wait until finished with second()' instead of 'wait until finished with th
Spelling / Grammar / Typo	12/3/2011	KW	3.6	Stacks and Queues	215	On first paragraph on 2.6, "maximum" should be changed to "minimum" (3 occurrences). Order of items in picture sh reversed.
Minor	12/3/2011	KW	3.7	Stacks and Queues	218	In page 218, line 54 and 58 of code should call the method poll() instead of getFirst() for LinkedList, since getFirst() c but does not remove the first element of the list.
Enhancement	12/3/2011	Gayle	3.7	Stacks and Queues	218	changed lines 48 and 50 to call dequeueDogs and dequeueCats
Enhancement	12/3/2011	KW	3.6	Stacks and Queues	81	Updated beginning of problem to clarify what ascending means: "Write a program to sort a stack in ascending order (items on top)..."
Correction	12/5/2011	Siege	4	Trees and Graphs	85	On p85, there is pseudocode for DFS. The first test (i.e: testing the Node parameter) should be == instead of !=.
Minor	12/12/2011	Sathish	3	Stacks and Queues	80	dequeue() in Queue class should not take in any parameters
Spelling / Grammar / Typo	12/12/2011	KW	4.2	Trees and Graphs	222	In page 222, line 6 of the code should say "// operates as Queue" instead of "// operates as Stack".
Spelling / Grammar / Typo	12/12/2011	KW	4.2	Trees and Graphs	222	In page 222, line 16 of the code should say "u = q.removeFirst(); // i.e., dequeue()" instead of "u = q.removeFirst(); //
Spelling / Grammar / Typo	12/12/2011	KW	4	Trees and Graphs	85	In the middle of page 85, should say "An iterative solution involving a QUEUE usually works best." instead of "An ite involving a stack usually works best."
Correction	12/14/2011	JC	17.5	Moderate	439	Added additional condition on if-statement on line 49: "guess.charAt(i) != solution.charAt(i)". Corrected fix for cases l GYYG)
Correction	12/14/2011	KW	4.7	Trees and Graphs	231	Corrected issue where Solution #2 doesn't correctly handle the case where p is in the right subtree of q (or q is in the p). Added these lines to the top of the commonAncestorHelper function: if (root == null) return null; if (root == p || root == q) return root;
Spelling / Grammar / Typo	12/14/2011	KW	4.8	Trees and Graphs	234	In page 234 / 235, the second paragraph should say "..., then T2 is a subtree of T1." instead of "..., then T2 is a sub
Spelling / Grammar / Typo	12/16/2011	Siege	5	Bit Manipulation	91	The second sentence of the Update Bit discussion should be "Then, we shift the intended value, v, LEFT by i bits."
Enhancement	12/16/2011	Siege	5	Bit Manipulation	91	Removed unnecessary "public static" from clearBitsIthrough0() definition
Enhancement	12/18/2011	Siege	5.7	Bit Manipulation	92	Changed from "An array A[1...n] contains all the integers from 0 to n," to "An array A contains all the integers from 0
Minor	12/18/2011	Siege	7.4	Mathematics and Probability	268	On p268, the fourth paragraph under Division should end: ..."will equal x" (instead of "will equal a")
Spelling / Grammar / Typo	12/20/2011	HXP	10, 11	10, 11		Chapter numbers in solutions are reversed
Enhancement	12/20/2011	HXP	8.2	Object-Oriented Design	284	Changed CallHandler constructor from public to protected
Minor	12/24/2011	KW	5	Bit Manipulation	91	In page 91, the line 2 of code clearBitsMSBthroughI() should be "int mask = (1 << i) - 1;" instead of "int mask = (1 < since the bit i should also be cleared by requirement.
Spelling / Grammar / Typo	12/24/2011	KW	5.2	Bit Manipulation	92	Changed wording of problem to "... in binary with at most 32 characters."
Minor	12/24/2011	KW	5.2	Bit Manipulation	243	Changed "if (binary.length() > 32) {" to "if (binary.length() >= 32) {"
Spelling / Grammar / Typo	12/24/2011	KW	5.3	Bit Manipulation	246	comment should say "a = 1 << (c1 - 1); // 0s with a 1 at position c1 - 1" instead of "a = 1 << (c1 - 1); // 0s with a 1 a
Spelling / Grammar / Typo	12/24/2011	KW	5.3	Bit Manipulation	249	added subscript notation on some characters
Spelling / Grammar / Typo	12/24/2011	KW	5.3	Bit Manipulation	249	n page 249, the 3 comments after formulas for "Arithmetic Approach to Get Previous Number" should say "n" instea "// Removes trailing 1s. n is now 10000000.", "// Flips trailing 0s. n is now 01111111.", "// Flips last (c0-1) 0s. n is now 01110000.".
Spelling / Grammar / Typo	12/25/2011	KW	5.7	Bit Manipulation	253	In page 253, the conclusion should use subscript instead of "count2(1s)". That is, "If count_2(0s) <= count_2(1s), then LSB_2(v) = 0." "If count_2(0s) > count_2(1s), then LSB_2(v) = 1."
Spelling / Grammar / Typo	12/25/2011	KW	5.7	Bit Manipulation	253	"bfit" should be "bit"
Spelling / Grammar / Typo	12/25/2011	KW	5.7	Bit Manipulation	254	In page 254, should use subscript "..., so LSB_i(v) = 0." instead of "..., so LSBi(v) = 0.".
Correction	1/3/2012	KW	5.8	Bit Manipulation	256	the line 26 of code should be "screen[(width / 8) * y + (x1 / 8)] |= mask;" instead of "screen[(width / 8) * y + first_full_ mask;", since the expression (first_full_byte - 1) will NOT be the same as (x1 / 8) when (start_offset == 0), which ca one error.
Spelling / Grammar / Typo	1/7/2012	KW	7.6	Mathematics and Probability	271	In page 271, the first paragraph of solution 7.6 should be "..., since there are N^2 line segments and we need to ..." i since there are N^2 points and we need to ...".
Minor	1/7/2012	KW	7.7	Mathematics and Probability	274	In page 274, missing power "b" for 5. That should be "3^(a-1) * 5^b * 7^c" instead of "3^(a-1) * 5 * 7^c"
Minor	1/7/2012	KW	8	Object-Oriented Design	105	In page 105, the line 2 of code for class Restaurant should be "private static Restaurant _instance = null;" instead of Restaurant _instance = null;", which leads to the compile error "non-static variable instance cannot be referenced fro context".

Enhancement	1/7/2012	KW	7.5	Mathematics and Probability	270	Before code, added following clarification: "In the below code, we will assume the origin (0, 0) is in the upper left-han
Spelling / Grammar / Typo	1/17/2012	KW	9.5	Recursion and Dynamic Programming	324	In page 324, the solution should say "We solve for f(n-1), and then push a_n into every spot in each of these strings. "We solve for f(n-1), and then push a_3 into every spot in each of these strings."
Spelling / Grammar / Typo	1/17/2012	KW	9.6	Recursion and Dynamic Programming	327	In page 327, the line 11 of code should be "str[count] = ‘(’;" instead of "str[count] = ‘(‘;". That is, should use right sing printing.
Spelling / Grammar / Typo	1/17/2012	KW	9.9	Recursion and Dynamic Programming	331	In page 331, it should be "... with queen at (7, 7)" instead of "... with queen at (7, 7) +". That is, the equation has a ta plus sign.

