ERRATA 1.2
for
Introduction to Algorithms
by Cormen, Leiserson, and Rivest
July 28, 1994

This list describes the known bugs in the second and subsequent printings of the first edition of *Introduction to Algorithms*. An errata sheet for the first printing is available separately.

Typically, page and line numbers are given to localize the error. A negative line number indicates numbering from the bottom up. The finder of each bug is credited on the right margin. Actual text from the book is surrounded by ⟪ ⟫. Replacement text, where provided, is surrounded by ⟪ ⟫.

A PostScript version of this errata sheet is available via an Internet electronic mail server. To receive instructions on how to use this service, send electronic mail to **`algorithms@theory.lcs.mit.edu`** with "**`Subject: help`**" in the message header. The instructions also describe how to submit bug reports by email and how to obtain errata for the first printing. We regret that we cannot personally respond to all mail.

Page xvi, line 28 *Hershel Safer*

Change ⟪Herschel Safer⟫ *to* ⟪Hershel Safer⟫.

Page 11, Exercise 1.2-3 *Julie Sussman*

Exercise 1.2-3 depends on material introduced later in Chapter 1. Move it to Section 1.4. Bug fixed

---

Page 15, Exercise 1.3-3 *Stanley Selkow*

Change ⟪Use mathematical induction to show that the solution of the recurrence⟫ *to* ⟪Use mathematical induction to show that when n is an exact power of 2, the solution of the recurrence⟫. Bug fixed

---

Page 24, lines 22–24 *Charles E. Leiserson*

Replace the text ⟪every member of $\Theta(g(n))$⟫ *with* ⟪every member $f(n) \in \Theta(g(n))$⟫. *After the end of the sentence, insert the sentence* ⟪(An **asymptotically positive** function is one which is strictly positive for all sufficiently large n.)⟫. *As a result of this change, on page 33, line 9, the words* ⟪**asymptotically positive**⟫ *should no longer be boldfaced.*

Page 27, line −11 *Ron Rivest*

The text ⟪on or above $g(n)$.⟫ *should be changed to* ⟪on or above $cg(n)$.⟫.

Page 32, line −2 *George E. Collins*

The condition ⟪$b \neq 0$⟫ *should be changed to* ⟪$b > 0$⟫.

Page 33, line −3 *Michael Ernst*

The statement ⟪Thus, any positive exponential function grows faster than any polynomial.⟫ *should be restated* ⟪Thus, any exponential function with a base strictly greater than 1 grows faster than any polynomial function.⟫.

Page 35, line 13 *Hal Gabow*

The text ⟪$\lim_{n\to\infty} \frac{\lg^b n}{2^{a \lg n}}$⟫ *should be replaced by* ⟪$\lim_{n\to\infty} \frac{\lg^b n}{(2^a)^{\lg n}}$⟫.

Page 35, equation (2.12) *Greg Shannon*

The upper bound for $n!$ is incorrect for $n \leq 7$, but is correct, although loose, for $n \geq 8$. The upper bound should be changed to read as follows: ⟪

$$n! \leq \sqrt{2\pi n}(n/e)^n e^{1/12n} .$$

⟫

Page 55, line 10 *Julie Sussman*

The symbol ⟪$\leq$⟫ *should be replaced by* ⟪$>$⟫.

Page 55, line 18 *Julie Sussman*

The constraint ⟪$c \geq 2$⟫ *should be changed to* ⟪$c \geq 1$⟫.

Page 67, line 2 of figure caption *Julie Sussman*

The text ⟪height $\log_b a$⟫ *should be changed to* ⟪height $\log_b n$⟫.

Page 71, line −1 *Julie Sussman*

The symbol ⟪$\leq$⟫ *should be replaced by* ⟪$=$⟫.

Page 73, Problem 4-4, line 2 *Richard Anderson*

Change ⟪$n \leq 2$⟫ *to* ⟪$n \leq 8$⟫.

Page 82, line 11 *Lon Sunshine*

Change ⟪equivalent to A.⟫ *to* ⟪equivalent to a.⟫.

Page 87, line 14 *Charles Leiserson*

The term **isolated** *is not defined. Add the sentence* ⟪A vertex whose degree is 0, such as vertex 4 in Figure 5.2(b), is **isolated**.⟫ *after* ⟪has degree 2.⟫.

Page 88, lines 12–13 *Bruce Maggs*

The text ⟪In an undirected graph, a path $\langle v_0, v_1, \ldots, v_k \rangle$ forms a **cycle** if $v_0 = v_k$ and $v_1, v_2, \ldots, v_k$ are distinct.⟫ *should be replaced by* ⟪In an undirected graph, a path $\langle v_0, v_1, \ldots, v_k \rangle$ forms a **(simple) cycle** if $k \geq 3$, $v_0 = v_k$, and $v_1, v_2, \ldots, v_k$ are distinct.⟫.

Page 90, Exercise 5.4-2 *Bruce Maggs*

The exercise should be eliminated, because the bug fix to the definition of a cycle in an undirected graph on page 88, lines 12–13 obviates it.

Page 92, line 10 *Julie Sussman*

The text ⟪one path⟫ *should be changed to* ⟪one simple path⟫.

Page 115, Exercise 6.3-9 *Bobby Blumofe*

The equation to prove should read ⟪$\mathrm{Var}[aX] = a^2 \mathrm{Var}[X]$⟫, *not* ⟪$\mathrm{Var}[aX] = a^2 \mathrm{Var}[x]$⟫ Bug fixed

Page 121, line −3 *Julie Sussman*

The expression ⟪$\ldots(1-p)^{(n-k)-i)}$⟫ *has an extra right parenthesis. It should be replaced by* ⟪$\ldots(1-p)^{(n-k)-i}$⟫.

Page 124, line −1, and page 125, line 1 *Tom Cormen*

Replace the inequalities ⟪$q \leq 1$, $e^{\alpha q} \leq e^{\alpha}$, and $e^{-\alpha p} \leq 1$⟫ *by* ⟪$q_i \leq 1$, $e^{\alpha q_i} \leq e^{\alpha}$, and $e^{-\alpha p_i}$⟫.

Page 130, line 11 *David Wolfe*

The last inequality in bounding $\mathrm{E}[n]$ *should be an equality.* Bug fixed

Page 131, lines -2 and -1, and page 132, lines 1–4 *Steve Ponzio*

The "fact" given on page 132, line 2, is false (e.g., for $n = 2^{10}$). The text is modified to read: ⟪

$$\begin{aligned}(1-1/\sqrt{n})^{\lfloor 2n/\lfloor \lg n\rfloor\rfloor} &\le (1-1/\sqrt{n})^{2n/\lg n - 1} \\ &\le e^{-(2n/\lg n - 1)/\sqrt{n}} \\ &= O(e^{-\lg n}) \\ &= O(1/n)\ .\end{aligned}$$

For this argument, we used inequality (2.7), $1 + x \le e^x$.

Thus, the probability that the longest streak exceeds $\lfloor \lg n \rfloor /2$ is at least $1 - O(1/n)$. Since the longest streak has length at least 0, the expected length of the longest streak is at least

$$(\lfloor \lg n \rfloor /2)(1 - O(1/n)) + 0 \cdot (1/n) = \Omega(\lg n)\ .$$

Bug fixed ⟫

Page 133 (Problem 6-2), line -8 to -6 *Julie Sussman*

The problem statement now gives a simpler assumption about the input, which guarantees that the input numbers are distinct: ⟪Assume that the numbers in A are a random permutation of n distinct numbers.⟫ *In part (a), the element x is also now specified to be randomly chosen from a set of* ⟪i⟫ *distinct numbers, instead of a set of* ⟪n⟫ *distinct numbers.*

Bug fixed

Page 140, first sentence of Section 7.1 *Dick Johnsonbaugh*

Add the word ⟪nearly⟫ *before the phrase* ⟪complete binary tree⟫.

