THE STATE OF THE ART OF COMPUTER PROGRAMMING

by

D. E. Knuth

STAN-CS-76-551
JUNE 1976

COMPUTER SCIENCE DEPARTMENT
School of Humanities and Sciences
STANFORD UNIVERSITY

The State of The Art of Computer Programming

Donald E. Knuth
Computer Science Department
Stanford University
Stanford, California 94305

This report lists all corrections and changes to volumes 1 and 3 of <u>The Art of Computer Programming</u>, as of May 14, 1976. The changes apply to the most recent printings of both volumes (February and March, 1975); if you have an earlier printing there have been many other changes not indicated here. Volume 2 has been completely rewritten and its second edition will be published early in 1977. For a summary of the changes made to volume 2, see <u>SIGSAM Bulletin</u> 9, 4 (November 1975), p. 10f -- the changes are too numerous to list except in the forthcoming book itself.

On any given day the author likes to feel that the last bug has finally disappeared, yet it appears likely that further amendments will be made as time goes by. Therefore a family of computer programs has been written to maintain a collection of errata, in the form printed here, but encoded as an ad-hoc sequence of ASCII characters. The author wishes to thank Juan Ludlow-Saldivar for the enormous amount of help he provided in order to get this system rolling. (Some readers who have access to the Stanford A.I.-Lab computer may wish to consult the change file before they report a "new" error; the file name is ACP.MAS [ART,DEK]. Entries for page nnn of volume k begin with βk01nnn (but change the 01 to 00 if nnn is the Arabic equivalent of a Roman numeral); since "β" is the control character "↑C", you may rather search for simply the string "k01nnn". The text of the correction usually includes special codes following the symbol "|", for things like font changes, etc.)

The author thanks all the bounty hunters who have reported difficulties they spotted. The reward to first finder of each error is still $1 for the first edition and $2 for the second, gratefully paid. Volume 4 remains rather far from completion, so there is plenty of time to work all the exercises in volumes 1 - 3 and to catch all the remaining errors therein.

This research was supported in part by National Science Foundation grant MCS 72-03752 A03 and by the Office of Naval Research contract N00014-76-C-0330.

The Art of Computer Programming

Errata et Addenda May 14 1976

1.xvii line 5 1

forcing himself $\leadsto$ being encouraged

1.xix line 10 2

answer $\leadsto$ answers

1.xix new quote for bottom of page 3

We can face our problem.
We can arrange such facts as we have
***with** order and method.*
--HERCULE POIROT, in *Murder on the Orient Express* (1934)

1.4 line 23 4

E0. $\leadsto$ **E0. (boldface)**

1.16 line 3 5

prove A6 $\leadsto$ prove that A6

1.19 line -1 6

$3n_0$ $\leadsto$ $3n$

1.19 lines -3 and -2 7

$T \leq 3n_0$, where n_0 is the original value of n, $\leadsto$ $T \leq n$,

1.26 **ex 25**

delete step L5 and move the l to the end of step L4

1.26 ex 25, change step L3 to: 9

L3. [Shift.] If $x-z < 1$, set $z \leftarrow z$ shifted right 1, $k \leftarrow k+1$, and repeat this step.

1.26 line 15, new sentence 10

hardware. $\rightsquigarrow$ hardware. Tho idea goes hack in essence to Henry Rriggs, who used it (in decimal rather than binary farm) to compute logarithm tables, published in 1624.

1.27 line 23 11

example, $\rightsquigarrow$ example --

1.36 exercise 40 12

. a period (.) should appear after the displayed equation

1.44 line 2 two changes 13

(i) the (q/p) and (p/q) don't match each other. (ii) the first two lines of p44 should h o moved hack to p43, otherwise the reader will think exercise 47 is complete without turning the page.

1.46 line 20 14

$1/12n \rightsquigarrow 1/(12n)$

1.50 ex 15 15

put spaces in the first matrix, i.e.

ah! $\rightsquigarrow$ *a b c*

def $\rightsquigarrow$ *d e f*

ghi $\rightsquigarrow$ *g h i*

1.52 line 7 after Table 1 *16*

Shih-chich ⇝ Shih-Chich

1.56 left side of **eq.** (17) *17*

move the k a little left, to center it

1.58 line 7 after (26) *18*

Shih-chich ⇝ Shih-Chich

1.58 line 8 after (26) *19*

the boldface *3* appears to be in wrong font (too small)

1.71 14 places *20*

change *B* to B (Roman type) in the notation for Beta function, namely in line 1, line 2, line 3 (twice), line 4 (thrice), line 5 (twice), line 7, line 10 (twice), line 12, line 15.

1.71 exercise 47 *21*

in displayed formula: change upper indices Cram n, $n+1/2$, $2n+1$, $2n+1-k$ to r, $r-1/2$, $2r$, $2r-k$ respectively
line 3: n . -1. ⇝ $r = -1/2$.