Spelling / Grammar / Typo	1/17/2012	Maxim	9.2	Recursion and Dynamic Programming	318	isFree(x-1,y) // "Try right" -> Actually you're checking left cell, should be "Try left"
						
						isFree(x,y-1) // "Try down" -> Actually you're checking up cell, should be "Try up"

Spelling / Grammar / Typo	1/20/2012	Susheel	1	Arrays and Strings	74	In the "Additionally Questions" section, some chapter numbers are incorrect (the chapter names are correct). Specifi Oriented Design(#7.10) and Same Sorting and Searching (#9.6)
Spelling / Grammar / Typo	1/20/2012	KW	11.1	Sorting and Searching	360	In page 360, the line 2-3 of code for solution 11.1 should swap comments: int indexA = lastA - 1; /* Index of last element in array a */ int indexB = lastB - 1; /* Index of last element in array b */
Spelling / Grammar / Typo	1/20/2012	RN	Behind the Scenes	Behind the Scenes	18	List of companies was missing Facebook (also changed order): "Microsoft, Amazon, Google, Apple, Facebook, and
Spelling / Grammar / Typo	1/20/2012	KW	11.6	Sorting and Searching	122	the problem description should add a clarification "... which each row and each column is sorted IN ASCENDING OR same as it's reiterated in page 365.
Spelling / Grammar / Typo	1/20/2012	KW	11.6	Sorting and Searching	366	it should say "... , since the first element of each column must increase in size from LEFT to RIGHT." instead of "... element of each column must increase in size from right to left."
Correction	1/20/2012	KW	11.5	Sorting and Searching	365	the method searchR() is missing the base case handling for the "not found" scenario (worked in some cases, but not could result in infinite loop. One possible fix is to add the below code at the very beginning: "if (first > last) return -1;"
Spelling / Grammar / Typo	1/21/2012	KW	11.6	Sorting and Searching	367	minor typo (and re-worded line slightly). "j" should read "j-1" in the following line of solution #2. Changed to: "We are t row and column is sorted. This means that element a[i][j] will be greater than the elements in row i between columns the elements in column j between rows 0 and i - 1." instead of "We are told that every row and every column is sorte that if we have an element a[i][j], this element is greater than the elements in row i between columns 0 and j and the column j between 0 and i - 1."
Spelling / Grammar / Typo	1/22/2012	Arun	1	Arrays and Strings	72	should be "The total time therefore is O(x + 2x + … + nx)" instead of The total time therefore is O(x + 2x + … + x)
Enhancement	1/22/2012	Gayle	1	Arrays and Strings	72	Added line at end of paragraph: This reduces to O(xn^2). (Why isn’t it O(xn^n)? Because 1 + 2 + … + n equals n(n+1
Correction	1/24/2012	KW	11.7	Sorting and Searching	373	for loop on line 38 should read "for (int i = 0; i < array.size(); i++) {" instead of "for (int i = 1; i < array.size(); i++) {"

Minor	1/26/2012	SV	5.3	Bit Manipulation	248	Line 5 of getPrev(int n):
						while (((temp & 1) == 1) && (temp != 0)) {
						Second part of if-statement is unnecessary.

Minor	2/1/2012	DC	3	Stacks and Queues	79	line 6 should say "Object item = top.data;" instead of "Node item = top.data;"
Minor	2/1/2012	RH	1.5	Arrays and Strings	178	removed str parameter from setChar (lines 17, 24, 28)
Enhancement	2/1/2012	Gayle	7	Mathematics and Probability	97	Should say "As you probably know, every positive integer" instead of "As you probably know, every number"
Spelling / Grammar / Typo	2/1/2012	KW	10.3	Scalability and Memory Limits	349	In page 367, the line 23 of code should read "in = new Scanner(new FileReader("file.txt"));" instead of "in = new Scan FileReader("input_file.txt"));", to ensure keep the same input file as stated in the line 8.
Minor	2/1/2012	KW	10.3	Scalability and Memory Limits	347	Problem should say "four billion non-negative integers" instead of "four billion integers"
Correction	2/1/2012	KW	10.3	Scalability and Memory Limits	347	Size of array is incorrect. Should say instead: long numberOfInts = ((long) Integer.MAX_VALUE) + 1; byte[] bitfield2 = new byte [(int) (numberOfInts / 8)];"
Minor	2/1/2012	JM	13	C and C++	138	should say "Performing p++ will skip ahead by sizeof(int) bytes" instead of "Performing p++ will skip ahead by four b for 64 bit integers
Spelling / Grammar / Typo	2/2/2012	SV	8	Object-Oriented Design	104	Server and Host inherit Employee should say Server and Host inherit from Employee
Enhancement	2/6/2012	Gayle	9.2	Recursion and Dynamic Programming	318	Instead of "To find all paths to (X,Y), we find all paths to (X-1,Y) and (X,Y-1)", say "To find a path to (X,Y), we look f adjacent coordinate: (X-1,Y) or (X,Y-1). Of course, if one of those squares is off limits, we ignore it. "
Enhancement	2/6/2012	Gayle	6.6	Brain Teasers	262	Added clarification to problem. Changed description to "There are 100 closed lockers in a hallway. A man begins by lockers. Next, he closes every second locker. Then, on his third pass, he toggles every third locker (closes it if it is if it is closed). This process continues for 100 passes, such that on each pass i, the man toggles every ith locker. A pass in the hallway, in which he toggles only locker #100, how many lockers are open?"
Spelling / Grammar / Typo	2/6/2012	Gayle	6.5	Brain Teasers	262	Changed "We go to floor 14, then 27, then 39, …. This takes 14 steps in the worse case. As in many other maximizi problems, the key in this problem is “worse case balancing.”" to "We go to floor 14, then 27, then 39, and so on. This in the worst case. As in many other maximizing / minimizing problems, the key in this problem is “worst case balanc
Enhancement	2/6/2012	Gayle	6.4	Brain Teasers	260	Changed "The two blue-eyed people see each other and are unsure whether c = 1 or c = 2." to "The two blue-eyed pe other, but be unsure whether c is 1 or 2."
Enhancement	2/6/2012	Gayle	6.4	Brain Teasers	260	Changed "Assuming all the people are intelligent, the person (the only person) with the blue eyes" to "Assuming all th intelligent, the blue-eyed person"
						On page 246, a = a << pos should be 'a = a << n'

Spelling / Grammar / Typo	2/15/2012	RH	5.3	Bit Manipulation	246	On page 246, a = a << pos should be 'a = a << p' Also on page 246, n &= ~((1 << pos) -1) should be 'n &= ~((1 << p) -1)' On page 247, the comment in the second last line should be '// Sequence of 1s followed by p + 1 zeros.'
Spelling / Grammar / Typo	2/15/2012	RH	14.4	Java	401	In the re-written code to Question 14.4, Line 3: String str = (String) sv.get(0) should be "String str = (String) vector.ge
Spelling / Grammar / Typo	2/15/2012	RH	13.1	C and C++	386	In the second paragraph of the solution, the sentence " So, initially, our array has lines 1 through K, then 1 through K be " So, initially, our array has lines 0 through K, then 1 through K+1 ..."
Spelling / Grammar / Typo	2/15/2012	RH	14.4	Java	402	On page 402, Line 22: int f2 = fool ->val; should be "int f2 = foo2 -> val;".
Spelling / Grammar / Typo	2/15/2012	RH	17.11	Moderate	448	On page 448, in the Chapter ' First Attempt (Fixed Number of Calls), in the first paragraph, "we might try to generate between 0 and 9, and then mod the resulting value by 10", this should be 'then mode the resulting value by 7'.
Spelling / Grammar / Typo	2/15/2012	RH	17.11	Moderate	448	in the third paragraph, " you'll note that this rand7() function will return 0 with 5/25th probability but return 0 with just 3 probability", this should be " you'll note that this rand7() function will return 4 with 5/25th probability but return 0 with j probability"
Spelling / Grammar / Typo	2/17/2012	KW	13	C and C++	137	In page 137, the code for "Operator Overloading" should be "BookShelf BookShelf::operator+(BookShelf &other) { ... "BookShelf BookShelf::operator+(Packet &other) { ... }".