Page 142, Exercise 7.1-3 *Dick Johnsonbaugh*

Change the exercise to read ⟪Show that in any subtree of a heap, the root of the subtree contains the largest value occurring anywhere in that subtree.⟫.

Page 142, Exercise 7.1-4 *Dick Johnsonbaugh*

Change the exercise to read ⟪Where in a heap might the smallest element reside, assuming that all elements are distinct?⟫.

Page 145, lines 20–22 *Julie Sussman*

The text ⟪Our tighter analysis relies on the properties that in an n-element heap there are at most $\lceil n/2^{h+1} \rceil$ nodes of any height h (see Exercise 7.3-3).⟫ *is replaced to read* ⟪Our tighter analysis relies on the properties that an n-element heap has height $\lfloor \lg n \rfloor$ (see Exercise 7.1-2) and at most $\lceil n/2^{h+1} \rceil$ nodes of any height h (see Exercise 7.3-3).⟫

Bug fixed

Page 147, Exercise 7.3-3 *Julie Sussman*

This exercise should be starred.

Bug fixed

Page 150, Exercise 7.5-1 *Charles Leiserson*

Since inserting 3 *into the heap is trivial, the exercise is changed to insert the value* 10 *instead.*

Bug fixed

Page 159, line 22 *Tian Yuxing*

The text ⟪asks to you show⟫ *should be replaced by* ⟪asks you to show⟫.

Page 160, Exercise 8.2-2 *Julie Sussman and Charles Leiserson*

Change ⟪when the array A is sorted in nonincreasing order.⟫ *to* ⟪when the array A contains distinct elements and is sorted in decreasing order.⟫.

Page 170, part (c) of Problem 8-4 *Margrit Betke*

Append the sentence ⟪Maintain the $O(n \lg n)$ expected running time of the algorithm.⟫ *to the end of the problem part.*

Page 183, Problem 9-1 *Julie Sussman and Charles Leiserson*

There are several minor errors in parts (b)–(d). They should be replaced by ⟪

b. Let $D(T)$ denote the external path length of a tree T; that is, $D(T)$ is the sum of the depths of all the leaves of T. Let T be a tree with $k > 1$ leaves, and let LT and RT be the left and right subtrees of T. Show that $D(T) = D(LT) + D(RT) + k$.

c. Let $d(k)$ be the minimum value of $D(T)$ over all decision trees T with $k > 1$ leaves. Show that $d(k) = \min_{1 \le i \le k-1} \{d(i) + d(k-i) + k\}$. (*Hint:* Consider a decision tree T with k leaves that achieves the minimum. Let i be the number of leaves in LT and $k - i$ the number of leaves in RT.)

d. Prove that for a given value of $k > 1$ and i in the range $1 \le i \le k-1$, the function $i \lg i + (k-i)\lg(k-i)$ is minimized at $i = k/2$. Conclude that $d(k) = \Omega(k \lg k)$.

⟫

Pages 185, first paragraph *Ronald Greenberg*

To remove ambiguity and needless later verbiage, the definition of "median" is changed so that the phrase "the median" of a set is always well-defined. This change also causes some small changes to be made on pages 190 and 193. The revised introductory paragraph to the chapter now reads as follows. ⟪ The ith **order statistic** of a set of n elements is the ith smallest element.

For example, the **minimum** of a set of elements is the first order statistic ($i = 1$), and the **maximum** is the nth order statistic ($i = n$). A **median**, informally, is the "halfway point" of the set. When n is odd, the median is unique, occurring at $i = (n+1)/2$. When n is even, there are two medians, occurring at $i = n/2$ and $i = n/2+1$. Thus, regardless of the parity of n, medians occur at $i = \lfloor (n+1)/2 \rfloor$ (the **lower median**) and $i = \lceil (n+1)/2 \rceil$ (the **upper median**). For simplicity in this text, however, we consistently use the phrase "the median" to refer to the lower median. ⟫

Bug fixed

Page 186, last paragraph

Dick Johnsonbaugh

The bound claimed in this paragraph is tightened to $3\lceil n/2 \rceil - 2$, *and a sentence justifying this is added:* ⟪The first pair needs only one comparison to establish the initial values for the current minimum and current maximum, which accounts for the -2 term.⟫ *Also, the word* ⟪necessary⟫ *is changed to* ⟪sufficient⟫ *in the first sentence.*

Bug fixed

Page 189, line 3 (2nd line of equations)

Julie Sussman

Bug fixed

This should read ⟪$=$⟫ *rather than* ⟪$\leq$.⟫ .

Page 190

Ron Rivest

The description of SELECT *is improved to include a description of the base case* ($n = 1$) *of the recursion; the following sentence is added:* ⟪ (If $n = 1$ then SELECT merely returns its only input value as the ith smallest.) ⟫

Bug fixed

Page 190, step 2 of procedure SELECT

Ronald Greenberg

To maintain consistency with the revised definition of median (see bug report for page 185), and for simplicity in the algorithm, step 2 is revised to read as follows: ⟪ Find the median of each of the $\lceil n/5 \rceil$ groups by insertion sorting the elements of each group (of which there are 5 at most) and then picking the median from the sorted list of group elements. ⟫

Bug fixed

Page 191, Exercise 10.3-1

Ron Rivest

To make the problem a little easier, the last part of the exercise, ⟪How about groups of 3?⟫ *is replaced by the text* ⟪Argue that SELECT will not run in linear time if groups of 3 are used.⟫

Bug fixed

Page 193, Problems 10-1 (a) and (b)

Ronald Greenberg

Bug fixed

Commas should be inserted before the word "and" in both parts.

Page 193, Problem 10-2

Ronald Greenberg

To achieve consistency with the revised definition of median (see bug report for page 185), the introductory paragraph of this problem was revised as follows:

⟪ For n distinct elements $x_1, x_2, \ldots, x_n$ with positive weights $w_1, w_2, \ldots, w_n$ such that $\sum_{i=1}^{n} w_i = 1$, the **weighted median** is the element x_k satisfying

$$\sum_{x_i < x_k} w_i < \frac{1}{2}$$

and

$$\sum_{x_i > x_k} w_i \leq \frac{1}{2} .$$

(This is actually the **weighted lower median**; the **weighted upper median** would be defined similarly.) ⟫ Bug fixed

---

Page 194, line -5 *Julie Sussman*

The words, "linear-time" are added after the phrase "worst-case." Bug fixed

---

Page 208, Exercise 11.2-8, line 3 *Julie Sussman*

The word ⟪index⟫ *should be changed to* ⟪pointer⟫.

Page 212, 4 lines after the code for FREE-OBJECT *Dale Russell*

The word ⟪three⟫ *should be replaced by* ⟪two⟫.

Page 217, Problem 11-3, line 6 *Julie Sussman*

The word "in" has been left out of the sentence. The sentence can be further improved by replacing the phrase ⟪much faster than linear time.⟫ *with* ⟪in $o(n)$ time.⟫.

