The first edition of this book was released on 9 August 2010.

The second edition of this book was released on 1 August 2011 (1 year gap).

The third edition of this book was released on 24 May 2013 (2 years gap).

The fourth edition of this book is NOT YET planned, but you can expect an exponential gap, i.e. 4 years after 2013.

As of Thursday, 13 Feb 2014, these are the known errata (yellow highlight => already fixed in Steven's soft copy):

a. Bugs fixes for the example codes

Page 127: memset(dfs_num, UNVISITED, sizeof dfs_num); was a legacy code. We have changed dfs_num from plain array to vi (vector of integers), therefore the initialization part should be written as dfs_num.assign(V, UNVISITED);

b. Correction to typos/grammatical/stylistic errors

Page 2, For very competitive programmer E: ... solve this 'well known' problem **in** ≤ 15 minutes...'.

Page 6. Footnote 5. "This number may vary", not "This number**s** may vary".

Page 9, Question 4 should not have the word "randomly". The points are only randomly scattered in Question 8, not in Question 4.

Page 10, Question 8, option (a). The complete search mentioned in question 4 (not question 3).

Page 27. The answer for exercise 1.2.2, question 3. should be " (f) Fenwick Tree ... then (e) Segment Tree" (the reference letters are wrong).

Page 28. The comment "Wed" written as the output for Exercise 1.2.3, task 3 is wrong. 9 August 2010 is "Mon".

Page 28. The typedef pair<int, ii> iii in Exercise 1.2.3. task 5 is slightly wrong, it should be typedef pairt<ii, int> iii, because I use pair first (for Month and Day), and then one more integer for Year

Page 29. The RegEx for Exercise 1.2.3 task 10 is not perfect as it may give extra spaces before and after the "***". The correct answer should be System.out.println(S.replaceAll("\\b[a-z] [0-9][0-9]\\b", "***"));

Page 141. Figure 4.13. The sentence "Edge 1-2 is fixed" is wrong. It should be "Edge 0-1" is fixed (off-by-one error).

Page 204. Binet's formula is NOT an approximation formula. It is a closed form formula to compute Fib(n). It is imprecise due to floating point computation. It is also not an O(1) method, but O(log n) method as raising a number to the power of n requires O(log n) steps.

Page 229. Exercise 5.4.4.2 answer. The "above" in "see above" refer to the fundamental counting principle mentioned in the answer of Exercise 5.4.4.1.

Page 231. There is an extra "and" in "Postfix Calculator and (Infix to Postfix) Conversion ~~and~~ (Section 9.27)".

Page 317. Section 9.2 about Bitonic TSP should be Section 9.3 (off-by-one error).