Correction	2/17/2012	DC	13.8	C and C++	394	Correction for issue: if you assign a SmartPointer to itself, old code would will end up decreasing the reference count when it should stay the same. This will endup freeing the ref_count integer, which will be dereferenced later on by the again, causing an access fault.
						
						Corrected by changing function to:
						SmartPointer<T> & operator=(SmartPointer<T> & sptr) {
						if (this == &sptr) return *this;
						
						/* If already assigned to an object, remove one reference. */
						if (*ref_count > 0) {
						remove();
						}
						
						ref = sptr.ref;
						ref_count = sptr.ref_count;
						++(*ref_count);
						return *this;
						}

Spelling / Grammar / Typo	2/25/2012	KW	13.9	C and C++	395	it should read "By doing (p1 + 16) & 11..10000, we are moving q back to a memory address divisible by 16. Doing ar last four bits of the memory address with 0000 guarantees us that this new value will be divisible by 16." instead of " 16) & 11..1000, we are moving q back to a memory address divisible by 16. Doing an AND of the last three bits of th address with 000 guarantees us that this new value will be divisible by 16."
Enhancement	2/25/2012	SV	8.2	Object-Oriented Design	283	Changed "ArrayList<Employee>[] employeeLevels" to "List<List<Employee>> employeeLevels;" and made other ne changes
Correction	3/10/2012	KW	13.4	C and C++	389	(1) line 11 of the code should read "strcpy(dest.ptr, src.ptr);" instead of "memcpy(dest.ptr, src.ptr);", since memcpy() "size_t num" as the 3rd parameter. (2) the line 10 of code should be "dest.ptr = (char *)malloc(strlen(src.ptr) + 1);" in "dest.ptr = malloc(strlen(src.ptr) + 1);", otherwise the compile error "invalid conversion from ‘void*’ to ‘char*’" will occ
Correction	3/10/2012	KW	13.9	C and C++	395	the line 4 of code should be "void* q = (void*) (((size_t)(p) + offset) & ~(alignment - 1));" instead of "void* q = (void*) offset) & ~(alignment - 1);", to avoid compile error.
Spelling / Grammar / Typo	3/10/2012	CM	17.1	Moderate	430	End of third paragraph should say "All that’s left to do is to set a equal to" instead of "All that’s left to do is to set b e
Spelling / Grammar / Typo	3/10/2012	KW	15	Databases	149	for "Query 1: Student Enrollment", all tables named "bookStudents" and "bookStudentCourses" should be corrected "Students" and "StudentCourses".
Correction	3/10/2012	SV	15.2	Databases	409	Fixed bug where query didn't return buildings with no open requests. Changed start of query to SELECT BuildingNam ISNULL(Count, 0) as 'Count' FROM Buildings LEFT JOIN
Spelling / Grammar / Typo	3/10/2012	KW	17.1	Moderate	431	should use subscript as "In line 1, doing the operation p = p_0 ^ q_0 will result in a 0 if p_0 = q_0 and a 1 if p_0 != q "In line 1, doing the operation p = p_0 ^ q_0 will result in a 0 if p_0 = q0 and a 1 if p_0 != q0".
Spelling / Grammar / Typo	3/16/2012	KW	17.14	Moderate	456	it should read "For example, when we first call p(thit), the current character being processed is just the first t." instea example, when we first call p(this), the current character being processed is just the first t."
Spelling / Grammar / Typo	3/16/2012	KW	17.4	Moderate	437	the comment for the line of 5 should be "int sb = sign(b); // if b >= 0, then 1 else 0" instead of "int sb = sign(b); // if a else 0".
Spelling / Grammar / Typo	3/16/2012	KW	18.4	Hard	466	it should read "We can apply almost the exact same logic to see that there are the same number of 2s in the 3rd digi - 63525 as there as in the range 0 - 70000." instead of "We can apply almost the exact same logic to see that there a number of 2s in the 3rd digit in the range 0 - 63525 as there as in the range 0 - 7000."
Spelling / Grammar / Typo	3/16/2012	KW	18.4	Hard	467	In page 467, under "Case digit = 2" it should be "if x[d] = 2: count2sInRangeAtDigit(x, d) = ..." instead of "if x[d] > 2: count2sInRangeAtDigit(x, d) = ...".
Spelling / Grammar / Typo	3/16/2012	KW	18.5	Hard	469	in page 469, it should be "list: {1a, 2a, 4b, 9a, 10b, 15a, 19b, 25a}" instead of "list: {1a, 2a, 4b, 9a, 10b, 15a, 19b, 20
Spelling / Grammar / Typo	3/16/2012	KW	18.4	Hard	466	it should read "This is the same amount as if we were just counting all the 2s in the 3rd digit between 1 and 60000." i is the same amount as if we were just counting all the 2s in the 3rd digit between 1 and 6000."
						the line 17 of code should be "for (int i = start - 1; i >= 0; i--) {" instead of "for (int i = start - 2; i >= 0; i--) {", otherwise one less than the correct answer in some cases. One test case is "int array[] = { 1, 2, 3, 4, 5, 11, 7, 12, 6, 7, 16, 18, 19 };". The expected result is (5, 9) but the origin

Correction	3/16/2012	KW	17.6	Moderate	441	(4, 9).
Enhancement	3/16/2012	MF	15	Databases	149	Added clarification below Query 2: Teacher Class Size to specify that you DO want to double count students. "Imple get a list of all teachers and how many students they each teach. If a teacher teaches the same student in two cour double count the student. Sort the list in descending order of the number of students a teacher teaches."
Spelling / Grammar / Typo	3/28/2012	KW	18.13	Hard	490	the line 7 of code should be "return lookup.containsKey(s);" instead of "return lookup.containsKey(s));".
Spelling / Grammar / Typo	3/28/2012	KW	18.11	Hard	480	changed bottomRight to bottomLeft in lines 28 and 36
Correction	3/28/2012	KW	18.12	Hard	482	the value of the actual cell was missing. the formula should be "Val(x, y) = Val(x - 1, y) + Val(x, y - 1) - Val(x - 1, y - instead of "Val(x, y) = Val(x - 1, y) + Val(y - 1, x) - Val(x - 1, y - 1)".
Enhancement	4/23/2012	SV	9.6	Recursion and Dynamic Programming	326	The check for set.contains is actually unnecessary, since HashSet performs this prior to insertion. Changed line 11 set if it’s not already in there. Note: * HashSet automatically checks for duplicates before * adding, so an explicit check is not necessary. */ set.add(s);
Clarification	4/24/2012	Gayle	4.9	Trees and Graphs	239	Changed first line on page 239 from "What is the time complexity of this algorithm?" to "What is the time complexity algorithm (assuming a balanced binary tree)?"
Correction	4/24/2012	DA	4.9	Trees and Graphs	239	Space complexity is O(log n) instead of O(n log n). Changed last line on page 239 to "The space complexity is O(log algorithm will recurse O(log n) times and the path parameter is only allocated once (at O(log n) space) during this rec
Clarification	4/25/2012	SV	9.11	Recursion and Dynamic Programming	338	Added clarification on what the definition of n is in the Catalan numbers. "...It is given by the Catalan numbers, wher number of operators:"
Correction	4/25/2012	DH	11	Sorting and Searching	120	Corrected line about radix sort's runntime. Change lined to "Radix Sort | Runtime: O(kn) (see below)"
Clarification	4/25/2012	DH	11	Sorting and Searching	120	Changed line from "radix sort has a worst case of O(kn)" to "radix sort has a runtime of O(kn)"
Enhancement	4/28/2012	GM	15.7	Databases	413	Changed the type of CourseEnrollment.Grade from integer to decimal. Deleted line saying, "We will assume that Cou