Page 217, Problem 11-3 *John Gateley*

The code for COMPACT-LIST-SEARCH *does not work properly when searching for the first element of the list. The test* ⟪$key[i] < k$⟫ *in line 3 of the code should be replaced by* ⟪$key[i] \leq k$⟫.

Page 228, line –16 *Margrit Betke*

The text ⟪$\alpha = 2000/3$⟫ *should be replaced with* ⟪$2000/3$⟫.

Page 234, lines 6–7 of first paragraph *Disk Johnsonbaugh*

The sentence, ⟪We would then modify the procedure HASH-SEARCH so that it keeps on looking when it sees the value DELETED, while HASH-INSERT would treat such a slot as if it were empty so that a new key can be inserted.⟫ *is replaced by the text,* ⟪We would then modify the procedure HASH-INSERT to treat such a slot as if it were empty so that a new key can be inserted. No modification of HASH-SEARCH is needed, since it will pass over DELETED values while searching. ⟫ Bug fixed

Page 234, line -11 *Ronald Greenberg*

Bug fixed

The text ⟪ $\langle h(k,1), h(k,2), \ldots, h(k,m)\rangle$ ⟫ *is replaced by* ⟪ $\langle h(k,0), h(k,1), \ldots, h(k,m-1)\rangle$ ⟫

Page 236, lines -9 to -6 *Ron Rivest*

Bug fixed

To clarify the example, the value of m' is given, so that the sentence now reads, ⟪ For example, if $k = 123456$, $m = 701$, and $m' = 700$, we have $h_1(k) = 80$ and $h_2(k) = 257$, so the first probe is to position 80, and then every 257th slot (modulo m) is examined until the key is found or every slot is examined. ⟫

Page 238, lines –7 and –6 *Dick Johnsonbaugh*

Add the words "at most" in two places, so these two lines now read as follows: ⟪number of probes in an unsuccessful search is at most $1/(1-.5)/2$. If it is 90 percent full, the average number of probes is at most $1/(1-.9) = 10$.⟫.

Page 239, line 4 *M. Veldhorst*

The text ⟪number of probes is $1/(1-\alpha)$⟫ *should be replaced with* ⟪number of probes is at most $1/(1-\alpha)$⟫.

Page 239, Theorem 12.7 *Dick Johnsonbaugh and Ron Rivest*

The bound in Theorem 12.7 is not very good for smallish α because of the unnecessary term $1/\alpha$. (Note that the bound in the theorem actually decreases as α increases from 0 to . 7!) Replace the text ⟪$\frac{1}{\alpha}\ln\frac{1}{1-\alpha} + \frac{1}{\alpha}$⟫ *in the statement of the theorem with the text* ⟪$\frac{1}{\alpha}\ln\frac{1}{1-\alpha}$⟫. *Replace lines –4 to –2 of the proof with* ⟪

$$\begin{aligned} \frac{1}{\alpha}(H_m - H_{m-n}) &= \frac{1}{\alpha}\sum_{k=m-n+1}^{m} 1/k \\ &\le \frac{1}{\alpha}\int_{m-n}^{m} (1/x)\,dx \\ &= \frac{1}{\alpha}\ln(m/(m-n)) \\ &= \frac{1}{\alpha}\ln(1/(1-\alpha)) \end{aligned}$$

⟫ *The constants on lines –2 and –1 of page 239 change from* ⟪3.387⟫ *to* ⟪1.387⟫ *and from* ⟪3.670⟫ *to* ⟪2.559⟫.

Page 239, line -3 *Dick Johnsonbaugh*

Add the words ⟪in a successful search⟫ *after the phrase* ⟪expected number of probes⟫.

Page 240, Exercise 12.4-4 *Dick Johnsonbaugh*

Replace the second sentence by ⟪Give upper bounds on the expected number of probes in an unsuccessful search and on the expected number of probes in a successful search.⟫

Page 240, Exercise 12.4-6 *Ron Rivest*

This exercise becomes irrelevant once the improvement to Theorem 12.7 noted in the erratum for page 239 has been made. Delete the exercise.

Page 242, part (d) of Problem 12-3 *Julie Sussman*

In order to be usable in part (3), the statement of part (d) is modified to read: ⟪Conclude that $P_k < 1/n^2$ for $k \geq k_0 = c \lg n / \lg\lg n$. ⟫ Bug fixed

---

Page 243, line 3 *Julie Sussman*

*The dot product notation in Exercise 12-5(c) is replaced by the equivalent summation notation, for clarity. Also, it now says "*mod$m$*" at the end of definition of h:* ⟪

$$h_{a,b}(x) = \sum_{i=0}^{r} a_i x_i + b \bmod m ,$$

⟫ Bug fixed

Pages 244 and 251–253 *Ronald Greenberg*

Referring to the "contents" of a node in a binary search tree is ambiguous, since it is unclear whether the contents includes the parent and child pointers. Use explicit reference to satellite data instead, entailing the following changes:

- *On page 244, line −5, change* ⟪In addition to a *key* field,⟫ *to* ⟪In addition to a *key* field and satellite data,⟫.
- *On page 251, lines −2 and −3, change* ⟪and replace the contents of z with the contents of y.⟫ *to* ⟪and replace z's key and satellite data with y's key and satellite data.⟫.
- *On page 252, Figure 13.4 caption, change* ⟪and then replace the contents of z with the contents of y.⟫ *to* ⟪and then replace z's key and satellite data with y's key and satellite data.⟫.
- *On page 253, line 16 of* TREE-DELETE*, change the line to read* ⟪▷ Copy y's satellite data, too.⟫.
- *On page 253, lines 8–9 of the paragraph following the* TREE-DELETE *pseudocode, change* ⟪the contents of z are moved from y to z, overwriting the previous contents⟫ *to* ⟪y's key and satellite data are moved to z, overwriting the previous key and satellite data.⟫.

Bug fixed

Page 250, Exercise 13.2-6 *Julie Sussman*

Replace Exercise 13.2-6 with ⟪Let T be a binary search tree whose keys are distinct, let x be a leaf node, and let y be its parent. Show that $key[y]$ is either the smallest key in T larger than $key[x]$ or the largest key in T smaller than $key[x]$.⟫.

Page 262, Problem 13-4 *Mark Kantrowitz*

Bug fixed

In the equation giving the Taylor expansion of $f(x)$, change ⟪$f^{(k)}(x-a)$⟫ *in the numerator to* ⟪$f^{(k)}(a)$⟫.

Page 263, line -2 *Ronald Greenberg*

Change ⟪to a leaf⟫ *to* ⟪down to a leaf⟫.

Page 266, caption for Figure 14.2, lines 2 and 5 *Julie Sussman*

In line 2, the text ⟪RIGHT-ROTATE(T, x)⟫ *should be changed to* ⟪RIGHT-ROTATE(T, y)⟫, *and in line 5, the text* ⟪LEFT-ROTATE(T, y)⟫ *should be changed to* ⟪LEFT-ROTATE(T, x)⟫.

Page 267, Exercise 14.2-3 *Julie Sussman*

The exercise is already pretty much answered in the text. It should be eliminated.

Page 267, Exercise 14.2-4, line 3 *Rosario Gennaro*

The text ⟪a left rotation is performed on node x⟫ *should be replaced by* ⟪a right rotation is performed on node y⟫.