Correction	4/28/2012	SV	15.7	Databases	413	Fixed bug where code didn't work in case of ties. CHANGED
						------------ 	
						Our SQL query to get the list of honor roll students might look like this:
						SELECT StudentName, GPA
						FROM (
						SELECT top 10 percent Avg(CourseEnrollment.Grade) AS GPA,
						CourseEnrollment.StudentID
						FROM CourseEnrollment
						GROUP BY CourseEnrollment.StudentID
						ORDER BY Avg(CourseEnrollment.Grade)) Honors
						INNER JOIN Students ON Honors.StudentID = Students.StudentID
						------------ 	
						TO
						------------ 	
						Using the Microsoft SQL Server TOP ... PERCENT function, we might (incorrectly) first try a query like this:
						/* Incorrect Code */
						SELECT TOP 10 PERCENT AVG(CourseEnrollment.Grade) AS GPA,
						CourseEnrollment.StudentID
						FROM CourseEnrollment
						GROUP BY CourseEnrollment.StudentID
						ORDER BY AVG(CourseEnrollment.Grade)
						The problem with the above code is that it will return literally the top 10% of rows, when sorted by GPA. Imagine a s there are 100 students, and the top 15 students all have 4.0 GPAs. The above function will only return 10 of those st is not really what we want. In case of a tie, we want to include the students who tied for the top 10% -- even if this m honor roll includes more than 10% of the class.
						To correct this issue, we can build something similar to this query, but instead first get the GPA cut off.
						DECLARE @GPACutOff float;
						SET @GPACutOff = (SELECT min(GPA) as ‘GPAMin’
						FROM (
						SELECT TOP 10 PERCENT AVG(CourseEnrollment.Grade) AS GPA
						FROM CourseEnrollment
						GROUP BY CourseEnrollment.StudentID
						ORDER BY GPA desc)
						Grades);
						Then, once we have @GPACutOff defined, selecting the students with at least this GPA is reasonably straightforwa
						SELECT StudentName, GPA
						FROM (
						SELECT AVG(CourseEnrollment.Grade) AS GPA,
						CourseEnrollment.StudentID
						FROM CourseEnrollment
						GROUP BY CourseEnrollment.StudentID
						HAVING AVG(CourseEnrollment.Grade) >= @GPACutOff) Honors
						INNER JOIN Students ON Honors.StudentID = Student.StudentID

Correction	4/28/2012	Srinivas	10.3	Scalability and Memory Limits	115	Added clarification / correction that the second part assumes that all the values are distinct. "What if you have only memory? Assume that all the values are distinct."

Enhancement	4/28/2012	Srinivas	10.3	Scalability and Memory Limits	348	Updated line from "we know how many values we should find in each block" to "Since all the values are distinct, we values we should find in each block. "
Minor	5/15/2012	RQ	Introduction	Introduction	47	The book lists the exact value of 2^20 as 1,048,536. It should be 1,048,576
Enhancement	6/11/2012	LL	Introduction	Introduction	56	Changed "This structure is problematic because the polynomial could not have terms with negative or non-integer ex "This structure is problematic because it could not support polynomials with negative or non-integer exponents."
Enhancement	6/11/2012	Raj	7.3	Mathematics and Probability	266	Changed "We may recall from grade school that if two lines are not parallel, then they intersect. Thus, to check if tw we just need to check if the slopes are different." to "If two different lines are not parallel, then they intersect. Thus, lines intersect, we just need to check if the slopes are different (or if the lines are identical)."
Spelling / Grammar / Typo	6/11/2012	Raj	17.8	Moderate	444	Changed "runningSum" and "currentSum" to "sum"

Minor	7/11/2012	MF	8.1	Object-Oriented Design	[downloa
					code
					only]
						[Note: this issue is not in the book. It's only in the downloadable code.] card1 and card2 in dealInitial() are never use
						say:
						hand.addCard(card1);
						hand.addCard(card2);

Correction	7/16/2012	Gayle	13	C and C++	133	Inserted "public:" line at line 17

Clarification	7/16/2012	HF	13	C and C++	134	Changed person destructor to below code, to remove implication that this is intended to be the same Person class a
						earlier.
						~Person() {
						delete obj; // free any memory allocated within class
						}

Clarification	7/16/2012	HF	13	C and C++	134	Added lined in first paragraph under "Constructors and Destructors" to clarify that the below code is not the default c "automatically generates one called the Default Constructor. Alternatively, we can define our own constructor."
Enhancement	7/17/2012	JX	16	Threads and Locks	416	Changed third paragraph to "A thread exists within a process and shares the process' resources (including its heap s threads within the same process will share the same heap space. This is very different from processes, which cann access the memory of another process. Each thread still has its own registers and its own stack, but other threads write the heap memory."
Enhancement	7/17/2012	SV	3.6	Stacks and Queues	216	Rewrote solution for additional clarity. Starting from 4th paragraph, solution now reads: -------- Imagine we have the following stacks, where s2 is "sorted" and s1 is not: s1 = [5, 10, 7] s2 = [12, 8, 3, 1] When we pop 5 from s1, we need to find the right place in s2 to insert this number. In this case, the correct place is above 3. How do we get it there? We can do this by popping 5 from s1 and holding it in a temporary variable. Then, 8 over to s1 (by popping them from s2 and pushing them onto s1) and then push 5 onto s2. Step 1 s1 = [10, 7] s2 = [12, 8, 3, 1] tmp = 5 Step 2 s1 = [8, 12, 10, 7] s2 = [3, 1] tmp = 5 Step 3 s1 = [8, 12, 10, 7] s2 = [5, 3, 1] tmp = -- Note that 8 and 12 are still in s1 -- and that's okay! We just repeat the same steps for those two numbers as we did popping off the top of s1 and putting it into the "right place" on s2. (Of course, since 8 and 12 were moved from s2 to because they were larger than 5, the "right place" for these elements will be right on top of 5. We won't need to muck s2's other elements, and the inside of the below while loop will not be run when tmp is 8 or 12.) --------------
Enhancement	7/17/2012	LL	Introduction	Introduction	56	Changed "polynomial" to "expression", since polynomials technically cannot have negative or non-integer exponents
Enhancement	7/25/2012	KH	4.4	Trees and Graphs	224	Changed problem statement from "binary search tree" to "binary tree", since the fact that it's a binary search tree do
Minor	7/25/2012	KH	4.4	Trees and Graphs	224	Changed "in-order traversal" to "pre-order traversal" in following sentence: "We can implement a simple modification traversal algorithm"
Spelling / Grammar / Typo	7/31/2012	KA	Introduction	Introduction	18	Left out "Facebook" on last paragraph
						Fixed issue with hashing. Issue was that two lines which are equivalent may not actually hash to the same value. R as follows: This problem seems quite straightforward at first. And it is -- sort of. We just "draw" an infinite line (that is, not a line segment) between every two points and, using a hashtable, track wh most common. This will take O(N2) time, since there are N2 line segments.

most common. This will take O(N2) time, since there are N2 line segments.
We will represent a line as a slope and y-intercept (as opposed to a pair of points), which allows us to easily check t
from (x1, y1) to (x2, y2) is equivalent to the line from (x3, y3) to (x4, y4).
To find the most common line then, we just iterate through all lines segments, using a hashtable to count the numbe
seen each line. Easy enough!
However, there's one little complication. We're defining two lines to be equal if the lines have the same slope and y-i
are then, furthermore, hashing the lines based on these values (specifically, based on the slope). The problem is tha
numbers cannot always be represented accurately in binary. We resolve this by checking if two floating point numbe
epsilon value of each other.
What does this mean for our hashtable? It means that two lines with "equal" slopes may not be hashed to the same
this, we will round the slope down to the next epsilon and use this flooredSlope as the hash key. Then, to retrieve all
potentially equal, we will search the hashtable at three spots: flooredSlope, flooredSlope - epsilon, and flooredSlope
will ensure that we've checked out all lines that might be equal.

```
Line findBestLine(GraphPoint[] points) {
Line bestLine = null;
int bestCount = 0;
HashMap<Double, ArrayList<Line>> linesBySlope =
new HashMap<Double, ArrayList<Line>>();

for (int i = 0; i < points.length; i++) {
for (int j = i + 1; j < points.length; j++) {
Line line = new Line(points[i], points[j]);
insertLine(linesBySlope, line);
int count = countEquivalentLines(linesBySlope, line);
if (count > bestCount) {
bestLine = line;
bestCount = count;
}
}
}
return bestLine;
}

int countEquivalentLines(ArrayList<Line> lines, Line line) {
if (lines == null) return 0;
int count = 0;
for (Line parallelLine : lines) {
if (parallelLine.isEquivalent(line) count++;
}
return count;
}

int countEquivLines(
HashMap<Double, ArrayList<Line>> linesBySlope, Line line) {
double key = Line.floorToNearestEpsilon(line.slope);
double eps = Line.epsilon;
int count = countEquivalentLines(linesBySlope.get(key), line) +
countEquivalentLines(linesBySlope.get(key - eps), line) +
countEquivalentLines(linesBySlope.get(key + eps), line);
return count;
}

void insertLine(HashMap<Double, ArrayList<Line>> linesBySlope,
Line line) {
ArrayList<Line> lines = null;
double key = Line.floorToNearestEpsilon(line.slope);
if (!linesBySlope.containsKey(key)) {
lines = new ArrayList<Line>();
linesBySlope.put(key, lines);
} else {
lines = linesBySlope.get(key);
}
lines.add(line);
}

public class Line {
public static double epsilon = .0001;
public double slope, intercept;
private boolean infinite_slope = false;

public Line(GraphPoint p, GraphPoint q) {
if (Math.abs(p.x - q.x) > epsilon) { // if x's are different
slope = (p.y - q.y) / (p.x - q.x); // compute slope
intercept = p.y - slope * p.x; // y intercept from y=mx+b
} else {
```

Correction	8/9/2012	CS	7.6	Mathematics and Probability	271	} else {
						infinite_slope = true;
						intercept = p.x; // x-intercept, since slope is infinite
						}
						}
						
						public static double floorToNearestEpsilon(double d) {
						int r = (int) (d / epsilon);
						return ((double) r) * epsilon;
						}
						
						public boolean isEquivalent(double a, double b) {
						return (Math.abs(a - b) < epsilon);
						}
						
						public boolean isEquivalent(Object o) {
						Line l = (Line) o;
						if (isEquivalent(l.slope, slope) &&
						isEquivalent(l.intercept, intercept) &&
						(infinite_slope == l.infinite_slope)) {
						return true;
						}
						return false;
						}
						}
						We need to be careful about the calculation of the slope of a line. The line might be completely vertical, which means have a y-intercept and its slope is infinite. We can keep track of this in a separate flag (infinite_slope). We need to c condition in the equals method.

Minor	8/12/2012	OR	Introduction	Introduction	52	For consistency, line should say "By simple arithmetic, this reduces to (30h - 5.5m) % 360".
Spelling / Grammar / Typo	8/13/2012	PJ	Introduction	Introduction	56	correct spelling of "sacrafice" to "sacrifice"
Enhancement	8/13/2012	PJ	6.3	Brain Teasers	260	Separated step which says: "[2][0] Dumped 3 quart" into "[2][0] Dumped 3 quart" and "[0][2] Filled 3 quart with 5 quar Changed "Note that many brain teasers have a mathematical or computer science root " to "Many brain teasers have root "
Minor	8/30/2012	RS	4	Trees and Graphs	85	In the pseudocode for DFS, on line 6, an opening brace is missing (matching the closing brace on line 8).
Spelling / Grammar / Typo	9/5/2012	D	18.6	Hard	469	Changed "The basic algorithm operates works like this" to "The basic algorithm operates like this"
Spelling / Grammar / Typo	9/5/2012	D	18.5	Hard	167	Changed "searching operate" to "searching operate"
Enhancement	9/5/2012	D	18.5	Hard	469	Adding line at end of solution saying, "After the initial indexing of the file, this takes O(p + k) time, where p and k are occurrences of each word."
Enhancement	9/5/2012	D	18.5	Hard	167	Changed question to: "You have a large text file containing words. Given any two words, find the shortest distance (i number of words) between them in the file. If the operation will be repeated many times for the same file (but differen words), can you optimize your solution?"
Spelling / Grammar / Typo	9/6/2012	D	9	Recursion and Dynamic Programming	107	Changed "built off" to "built off of"
Spelling / Grammar / Typo	9/10/2012	D	11.5	Sorting and Searching	365	Last sentence of solution 11.5: "... that are you a careful coder." should be "that you are a careful coder"
Spelling / Grammar / Typo	9/11/2012	D	10.7	Scalability and Memory Limits	354	Should say "The result for a given query" instead of "The result for a given queries"
Spelling / Grammar / Typo	9/11/2012	D	13.8	C and C++	393	References to SmarterPointer should say SmartPointer
Spelling / Grammar / Typo	9/11/2012	D	16.6	Threads and Locks	160	Problem statement should reference Method B instead of Method C

Minor	9/13/2012	UN	9.7	Recursion and Dynamic Programming	328	Fixed bug where code hits infinite loop if old color == new color. Changed line in initial function to:
						boolean paintFill(Color[][] screen, int x, int y, Color ncolor) {
						if (screen[y][x] == ncolor) return false;
						return paintFill(screen, x, y, screen[y][x], ncolor);
						}