Page 267, Exercise 14.2-5 *Rosario Gennaro*

The exercise should be amended to refer to binary search trees, instead of just trees. The new text is ⟪Show that any arbitrary n-node binary search tree can be transformed into any other arbitrary n-node binary search tree using $O(n)$ rotations. (*Hint:* First show that at most $n-1$ right rotations suffice to transform the tree into a right-going chain.)⟫.

Page 268, line 15 of RB-INSERT *Hubert Wagener*

A right bracket is missing: ⟪$color[p[p[x]]] \leftarrow$ RED⟫.

Page 273, line 15 of RB-DELETE *Hirendu Vaishnav*

This comment is ambiguous. Replace it with ⟪$\triangleright$ Copy y's satellite data, too. ⟫.

Page 277, Exercise 14.4-1 *Ronald Greenberg*

The claim is false unless the root was black before RB-DELETE *executes. The exercise should be rewritten* ⟪Argue that if a red-black tree has a black root before RB-DELETE executes, then it has a black root afterwards.⟫.

Page 278, Problem 14-1 *Julie Sussman*

That assumption that there is no parent field, which is explicit in parts (b) and (c) needs to be explicit in the problem text preceding part (a). The text ⟪Assume that each tree node has the fields *key*, *left*, and *right* but no parent field. (See also Exercise 14.3-6.)⟫ *should be moved from part (b) to just before part (a).*

Page 280, line 5 *Patricio Poblete*

Change ⟪an insertion⟫ *to* ⟪a deletion⟫.

Page 286, Exercises 15.1-1 and 15.1-2 *Julie Sussman*

Change references to ⟪Figure 15.2⟫ *to* ⟪Figure 15.1⟫.

Page 286, Exercise 15.1-7 *Peter Csaszar*

Chnage ⟪to to count⟫ *to* ⟪to count⟫.

Page 304, line −7 *Hoon Choi*

Change ⟪the the parenthesization⟫ *to* ⟪the parenthesization⟫.

---

page 325, Problem 16-2, lines 5–7 *George Madrid and Julie Sussman*

The number of extra space characters should be constrained to be nonegative, and it should be specifed that $i \leq j$. Change the sentence beginning ⟪If a given line⟫ *to read* ⟪If a given line contains words i through j, where $i \leq j$, and we leave exactly one space between words, the number of extra space characters at the end of the line is $M - j + i - \sum_{k=i}^{j} l_k$, which must be nonnegative so that the words fit on the line.⟫.

Page 326, Problem 16-3, first line of display *Hoon Choi*

Change ⟪$3 \cdot \text{cost(insert)}$⟫ *to* ⟪$4 \cdot \text{cost(insert)}$⟫.

Page 327, line −9 *James Park*

Change the reference ⟪[106]⟫ *for the Hu and Shing article to the two references listed in the erratum for page 992.* Bug fixed

---

Page 339, line 4 of figure 17.4 caption *Dale Russell*

Bug fixed

The equation ⟪ `f` = 100 ⟫ *is replaced by the correct text* ⟪ `f` = 101 ⟫ .

Page 346, line 20 *Dean Kelley*

Bug fixed

The text ⟪ the addition of x to A ⟫ *should read* ⟪ the addition of e to A ⟫

Pages 428–429, Figure 21.3 *Seongbin Park*

Bug not fixed

The auxiliary array A in Figure 21.3 should run from 0 to 3, not 0 to 4. The Fibonacci heap in part (a) has $n[H] = 15$ nodes. By Exercise 21.2-3, $D(n[H]) \le \lfloor \lg n[H] \rfloor$. Array A runs from 0 to $D(n[H])$ and $\lfloor \lg 15 \rfloor = 3$.

Page 439, line 5 of Chapter notes *Hal Gabow*

Bug fixed

Change ⟪ Driscoll, Sarnak, Sleator, and Tarjan ⟫ *to* ⟪ Driscoll, Gabow, Shrairman, and Tarjan ⟫

Page 440, line −2 *Ron Rivest*

Change ⟪ is pointed to by x ⟫ *to* ⟪ is x ⟫ .

Pages 443–446, Section 22.2 *Scot Drysdale*

Bug not fixed

The linked-list representation of disjoint sets requires that each list also include a pointer to its last element. Otherwise, the append operation does not take $O(1)$ time.

Page 443, lines −7 and −5 *Dick Johnsonbaugh*

There are $q-1$ UNION operations being executed, so replace ⟪ $q = m - n = \lfloor m/2 \rfloor - 1$ ⟫ *in line −7 with* ⟪ $q = m - n + 1 = \lfloor m/2 \rfloor$ ⟫ *and replace* ⟪ $m = n + q$ ⟫ *in line −5 with* ⟪ $m = n + q - 1$ ⟫ .

Page 444, Figure 22.3 caption *Dick Johnsonbaugh*

Change ⟪ $O(m^2)$ ⟫ *to* ⟪ $\Theta(m^2)$ ⟫ .

Page 446, Exercise 22.2-3 *Julie Sussman*

Bug fixed

Change ⟪ Argue on the basis of Theorem 22.1 that we can obtain ⟫ *to* ⟪ Adapt the proof of Theorem 22.1 to obtain ⟫ .

Page 450, Exercise 22.3-4 *Greg Shannon*

Because the pseudocode for the UNION *operation calls* FIND-SET, *a sequence of more than one* UNION *operation must contain some calls to* FIND-SET *before a call to* UNION. *Change the two appearances of* ⟪UNION⟫ *to* ⟪LINK ⟫ *in the exercise.*

Page 457, computation of $N(j)$ and $P(n)$ *Paul Beame*

Since there are at most n nodes, we have $N(0) \leq n$, which in turn implies that $N(j) \leq n/B(j)$ for all $j \geq 0$. The constant 3/2 in the computation of $P(n)$ can therefore be eliminated. Bug fixed

---

Page 467, line 19 *Julie Sussman*

The phrase ⟪the the transpose⟫ *should have the extra "the" removed.* Bug fixed

---

Page 474, line 8 *Dale Russell*

The sentence, ⟪Line 14...⟫, *which contains "only" twice, is modified to contain it only once:* ⟪Line 14 is therefore executed only for vertices with finite d values.⟫ Bug fixed

---

Page 475, line 3 *Jeff Shallit*

Replace ⟪lemma⟫ *with* ⟪theorem⟫. Bug fixed

Page 475, line -6 *Julie Sussman*

The phrase ⟪reachable from v⟫ *should read* ⟪reachable from s⟫. Bug fixed

---

Page 479, line 6 of text *Julie Sussman*

The phrase ⟪lines 1–2⟫ *should read* ⟪lines 1–3⟫. Bug fixed

Page 479, line -2 *Dale Russell*

The phrase ⟪lines 2–5⟫ *should read* ⟪lines 3–6⟫. Bug fixed

Page 480, line 11 *Haluk Konuk*

The text ⟪parenthesis "u)," then⟫ *should read* ⟪parenthesis "u)," then⟫ Bug fixed

---

Page 482, line -8 *Nils Thommesen*

To avoid possible confusion (as might occur in trying to solve Problem 23-1(a), the definition of a back edge is expanded to help remind the reader that self-loops are not of concern in undirected graphs. The new text reads: ⟪**Back edges** are those edges (u, v) connecting a vertex u to an ancestor v in a depth-first tree. Self-loops, which may occur in directed graphs, are considered to be back edges.⟫ Bug fixed

Page 487, lines -10 to -8, Exercise 23.4-1 — *Julie Sussman*

Bug fixed

This exercise should be modified to include the following phrase at the end: ⟪under the assumption of Exercise 23.3-2⟫.