						Bug where code didn't compute start / end points of line correctly when one square was inside the other. Rewrote sol and rewrote comments for extend(...):
						/* Return the point where the line segment connecting mid1 and
						* mid2 intercepts the edge of square 1. That is, draw a line
						* from mid2 to mid1, and continue it out until the edge of
						* the square. */
						public Point extend(Point mid1, Point mid2, double size) {
						/* Find what direction the line mid2 -> mid1 goes */
						double xdir = mid1.x < mid2.x ? -1 : 1;
						double ydir = mid1.y < mid2.y ? -1 : 1;

Minor	9/15/2012	Bostonian	7.5	Mathematics and Probability	270	/* If mid1 and mid2 have the same x value, then the slope
						* calculation will throw a divide by 0 exception. So, we
						* compute this specially. */
						if (mid1.x == mid2.x) {
						return new Point(mid1.x, mid1.y + ydir * size / 2.0);
						}
						double slope = (mid1.y - mid2.y) / (mid1.x - mid2.x);
						double x1 = 0;
						double y1 = 0;
						
						/* Calculate slope using the equation (y1 - y2) / (x1 - x2).
						* Note: if the slope is “steep” (>1) then the end of the
						* line segment will hit size / 2 units away from the middle
						* on the y axis. If the slope is “shallow” (<1) the end of
						* the line segment will hit size / 2 units away from the
						* middle on the x axis. */
						if (Math.abs(slope) == 1) {
						x1 = mid1.x + xdir * size / 2.0;
						y1 = mid1.y + ydir * size / 2.0;
						} else if (Math.abs(slope) < 1) {
						x1 = mid1.x + xdir * size / 2.0;
						y1 = slope * (x1 - mid1.x) + mid1.y;
						} else {
						y1 = mid1.y + ydir * size / 2.0;
						x1 = (y1 - mid1.y) / slope + mid1.x;
						}
						return new Point(x1, y1);
						}
						
						public Line cut(Square other) {
						/* Calculate where a line between each middle would collide with the edges of the squares */
						Point point_1 = extend(this.middle(), other.middle(), this.size);
						Point point_2 = extend(this.middle(), other.middle(), -1 * this.size);
						Point point_3 = extend(other.middle(), this.middle(), other.size);
						Point point_4 = extend(other.middle(), this.middle(), -1 * other.size);
						
						/* Of above points, find start and end of lines. Start is farthest left (with top most as a tie breaker)
						* and end is farthest right (with bottom most as a tie breaker */
						Point start = point_1;
						Point end = point_1;
						Point[] points = {point_2, point_3, point_4};
						for (int i = 0; i < points.length; i++) {
						if (points[i].x < start.x || (points[i].x == start.x && points[i].y < start.y)) {
						start = points[i];
						} else if (points[i].x > end.x || (points[i].x == end.x && points[i].y > end.y)) {
						end = points[i];
						}
						}
						
						return new Line(start, end);
						}

Enhancement	9/15/2012	Bostonian	9.2	Recursion and Dynamic Programming	318	Rather than calling path.add in the beginning of the function (and then removing the point if it fails), call path.add onc
						successful. This is for both the recursive and DP problems. Recursive code looks as follows: public static boolean g
						y, ArrayList<Point> path) {
						Point p = new Point(x, y);
						if (x == 0 && y == 0) {
						return true; // found a path
						}
						boolean success = false;
						if (x >= 1 && isFree(x - 1, y)) { // Try right
						success = getPath(x - 1, y, path); // Free! Go right
						}
						if (!success && y >= 1 && isFree(x, y - 1)) { // Try down
						success = getPath(x, y - 1, path); // Free! Go down
						}
						if (success) {
						path.add(p); // Wrong way! Better stop going this way
						}
						return success;
						}

						Fixed pseudocode to use consistent variable names. moveDisks(int n, Tower origin, Tower destination, Tower buffer /* Base case */ if (n <= 0) return;

Spelling / Grammar / Typo	9/18/2012	Matt	3.4	Stacks and Queues	212	if (n <= 0) return;
						
						/* move top n - 1 disks from origin to buffer, using destination
						* as a buffer. */
						moveDisks(n - 1, origin, buffer, destination);
						
						/* move top from origin to destination
						moveTop(origin, destination);
						
						/* move top n - 1 disks from buffer to destination, using
						* origin as a buffer. */
						moveDisks(n - 1, buffer, destination, origin);
						}

Minor	9/19/2012	D	10.6	Scalability and Memory Limits	353	400 should say 4000 (multiple places in solution)
Spelling / Grammar / Typo	9/19/2012	Gayle	10.3	Scalability and Memory Limits	347	reversed order of words. changed "and non-negative 2^31 integers" to "and 2^31 non-negative integers"
Enhancement	9/19/2012	Gayle	10.3	Scalability and Memory Limits	348	Changed "If we count only 998 values in a particular range" to "If we count only 999 values in a particular range"
Minor	9/19/2012	BW	10.3	Scalability and Memory Limits	349	Fixed wording issue with follow-up question, as it's not actually possible for there to be 4 billion distinct non-negative (1) Changed question wording of second part to specify: "Assume that all the values are distinct and we now have no billion non-negative integers. (2) Changed "Let arraySize represent the number of blocks in the first pass. Note that arraySize = 2^32 / rangeSize, 2^32 integers." to "Let arraySize represent the number of blocks in the first pass. Note that arraySize = 2^31 / range there are 2^31 non-negative integers." (3) Changed equations on pg 349 to read 2^31 instead of 2^32 and 2^10 instead of 2^11 (4) Changed equation on pg 349 to read 2^10 <= rangeSize <= 2^26
Enhancement	9/20/2012	Gayle	10.3	Scalability and Memory Limits	348	Changed findOpenNumber2 to findOpenNumber and bitfield2 to bitfield
Spelling / Grammar / Typo	9/24/2012	SF	2	Linked Lists	77	top line on the page should say "linked list problem" instead of "linked problem"
Enhancement	9/24/2012	SF	11	Sorting and Searching	120	Added description of binary search. Updated description to below: When we think of searching algorithms, we genera binary search. Indeed, this is a very useful algorithm to study. In binary search, we look for an element x in a sorted array by first comparing x to the midpoint of the array. If x is le midpoint, then we search the left half of the array. If x is greater than the midpoint, then we search the right half of th then repeat this process, treating the left and right halves as subarrays. Again, we compare x to the midpoint of this then search either its left or right side. We repeat this process until we either find x or the subarray has size 0. Note that although the concept is fairly simple, getting all the details right is far more difficult than you might think. A code below, pay attention to the plus ones and minus ones.
Spelling / Grammar / Typo	9/24/2012	SF	12	Testing	128	"Many candidates balk at a question like this, given unrealistic answers" should be: "Many candidates balk at a question like this, giving unrealistic answers"
Spelling / Grammar / Typo	10/3/2012	FY	1.4	Arrays and Strings	175	In solution code to Question 1.4, Line 2: variable "i" doesn't need to be initialized to 0 sine it would be initialized in the following for loop.
Minor	10/9/2012	Itai	3	Stacks and Queues	80	Line 14: should return Object not Node
Minor	10/15/2012	CR	4.3	Trees and Graphs	86	Clarified that the array has unique integer elements. "Given a sorted (increasing order) array with unique integer elem algorithm to create a binary search tree with minimal height."
Spelling / Grammar / Typo	10/25/2012	FY	9.3	Recursion and Dynamic Programming	109	Adding clarification in question to specify that the array has unique integers (solution question was specified correctl was not). Question should read: A magic index in an array A[0...n-1] is defined to be an index such that A[i] = i. Give of distinct integers, write a method to find a magic index, if one exists, in array A.