Page 487, lines -3 to -1, Exercise 23.4-2 — *Dr. M. Veldhorst*

Bug fixed

This exercise should be deleted, as the first part is incorrect as stated.

Page 494, lines -4 to -3. Exercise 23.5-7 — *Eric Conrad*

Bug fixed

The given definition of semiconnected, ⟪A directed graph $G = (V, E)$ is said to be **semiconnected** if, for any two vertices $u, v \in V$, we have $u \leadsto v$ or $v \leadsto u$.⟫ *can be made clearer, as follows:* ⟪A directed graph $G = (V, E)$ is said to be **semiconnected** if, for all pairs of vertices $u, v \in V$, we have $u \leadsto v$ or $v \leadsto u$.⟫

Page 496, Problem 23-2 — *Jeff Shallit*

Part (b) of the problem is buggy as stated, and should be changed to read as follows: ⟪Let v be a nonroot vertex of G_π. Prove that v is an articulation point of G if and only if v has a child s such that there is no back edge from s or any descendant of s to a proper ancestor of v.⟫.

Page 505, last paragraph — *Julie Sussman*

The sentence ⟪At each step, a light edge connecting a vertex in A to a vertex in $V - A$ is added to the tree.⟫ *does not type check. It should be replaced by the sentence* ⟪At each step, a light edge is added to the tree A that connects A to an isolated vertex of $G_A = (V, A)$.⟫.

Page 513, last paragraph — *Hal Gabow*

Bug fixed

The reference to the best min spanning tree algorithm is time $O(E \lg \beta)$, not $O(E\beta)$. The reference is "Efficient algorithms for finding minimum spanning trees in undirected and directed graphs", H.N. Gabow, Z. Galil, T.H. Spencer and R.E. Tarjan, Combinatorica 6, 2, 1986, pp. 109-122.

Page 521, Figure 25.3 — *Julie Sussman*

Bug fixed

The procedure RELAX *takes three parameters. Add the parameter w to the calls to* RELAX *in the figure, and change the first sentence of the caption from* ⟪Relaxation of an edge (u, v).⟫ *to* ⟪Relaxation of an edge (u, v) with weight $w(u, v) = 2$.⟫.

Page 522, proof of Corollary 25.6 — *Julie Sussman*

Replace ⟪$\leq d[v]$; thus, so⟫ *to* ⟪$\leq d[v]$, and thus⟫.

Page 529, Figure 25.6 caption, line 3 *Julie Sussman*

Change ⟪$V - S$⟫ *to* ⟪S⟫.

Page 530, line –17 *Julie Sussman*

Change ⟪lines 4–8⟫ *to* ⟪lines 7–8⟫. Bug fixed

Page 534, Theorem 25.14 *Julie Sussman*

In the last line of the theorem statement, change ⟪reachable from S⟫ *to* ⟪reachable from s⟫. Bug fixed

Page 536, line –6 *Julie Sussman*

Change ⟪Unlike Dijkstra's algorithm, however, we use only $O(1)$ time per edge.⟫ *to* ⟪Unlike Dijkstra's algorithm, there is no priority queue, and so we use only $O(1)$ time per edge.⟫. Bug fixed

Page 537, line 6 of Figure 25.8 caption *Julie Sussman*

Change ⟪was used as v⟫ *to* ⟪was used as u⟫. Bug fixed

Page 547, Problem 25-4 *Julie Sussman*

Add the sentence ⟪We assume that all vertices are reachable from the source.⟫ *to the end of the second paragraph.* Bug fixed

Page 551, line 11 *Michael Ernst*

Change ⟪and otherwise π_{ij} is some predecessor of j on a shortest path from i⟫ *to* ⟪and otherwise π_{ij} is the predecessor of j on some shortest path from i⟫. Bug fixed

Page 555, Improving the running time *Julie Sussman*

An exercise shold be added at the end of the section to show that the multiplication performed by EXTEND-SHORTEST-PATHS *is associative and corresponds to extending shortest paths.* Bug not fixed

Page 560, line 6 of FLOYD-WARSHALL *Julie Sussman*

Add the keyword ⟪**do**⟫. Bug fixed

Page 560, first line after code for FLOYD-WARSHALL *Julie Sussman*

Change sentence to read ⟪Figure 26.4 shows the matrices $D^{(k)}$ computed by the Floyd-Warshall algorithm for the graph in Figure 26.1.⟫.

Page 562, line –10 *Julie Sussman*

Bug fixed

Change ⟪the logical operations ∨ and ∧⟫ *to* ⟪the logical operations ∨ (logical OR) and ∧ (logical AND)⟫.

Page 565, Exercise 26.2-8 *Ronald Greenberg*

Change the exercise to read as follows: ⟪Suppose that the transitive closure of a directed acyclic graph can be computed in $f(|V|, |E|)$ time, where f is a monotonically increasing function of $|V|$ and $|E|$. Show that the time to compute the transitive closure $G^* = (V, E^*)$ of a general directed graph $G = (V, E)$ is $f(|V|, |E|) + O(V + E^*)$.⟫.

Page 572, line –1 *Tom Cormen*

Replace the last line of the display with the following: ⟪

$$= \lambda(p_1) \odot (\bar{1} \oplus \lambda(c) \oplus (\lambda(c) \odot \lambda(c)) \oplus (\lambda(c) \odot \lambda(c) \odot \lambda(c)) \oplus \cdots) \odot \lambda(p_2) .$$

⟫

Page 585, line 1 *Ruben Glueck*

The two occurrences of the variable ⟪t_j⟫ *should be replaced by* ⟪t_i⟫.

Page 591, line –8 *Julie Sussman*

The term **minimum cut** *is never defined. Its definition should be added after the definition of* **capacity**: ⟪A **minimum cut** of a network is a cut whose capacity is minimum over all cuts of the network.⟫.

Page 593, line –12 *Rosario Gennaro*

The variable ⟪T⟫ *should be lower case:* ⟪no path from s to t in G_f⟫.

Page 598, line –5 *Julie Sussman*

Change ⟪at least 1⟫ *to* ⟪at least 0⟫.

Page 600, Exercise 27.2-10 *Julie Sussman*

Prepend the following sentence to the beginning of the exercise: ⟪Suppose that a flow network $G = (V, E)$ has symmetric edges, that is, $(u, v) \in E$ if and only if $(v, u) \in E$.⟫. *Also, in line 2 of the exercise, replace* ⟪$\delta(u, t)$⟫ *by* ⟪$\delta(v, t)$⟫.

Page 600, line –17 *Michael Ernst*

Change ⟪a maximum-flow problem⟫ *to* ⟪maximum-flow problems⟫.

Page 600, line -2 *Julie Sussman*

After ⟪between L and R.⟫, *the text* ⟪We further assume that no vertex in V is isolated.⟫ *should be inserted.*

Page 602, line −11 *Julie Sussman*

The variable ⟪m⟫ *should be upper case:* ⟪M⟫.

Page 603, line −5 *Nabil Kahale*

Replace ⟪so is f'.⟫ *with* ⟪we can assume that f' is integer-valued.⟫.

Page 621, line −4 *Thomas Lengauer*

The text ⟪in L must⟫ *should be changed to* ⟪in L (except possibly the first, which has no excess) must⟫.

Page 628, Problem 27-5, part (d) *Julie Sussman and Ron Rivest*

The problem part should be amended to read ⟪Show that the capacity of a minimum cut of the residual graph G_f is at most $2K\,|E|$ each time line 4 is executed.⟫.