Enhancement	10/25/2012	RB	11	Sorting and Searching	118	Modified mergesort implementation to use one helper array the entire time, without needing to allocate / deallocate a helper[] is passed into the mergesort function and merge function, with a new mergesort wrapper function being adde
						void mergesort(int[] array) {
						int[] helper = new int[array.length];
						mergesort(array, helper, 0, array.length - 1);
						}

Spelling / Grammar / Typo	11/1/2012	EB	12.3	Testing	379	"Test with y larger than the width" should be "Test with y larger than the height".
Enhancement	11/14/2012	JV	4.6	Trees and Graphs	229	Removed unnecessary check for n.parent == null in "if (n.parent == null || n.right != null)" on line 6
Minor	11/14/2012	EB	16.6	Threads and Locks	425	Added declaration of lock 3 and merged declarations to same line. Declarations now look like "public ReentrantLock lock3;". Added declaration of sem3 and merged declarations to same line. Declarations now look like "public Semapl sem2, sem3;"
Enhancement	11/14/2012	HH	5.1	Bit Manipulation	254	Changed implementation of BitInteger to reverse bit order. Least significant bit is now bit 0. There were minor change to support this: (1) findMissing now passes in 0 initially (2) the comparison around line 7 is now "if (column >= BitInteger.INTEGER_SIZE)" (3) Recursive call does column + 1 instead of column - 1
Spelling / Grammar / Typo	11/26/2012	CS	Introduction	Introduction	35	In the "Broad" bullet point under the description of what makes a good network, it says "...Some of those friends--wh friends or friends--will probably be looking...". It should be "friends of friends", not "friends or friends".
Spelling / Grammar / Typo	11/26/2012	WG	Introduction	Introduction	65	In item #3, "It's much for effective..." should be "It's much more effective..."
Spelling / Grammar / Typo	1/8/2013	EG	9.4	Recursion and Dynamic Programming	321	"doing {2 * 2 * ... } 2n times gives us 2^n" should be "doing {2 * 2 * ... } n times gives us 2^n"

Spelling / Grammar / Typo	1/8/2013	EG	9.4	Recursion and Dynamic Programming	321	"doing {2 * 2 * ...} 2n times gives us 2^n" should be "doing {2 * 2 * ...} n times gives us 2^n".
Minor	1/8/2013	ZS	4.1	Trees and Graphs	221	Changed last line of problem from "This code runs in O(N) time and O(log(N)) space" to "This code runs in O(N) time space, where H is the height of the tree."
Enhancement	1/8/2013	EX	1.1	Arrays and Strings	172	If-statement in second solution should say "if (str.length() > 26) return false;" instead of "if (str.length() > 256) return code only handles lower case characters
Clarification	1/8/2013	SX	4.9	Trees and Graphs	238	Clarified problem to say "You are given a binary tree in which each node contains a value. Design an algorithm to pri which sum to a given value. The path does not need to start or end at the root or a leaf."
Enhancement	1/23/2013	EX	2.5	Linked Lists	191	Removed unnecessary comparison in line 22 -- "if (l1 != null || l2 != null || value >= 10) {"
Enhancement	1/24/2013	SF	7	Mathematics and Probability	99	Changed line 8 from "while (prime <= Math.sqrt(max)) {" to "while (prime <= max) {"

Enhancement	1/28/2013	Gayle	18.2	Hard	463	Rewrote solution for additional clarity. New solution reads as follows: This is a very well-known interview question, ar algorithm. If you aren't one of the lucky few to already know this algorithm, read on.
						Let's imagine our n-element array. Suppose it looks like this:
						[1] [2] [3] [4] [5]
						Using our Base Case and Build approach, we can ask this question: suppose we had a method shuffle(...) that worke elements. Could we use this to shuffle n elements?
						Sure. In fact, that's quite easy. We would first shuffle the first n - 1 elements. Then, we would take the nth element a swap it with an element in the arary. That's it!
						Recursively, that algorithm looks like this:
						/* Random number between lower and higher, inclusive */
						int rand(int lower, int higher) {
						return lower + (int)(Math.random() * (higher - lower + 1));
						}
						
						int[] shuffleArrayRecursively(int[] cards, int i) {
						if (i == 0) return cards;
						
						shuffleArrayRecursively(cards, i - 1); // Shuffle earlier part
						int k = rand(0, i); // Pick random index to swap with
						
						/* Swap element k and i */
						int temp = cards[k];
						cards[k] = cards[i];
						cards[i] = temp;
						
						/* Return shuffled array */
						return cards;
						}
						What would this algorithm look like iteratively? Let's think about it. All it does is moving through the array and, for ea swapping array[i] with a random element between 0 and i, inclusive.
						This is actually a very clean algorithm to implement iteratively:
						void shuffleArrayInteratively(int[] cards) {
						for (int i = 0; i < cards.length; i++) {
						int k = rand(0, i);
						int temp = cards[k];
						cards[k] = cards[i];
						cards[i] = temp;
						}
						}
						The iterative approach is usually how we see this algorithm written.

						Rewrote solution for additional clarity: Like the prior problem which was similar, (problem 18.2 on page 463), we can l problem recursively using the Base Case and Build approach.
						Suppose we have an algorithm which can pull a random set of m elements from an array of size n - 1. How can we u algorithm to pull a random set of m elements from an array of size n?
						We can first pull a random set of size m from the first n - 1 elements. Then, we just need to decide if array[n] should our subset (which would require pulling out a random element from it). An easy way to do this is to pick a random nu through n. If k < m, then insert array[n] into subset[k]. This will both "fairly" (i.e., with proportional probability) insert a subset and "fairly" remove a random element from the subset.
						The pseudocode for this recursive algorithm would look like this:
						int[] pickMRecursively(int[] original, int m, int i) {
						if (i + 1 == m) { // Base case
						/* return first m elements of original */
						
						} else if (i + m > m) {
						int[] subset = pickMRecursively(original, m, i - 1);
						int k = random value between 0 and i, inclusive
						if (k < m) {
						subset[k] = original[i];
						}
						return subset;
						}
						return null;
						}

Enhancement	1/28/2013	Gayle	18.3	Hard	464	This is even cleaner to write iteratively. In this approach, we initialize an array subset to be the first m elements in or
						iterate through the array, starting at element m, inserting array[i] into the subset at (random) position k whenever k <
						int[] pickMIteratively(int[] original, int m) {
						int[] subset = new int[m];
						
						/* Fill in subset array with first part of original array */
						for (int i = 0; i < m ; i++) {
						subset[i] = original[i];
						}
						
						/* Go through rest of original array. */
						for (int i = m; i < original.length; i++) {
						int k = rand(0, i); // Random # between 0 and i, inclusive
						if (k < m) {
						subset[k] = original[i];
						}
						}
						
						return subset;
						}
						Both solutions are, not surprisingly, very similar to the algorithm to shuffle an array.