Page 628, Problem 27-6 *Charles Leiserson*

The problem is confused between the notion of net flow and positive flow. The problem should be rewritten as follows: ⟪

27-6 Maximum flow with negative capacities

Suppose that we allow a flow network to have negative (as well as positive) edge capacities. In such a network, a feasible flow need not exist.

a. Consider an edge (u, v) in a flow network $G = (V, E)$ with $c(u, v) < 0$. Briefly explain what such a negative capacity means in terms of the positive flow between u and v.

Let $G = (V, E)$ be a flow network with negative edge capacities, and let s and t be the source and sink of G. Construct the ordinary flow network $G' = (V', E')$ with capacity function c', source s', and sink t', where

$$V' = V \cup \{s', t'\}$$

and

$$\begin{aligned} E' = E &\cup \{(u, v) : (v, u) \in E\} \\ &\cup \{(s', v) : v \in V\} \\ &\cup \{(u, t') : u \in V\} \\ &\cup \{(s, t), (t, s)\} \ . \end{aligned}$$

We assign capacities to edges as follows. For each edge $(u, v) \in E$, we set $c'(u, v) = c'(v, u) = (c(u, v) + c(v, u))/2$. For each vertex $u \in V$, we set $c'(s', u) = \max(0, (c(V, u) - c(u, V))/2)$ and $c'(u, t') = \max(0, (c(u, V) - c(V, u))/2)$. We also set $c'(s, t) = c'(t, s) = \infty$.

b. Prove that if a feasible flow exists in G, then all capacities in G' are nonnegative and a maximum flow exists in G' such that all edges into the sink t' are saturated.

c. Prove the converse of part (b). Your proof should be constructive, that is, given a flow in G' that saturates all the edges into t', your proof should show how to obtain a feasible flow in G.

d. Describe an algorithm that finds a maximum feasible flow in G. Let $MF(n, m)$ denote the worst-case running time of an ordinary maximum flow algorithm on a graph with n vertices and m edges. Analyze your algorithm for computing the maximum flow of a flow network with negative capacities in terms of MF.

⟫

Page 637, line −5 *Julie Sussman*

The **size** *of a comparison network is not defined. Replace* ⟪physical size⟫ *with* ⟪**size**, the number of comparators that it contains,⟫.

Page 642, first line of Section 28.3 *Hal Gabow*

The definition of bitonic sequence is wrong. The first two sentences of the section should be changed to read ⟪The first step in our construction of an efficient sorting network is to construct a comparison network that can sort any **bitonic sequence**: a sequence that monotonically increases and then monotonically decreases, or can be circularly shifted to become so. For example, the sequences $\langle 1, 4, 6, 8, 3, 2\rangle$, $\langle 6, 9, 4, 2, 3, 5\rangle$, and $\langle 9, 8, 3, 2, 4, 6\rangle$ are all bitonic.⟫.

Page 645, Exercise 28.3-1 *Julie Sussman*

The length of the sequences is not specified. The exercise should be rephrased ⟪How many zero-one bitonic sequences of length n are there?⟫.

Page 645, Exercise 28.3-4 *Hal Gabow*

The exercise is wrong for the definition of **bitonic sequence** *given on page 642. The exercise is correct after the definition is corrected as specified in the erratum for page 642.*

Page 646, line −14 *Bruce Maggs*

The text ⟪the the second⟫ *should be replaced by* ⟪the second⟫.

Page 648, Exercise 28.4-3 *Julie Sussman*

The exercise should state that the $n-1$ *items are already sorted. Replace the words* ⟪ $n-1$ items ⟫ *with the words* ⟪ $n-1$ sorted items ⟫.

Page 651, Problem 28-1, line –5 *Julie Sussman*

The range ⟪ $i = 2, 3, \ldots, n-1$ ⟫ *should be replaced by* ⟪ $i = 1, 2, \ldots, n$ ⟫.

Page 652, line –5 *Julie Sussman*

The notation for permutations used in this problem has not been sufficiently introduced. Replace ⟪ $\pi = \langle 4, 7, 3, 5, 1, 6, 8, 2 \rangle$ ⟫ *with* ⟪ $\pi = \langle \pi(1), \pi(2), \ldots, \pi(8) \rangle = \langle 4, 7, 3, 5, 1, 6, 8, 2 \rangle$ ⟫.

Page 663, line –6 *Selim Akl*

Change ⟪ We can think of y_i as a "prefix" of $x_0 \otimes x_1 \otimes \cdots \otimes x_n$; ⟫ *to* ⟪ We can think of y_i as a "prefix" of the "product" $x_0 \otimes x_1 \otimes \cdots \otimes x_n$; ⟫. Bug fixed

Page 666, line 1 *Julie Sussman*

Change ⟪ $[i, k]$ ⟫ *to* ⟪ $[i, j-1]$ ⟫. Bug fixed

Page 667, Figure 29.9 *Tom Cormen*

The label ⟪ y_0 ⟫ *at the top of each tree should be* ⟪ y_8 ⟫.

Page 672, line 3 of Figure 29.13 caption *Julie Sussman*

Change ⟪ Here, $n = 8$. ⟫ *to* ⟪ Here, $n = 4$. ⟫. Bug fixed

Page 677, line 5 *Selim Akl*

Change ⟪ $\lfloor 2n/3 \rfloor$ ⟫ *to* ⟪ $\lfloor n/3 \rfloor$ ⟫. Bug fixed

Page 696, line 16 *Tom Cormen*

Replace ⟪ for $0 \le i \le j < k \le n$. ⟫ *with* ⟪ for $1 \le i \le j < k \le n$. ⟫.

Page 700, line 15 *Julie Sussman*

Replace ⟪ the the ⟫ *with* ⟪ the ⟫.

Page 702, line 7 of FIND-ROOTS *Asterio Tanaka*

The root fields are assigned to improperly. Replace line 7 with the following two lines of code: ⟪

then if *parent*[*parent*[*i*]] = NIL
then *root*[*i*] ← *root*[*parent*[*i*]]

⟫ *The lines must be renumbered, which affects some of the references in the subsequent text.*

Page 718, line –4 *Michael Ernst*

Replace ⟪of processors⟫ *with* ⟪of the processors⟫.

Page 752, line –9 *Anthony Martin Hill*

Replace ⟪work backward to to⟫ *with* ⟪work backward to⟫.

Page 759, line 4 of LUP-DECOMPOSITION *Tom Cormen*

Replace ⟪$n-1$⟫ *with* ⟪n⟫.

Page 760, caption of Figure 31.2 *Tom Cormen*

In conjunction with the bug fix for line 4 of LUP-DECOMPOSITION *on page 759, replace* ⟪**(g)**–**(i)** The third step finishes the algorithm.⟫ *with* ⟪**(g)**–**(i)** The third step. No further changes occur on the fourth and final step. ⟫.

Page 764, Theorem 31.12 *Hal Gabow*

The conditions on $M(n)$ in the statement of the theorem are incorrect. Change the statement of the theorem to ⟪Suppose we can multiply two $n \times n$ real matrices in time $M(n)$, where $M(n) = \Omega(n^2)$ and $M(n)$ satisfies the two regularity conditions $M(n+k) = O(M(n))$ for any k in the range $0 \leq k \leq n$ and $M(n/2) \leq cM(n)$ for some constant $c < 1/2$. Then we can compute the inverse of any real nonsingular $n \times n$ matrix in time $O(M(n))$.⟫ *The proof should explicitly reference these conditions where needed.*

Page 773, line 13 *Jeff Shallit*

Replace ⟪$A_{ij} = 1$⟫ *with* ⟪$a_{ij} = 1$⟫.

Page 783, Exercise 32.1-4 *Rosario Gennaro*

The hint as given is incorrect and misleading. The problem is revised to read: ⟪Show how to use equation (32.5) to interpolate in time $\Theta(n^2)$. (*Hint:* First compute $\prod_j (x - x_j)$ and then divide by $(x - x_k)$ as necessary for the numerator of each term. See Exercise 32.1-1.) ⟫

Bug fixed

Page 791, line 2 of Exercise 32.2-8 *Ron Rivest*

In Exercise 32.2-8, page 791 (chirp transform), the expression ⟪

$$y_k = \sum_{j=0}^{n-1} z^j$$

⟫ *should read* ⟪

$$y_k = \sum_{j=0}^{n-1} z^{kj}$$

⟫ Bug fixed

Page 798, Problem 32-4, last line on page *Ron Rivest*

The remainder of $3x^3 + x^2 - 3x + 1$ *when divided by* $x^2 + x + 2$ *is not* $5x - 3$, *but* $-7x + 5$. Bug fixed

Page 799, line 4 *Dina Kravets*

The variable ⟪ p_{ij} ⟫ *should be* ⟪ P_{ij} ⟫. Bug fixed

Page 812, line 1 (not counting caption) *Ron Rivest*

The phrase ⟪an arbitrary pair of integers⟫ *is replaced by* ⟪an arbitrary pair of nonnegative integers⟫. Bug fixed

Page 816, line -8 *Rosario Gennaro*

The multiplicative group over Z_n *should be denoted as* ⟪ $(Z_n^*, \cdot_n)$ ⟫ *instead of* ⟪ $(Z_n, \cdot_n)$ ⟫. Bug fixed

Page 817, Theorem 33.14 *Nabil Kahale*

The statement of the theorem and the following comment are modified to include the condition that the set S' *must be nonempty, as follows:* ⟪

Theorem 0.1 (A nonempty closed subset of a finite group is a subgroup)
If $(S, \oplus)$ is a finite group and S' is any nonempty subset of S such that $a \oplus b \in S'$ for all $a, b \in S'$, then $(S', \oplus)$ is a subgroup of $(S, \oplus)$.

⟫ Bug fixed

Page 829, Proof of Corollary 33.35 *Sholom Rosen*

The corollary is stated correctly, but the proof is not, since the contrapositive to Theorem 33.34 is stronger than the corollary. (Indeed the contrapositive *is just the second sentence in the proof supplied.) The proof is modified to read:* ⟪ Theorem 33.34 implies that if there exists a nontrivial square root of 1, modulo n, then n can't be a prime or a power of a prime. Furthermore, we must have $n > 1$ for a nontrivial square root of 1 to exist. Therefore, n must be composite. ■

Bug fixed ⟫

Page 834, line -7 *Ron Rivest*

Bug fixed *The use of absolute value notation to denote the length of numbers (in binary) is changed to use the binary logarithm function instead.*

Page 837, line -9 *Victor S. Miller*

Bug fixed *The equation* ⟪ $\pi(n) = 50,847,478$ ⟫ *for* $n = 10^9$ *is incorrect. The correct value is* ⟪ $\pi(n) = 50,847,534$ ⟫. *(Reference: "Computing* $\pi(x)$*: The Meissel-Lehmer Method", by J. C. Lagarias, V. S. Miller, and A. M. Odlyzko, Mathematics of Computation 44, 170(April 1985), 537–560.)*

Page 838, lines 15–17 *Ron Rivest*

Bug fixed *This sentence is rewritten to read,* ⟪ When it works, trial division has the advantage that it not only determines whether n is prime or composite, but also determines one of n's prime factors if n is composite. ⟫

Page 840, line –5 *Len Adleman*

The expression ⟪ $a^{n-1} \pmod 1$ ⟫ *should read* ⟪ $a^{n-1} \pmod n$ ⟫.

Page 844, Exercise 33.8-1 *Boaz Patt*

Bug fixed *The phrase* ⟪ an integer n ⟫ *is replaced to read* ⟪ an odd integer n ⟫.

Page 850, Problem 33-3(c) *Stephen Guattery*

The matrix given in the hint should be: ⟪

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$$

Bug fixed ⟫

Page 853, line 9 *Ron Rivest*

Bug fixed *The text* ⟪ of length m ⟫ *is replaced by* ⟪ of length $m \leq n$ ⟫

Page 862, line 8 (Exercise 34.2-4) *Rosario Gennaro*

Bug fixed *The set* ⟪ $\{0, 1, \ldots, n-1\}$ ⟫ *should be* ⟪ $\{0, 1, \ldots, q-1\}$ ⟫.

Page 881, line 3 *Tian Yuxing*

The statement ⟪ $\lambda[a] = 0$ ⟫ *should be* ⟪ $\lambda[a] \leftarrow 0$ ⟫ Bug fixed

Pages 895–898 *Hal Gabow and Danny Sleator*

The ANY-SEGMENTS-INTERSECT *procedure can miss intersections in which the right endpoint of one segment coincides with the left endpoint of another. To correct the procedure, forget about y-coordinates when breaking ties in line 2. Instead, break ties by putting left endpoints before right endpoints. That way, the left endpoint is still in the sweep-line status when a coincident right endpoint becomes the event point.*

Also, the pseudocode, as is, works even when three or more segments intersect at a common point despite the assumption in the text that this case does not occur. In fact, the pseudocode can even handle vertical segments if we call the top endpoint the left endpoint and the bottom endpoint the right endpoint. Exercise 35.2-8 on page 898 is then obviated. Bug not fixed

Page 902, line 10 of GRAHAM-SCAN *Thomas Lengauer*

Change ⟪ PUSH(S, p_i) ⟫ *to* ⟪ PUSH(p_i, S) ⟫. Bug fixed

Page 908, line 4 of Section 35.4 *Dick Johnsonbaugh*

Change ⟪ $\sqrt{(x_1 - x_2)^2 - (y_1 - y_2)^2}$ ⟫ *to* ⟪ $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ ⟫. Bug fixed

Page 912, Exercise 35.4-2 *Dick Johnsonbaugh*

Delete the last sentence of the exercise. Bug fixed

Page 917, lines 2–3 of Section 36.1 *Tian Yuxing*

Change ⟪ These problems are generally regarded as tractable. The reason why is a philosophical, not a mathematical, issue. ⟫ *to* ⟪ These problems are generally regarded as tractable, but for philosophical, not mathematical, reasons. ⟫. Bug fixed

Page 919, line 6 *Hoon Choi*

Change ⟪ must must be represented ⟫ *to* ⟪ must be represented ⟫.

Page 919, second paragraph under subheading Encodings *Thomas Hofmeister*

Before the sentence beginning "A concrete problem ...", add the footnote ⟪ We assume that the algorithm's output is separate from its input. Because it takes at least one time step to produce each bit of the output and there are $O(T(n))$ time steps, the size of the output is $O(T(n))$. ⟫. Bug fixed

Page 920, line 8 *Thomas Hofmeister*

Bug fixed

Change ⟨⟨ $\lceil \lg k \rceil$ ⟩⟩ *to* ⟨⟨ $\lfloor \lg k \rfloor + 1$ ⟩⟩.