Correction	1/28/2013	MK	16.6	Threads and Locks	427	Corrected incorrect explanation. New wording is: By applying the word synchronized to a method, we ensure that two execute synchronized methods on the same object instance at the same time. So, the answer to the first part really depends. If the two threads have the same instance of the object, then no, they simultaneously execute method A. However, if they have different instances of the object, then they can. Conceptually, you can see this by considering locks. A synchronized method applies a “lock” on all synchronized me instance of the object. This blocks other threads from executing synchronized methods within that instance. In the second part, we’re asked if thread1 can execute synchronized method A while thread2 is executing non-synch B. Since B is not synchronized, there is nothing to block thread1 from executing A while thread2 is executing B. This regardless of whether thread1 and thread2 have the same instance of the object or not. Ultimately, the key concept to remember here is that only one synchronized method can be in execution per instance Other threads can execute non-synchronized methods on that instance, or they can execute any method on a differe the object.
Minor	1/29/2013	AL	1.5	Arrays and Strings	176	countCompression should check for empty string. Added line at line 33: "if (str == null || str.isEmpty()) return 0;"
Minor	1/31/2013	EX	4.5	Trees and Graphs	226	Changed solution to reject trees with duplicate values, since solution couldn't properly validate trees with duplicate va (1) Our first thought might be to do an in-order traversal, copy the elements to an array, and then check to see if the arra This solution takes up a bit of extra memory, but it works -- mostly. The only problem is that it can’t handle duplicate values in the tree properly. For example, the algorithm cannot distin the two trees below (one of which is invalid) since they have the same in-order traversal. Valid BST [20.left = 20] Invalid BST [20.right = 20] However, if we assume that the tree cannot have duplicate values, then this approach works. The pseudocode for th something like: (2) Changed line 14 of pseudocode from "if (array[i] < array[i - 1]) return false;" to "if (array[i] <= array[i - 1]) return fal (3) Changed line 9 of real code from "if (n.data < last_printed) return false;" to "if (n.data <= last_printed) return false;
Clarification	1/31/2013	DS	3.6	Stacks and Queues	215	Clarified problem wording to say: "Write a program to sort a stack in ascending order (with biggest items on top). You most one additional stack to hold items, but you may not copy the elements into any other data structure (such as a stack supports the following operations: push, pop, peek, and isEmpty. " Changed second paragraph to: "Unfortunately, we’re only allowed one additional stack. Can we do better? Yes." Added final notes at bottom of solution: "If we were allowed to use unlimited stacks, we could implement a modified mergesort. With the mergesort solution, we would create two extra stacks and divide the stack into two parts. We would recursiv stack, and then merge them back together in sorted order into the original stack. Note that this would require the crea additional stacks per level of recursion. With the quicksort solution, we would create two additional stacks and divide the stack into the two stacks based on element. The two stacks would be recursively sorted, and then merged back together into the original stack. Like the this one involves creating two additional stacks per level of recursion."
Spelling / Grammar / Typo	2/1/2013	Gayle	8.2	Object-Oriented Design	284	Fixed line wrap issue on line 30 of code; removed excess brace on line 16
Spelling / Grammar / Typo	2/4/2013	Gayle	17.4	Moderate	437	Comment on line 5 should read "int sb = sign(b); // if b >= 0, then 1 else 0" instead of "int sb = sign(b); // if b >= 1, tl
Enhancement	3/11/2013	QL	8	Object-Oriented Design	105	Added line at line 3: "protected Restaurant() { ... }"
Minor	3/22/2013	Siege	4.5	Trees and Graphs	228	On the very top of p228, the range for the right hand branch should be (min = 20, max = INT_MAX) instead of (min = INT_MAX).

Spelling / Grammar / Typo	3/22/2013	BW	18.6	Hard	469	Should say "max heap" instead of "min heap"
Enhancement	3/22/2013	Gayle	18.6	Hard	469	Changed second sentence of second paragraph under "max heap" to "On each element, if it's smaller than the root, the list and delete the largest element (which will be the root)."
Spelling / Grammar / Typo	3/22/2013	BW	18.9	Hard	474	At the bottom of the page, "heap1.top()" should read "maxHeap.top()".
Clarification	3/22/2013	AS	1.4	Arrays and Strings	73	Added "true" string length to example. Example now reads: "Input: “Mr John Smith ”, 13 Output: “Mr%20John%20Smith”
Spelling / Grammar / Typo	4/1/2013	BW	9.2	Recursion and Dynamic Programming	318	"Try right" should be "try left" and "try down" should be "try up" in second solution
Spelling / Grammar / Typo	4/2/2013	W	7.6	Mathematics and Probability	272	On line 30, countEquivLines should be countEquivalentLines
Clarification	4/4/2013	SC	5.5	Bit Manipulation	92	Changed question to clarify problem: "Write a function to determine the number of bits you would need to flip to conv integer B."
Enhancement	4/4/2013	Gayle	5.5	Bit Manipulation	92	Added better example: Input: 29 (or: 11101), 15 (or: 01111) Output: 2
Spelling / Grammar / Typo	5/3/2013	BW	2.7	Linked Lists	200	"nobe" in the line 10 on the top should read "node".
Correction	5/9/2013	LN	4.1	Trees and Graphs	220	The runtime of the first solution is O(N log N), not O(N^2). The last sentence of the second paragraph should read: " O(N log N) since each node is “touched” once per node above it. "
Enhancement	5/9/2013	SJ	9.6	Recursion and Dynamic Programming	326	Removed if statement on line 17 ("if (!set.contains(“()” + str)) {") since set already checks for duplicates
Spelling / Grammar / Typo	5/9/2013	Gayle	1.3	Arrays and Strings	174	Changed references to "anagram" to "permutation"
Spelling / Grammar / Typo	6/1/2013	SV	Introduction	Introduction	18	"For this book, we sought out interviewing experts from five top companies" should be "For this book, we sought out experts from six top companies"
Spelling / Grammar / Typo	6/1/2013	SV	Introduction	Introduction	24	"a product demos" should be "a product demo"
Spelling / Grammar / Typo	6/1/2013	SV	Introduction	Introduction	43	"These techniques can be used separately or *in together*." should be "These techniques can be used separately or
Spelling / Grammar / Typo	6/1/2013	SV	Introduction	Introduction	43	Removed double period after "S.A.R.."
Spelling / Grammar / Typo	6/1/2013	SV	Introduction	Introduction	44	Removed double period after "and, in fact, may be confused by them."
Spelling / Grammar / Typo	6/1/2013	SV	Introduction	Introduction	53	Changed comma to period at the end of the sentence: "Next, we try to solve it for n = 3, assuming that you have the 1 and n = 2"
Spelling / Grammar / Typo	6/1/2013	SV	Introduction	Introduction	60	Changed "Writing flexible, general-purpose code may also mean using constants instead of variables" to "Writing flex purpose code may also mean using variables instead of hard-coded values"
Spelling / Grammar / Typo	6/1/2013	BW	9.9	Recursion and Dynamic Programming	333	On the second line, "where column[row] = c indicates that row r has a queen at column c" should read "where colum indicates that row r has a queen at column c".
Spelling / Grammar / Typo	6/7/2013	MD	7	Mathematics and Probability	100	Unknown character showing up instead of intersect symbol in Venn diagrams
Minor	6/7/2013	MS	4.4	Trees and Graphs	225	Changed "The first solution uses O(log N) recursive calls" to "The first solution uses O(log N) recursive calls (in a ba