Page 920, line −5 *Dale Russell*

Bug fixed

Change ⟨⟨where $n = |e_1(i)|$.⟩⟩ *to* ⟨⟨where $n = |e_2(i)|$.⟩⟩.

Page 922, lines 23–33 *Dick Johnsonbaugh*

The definitions should be made more precise. Change the paragraph to read as follows:

⟨⟨ Even if language L is accepted by an algorithm A, the algorithm will not necessarily reject a string $x \notin L$ provided as input to it. For example, the algorithm may loop forever. A language L is **decided** by an algorithm A if every binary string in L is accepted by A and every binary string not in L is rejected by A. A language L is **accepted in polynomial time** by an algorithm A if there is a constant k such that for any length-n string x, the algorithm accepts x in time $O(n^k)$ if and only if x is in L. A language L is **decided in polynomial time** by an algorithm A if there is a constant k such that for any length-n string $x \in \{0,1\}^*$, the algorithm correctly decides whether $x \in L$ in time $O(n^k)$. Thus, to accept a language, an algorithm need only worry about strings in L, but to decide a language, it must correctly accept or reject every string in $\{0,1\}^*$. ⟩⟩

Bug fixed

Page 923, line 3 *Anand Kanagala*

Bug fixed

Change ⟨⟨on an algorithm that determines⟩⟩ *to* ⟨⟨of an algorithm that determines⟩⟩.

Page 923, line 7 *Dale Russell*

Change ⟨⟨wee can provide⟩⟩ *to* ⟨⟨we can provide⟩⟩.

Page 924, Exercise 36.1-5 *Ron Rivest*

Bug fixed

Change ⟨⟨Suppose that a language L can accept any string $x \in L$ in polynomial time, but that the algorithm that does this runs in superpolynomial time if $x \notin L$.⟩⟩ *to* ⟨⟨Suppose that there is a language L for which there is an algorithm that accepts any string $x \in L$ in polynomial time, but this algorithm runs in superpolynomial time if $x \notin L$.⟩⟩.

Page 924, Exercise 36.1-6 *David Mix Barrington and Andrei Toom*

Bug fixed

Change ⟨⟨Show that an algorithm⟩⟩ *to* ⟨⟨Show that an otherwise polynomial-time algorithm⟩⟩.

Page 930, line 6 of running text *Thomas Hofmeister*

Change ⟪In Section 36.5, shall use⟫ *to* ⟪In Section 36.5, we shall use⟫. Bug fixed

---

Page 932, Theorem 36.4 *Thomas Hofmeister*

Change the theorem statement and proof to ⟪
Theorem 36.4
If any NP-complete problem is polynomial-time solvable, then P = NP. Equivalently, if any problem in NP is not polynomial-time solvable, then no NP-complete problem is polynomial-time solvable.

Proof Suppose that $L \in$ P and also that $L \in$ NPC. For any $L' \in$ NP, we have $L' \leq_{\mathrm{P}} L$ by property 2 of the definition of NP-completeness. Thus, by Lemma 36.3, we also have that $L' \in$ P, which proves the first statement of the theorem.

To prove the second statement, note that it is the contrapositive of the first statement. ■

⟫ Bug fixed

Page 932, line −5 *Thomas Hofmeister*

Change ⟪NP-compete⟫ *to* ⟪NP-complete⟫. Bug fixed

Page 942, line 11 *Thomas Hofmeister*

Change ⟪Why is the circuit ϕ⟫ *to* ⟪Why is the circuit C⟫. Bug fixed

---

Page 945, line 10 *Thomas Hofmeister*

Change ⟪as clauses of $f(\phi)$.⟫ *to* ⟪as clauses of ϕ'''.⟫. Bug fixed

---

Page 949, line −1 *Thomas Hofmeister*

Change ⟪graph G has vertex cover of size k⟫ *to* ⟪graph G has a vertex cover of size k⟫. Bug fixed

---

Page 953, paragraph beginning "Now suppose ..." *Dale Russell*

Change ⟪Let $S = \{x_{i_1}, x_{i_2}, \ldots, x_{i_m}\} \cup \{y_{j_1}, y_{j_2}, \ldots, y_{j_p}\}$.⟫ *to* ⟪Let $S' = \{x_{i_1}, x_{i_2}, \ldots, x_{i_m}\} \cup \{y_{j_1}, y_{j_2}, \ldots, y_{j_p}\}$.⟫ *and change* ⟪there are three 1's in set S in the e_j position:⟫ *to* ⟪set S contains three integers with 1's in the e_j position:⟫. Bug fixed

Page 959, lines –7 and –8

Dale Russell

Bug fixed

Change ⟪The formal language for the traveling-salesman problem⟫ *to* ⟪The formal language for the corresponding decision problem⟫.

Page 960, Exercise 36.5-1

Dick Johnsonbaugh

Bug fixed

Change ⟪whether G_1 is a subgraph of G_2⟫ *to* ⟪whether G_1 is isomorphic to a subgraph of G_2⟫.

Page 967, caption to Figure 37.1

Dale Russell

Bug fixed

The caption is modified to remove all references to the set A, and replace these with references to the set C of vertices in the vertex cover being constructed.

Page 974, Exercise 37.2-3

Hal Gabow

Bug not fixed

The hint says to use a minimum spanning tree. It should be to use a bottleneck spanning tree, since the bottleneck property is what is used in the proof of correctness of the algorithm. The text implies that a minimum spanning tree is a bottleneck tree, but does not state it explicitly. Anyway, since a bottleneck tree can be found faster than a min sp tree, ($O(E)$ versus nonlinear) saying "bottleneck" is the best hint. (Cross-reference to Exercise 24.2-6 on page 510?)

Page 983, line -2

Dick Johnsonbaugh

Bug fixed

The text ⟪the the⟫ *should be* ⟪the⟫.

Page 984, Problem 37-2

Luisa Gargano

Bug fixed

The definition of $G^{(k)}$ was given incorrectly (in particular, the definition of the edge set was wrong). The corrected definition now reads: ⟪ For any $k \geq 1$, define $G^{(k)}$ to be the undirected graph $(V^{(k)}, E^{(k)})$, where $V^{(k)}$ is the set of all ordered k-tuples of vertices from V and $E^{(k)}$ is defined so that $(v_1, v_2, \ldots, v_k)$ is adjacent to $(w_1, w_2, \ldots, w_k)$ if and only if for each i, $1 \leq i \leq k$, either vertex v_i is adjacent to w_i in G, or else $v_i = w_i$. ⟫

Page 992, reference [106]

James Park

Bug fixed

Replace reference [106] with two references: ⟪
T. C. Hu and M. T. Shing, *Computation of Matrix Chain Products. Part I. SIAM Journal on Computing*, 11(2):362–373, 1982.
T. C. Hu and M. T. Shing, *Computation of Matrix Chain Products. Part II. SIAM Journal on Computing*, 13(2):228–251, 1984.
⟫

Page 995, reference [176] *Michael Formann*

Change ⟪16th⟫ *to* ⟪17th⟫ *and* ⟪1975⟫ *to* ⟪1976⟫. Bug fixed

Page 1006, column 2, line 11 *Ronald Greenberg*

Change ⟪562–563⟫ *to* ⟪562–565⟫. Bug fixed

Page 1026, index entry for transitive closure *Ronald Greenberg*

Change ⟪562–563⟫ *to* ⟪562–565⟫. Bug fixed