1.78 lines -3 and -2 *22*

before the Renaissance. ⇝ during the Middle Ages.

1.80 line 2 *23*

1963-) ⇝ 1963-),

1.90 between (23) and (24) *24*

***series* ⇝ *series* (cf. (17))**

1.90 insert new sentence just after (26): 25

See D. A. Zave, *Inf. Proc. Letters* 6 (1976), to appear, for a further generalization.

1.90 replace (25) by new equation (25): 26

$$\frac{1}{(1-z)^{m+1}} \ln\left(\frac{1}{1-z}\right) = \sum_{k\ge 0}(H_{m+k}-H_m)\binom{m+k}{m} z^k, \quad m \ge 0$$

1.95 lines 4-8 27

move the copy for each step to the left next to the step numbers (standard format, see e.g. Algorithm E on p2)

1.98 line -4 28

Σ ⇝ Σ_k

1.101 lines 3 and 4 after Fig. 11 29

x ; that ⇝ x — that
values, we ⇝ values — we

1.102 line 5 30

distribution, the ⇝ distribution, we can improve significantly on Chebyshev's inequality: The

1.110 line after (13) 31

$f^{(2k+1)}(x)$ tends ⇝ $f^{(2k+1)}(x)$ and $f^{(2k+3)}(x)$ tend

1.130 line 11 32

c ⇝ C (Roman, not italics)

1.133 line 20 33

records ⇝ blocks

1.133 line 5 (two places) 34

record ↝ block

1.136 row 5 column 4 of the table 35

1+*T* ↝ *1+T*

1.144 Fig. 14 in both steps P7 and P6 36

P R I M E [K] ↝ PRIME[K]

1.144 line 4 37

fix broken type in the [of PRIME[M]

1.150 line 9 38

delete the exclamation point (!)

1.152 **ex** 3, first line of program 39

X+1 ↝ X+1 (0)

1.157 last line of **ex** 18 40

assume ↝ assume that

1.164 **line 5** 41

insert more space after the period, this line's too narrow

1.171 line no. 21 of the program 42

PERM+1 . . . ↝ PERM+1, . . .

1,180 line 8 43

itself " ⇝ itself."

1,180 top of page 44

the "1" is broken in "1.3.3"

1,208 line 14 45

the 0 is broken

1,226 line 16 46

O. J . ⇝ O.-J. --.

1,227 line -10 47

. print) ⇝ print),

1,237 Fig. 3(a) 48

delete the funny little box which appears between "third from top" and "fourth from top"

1,238 just after (1) 49

remove black speck

1,239 lines -3 and -2 50

delete the sentence "Is there . . . obtainable?"

1,243 bottom line 51

TOP ⇝ TOP (twice)

1.244 line 3 — 52

<L< ⇝ <L<

1.245 after step names G1 and G2 — 53

broken typo Γ for [

1.248 line -1 — 54

BASE, BASE+1, BASE+2, ⇝ BASE+1, BASE+2, BASE+3,

1.255 in (10) — 55

move the heavy bar to the right so that it is aligned vertically with the heavy bar in (11)

1.264 comment for line 18 of the program — 56

T3 ⇝ T4

1.265 new paragraph before the exercises — 57

In spite of the fact that Algorithm T is so efficient, we will see an even better algorithm for topological sorting in Section 7.4.

1.275 changes to Program A — 58

line 04: 6H ⇝ 1H
line 05: becomes line 06
line 06: becomes line 07
line 07: becomes line 05, and delete the "1H" and change x ⇝ $1+m$''
line 12: becomes the following two lines

```
12   LD2    1:3(LINK)        q´   Q←LINK(Q1).
13   JMP    2B               q´   Repeat.
```

lines 13 - X become lines 14-36
change 6B ⇝ 1B in what was line 17 (now line 18)

1.276 line -1 59

b^3. ⇝ b^3-1.

1.276 line -4 60

exceed *b* ⇝ exceed *h - l*

1.276 line 12 61

29 ⇝ 27 (twice)

1.284 Table 1, left column 62

the line for time 0200 is out of place, it belongs just before the line for time 0256

1.286 Fig. 12 63

. the shading in this figure mysteriously disappeared from the 3rd column of nodes, in the second edition. (First edition was OK!)

1.297 line 7 64

2419200 ⇝ 2,419,200

1.299 two lines before (11) 65

' is the lowest value ⇝ points to the bottom-most value

1.304 exercise 20 line 3 66

A(I,J) ⇝ A[I,J]

1,304 new exercise 67

21. [*20*] Suggest a storage allocation function for $n \times n$ matrices where n is variable. The elements A[I,J] for $1 \le I,J \le n$ should occupy n^2 consecutive locations, regardless of the value of n.

1,322 tree illustration near bottom of page 68

the number "(9)" must be inserted at the right of this diagram

1,325 line -13 69

P* ⇝ P*

1,333 between (2) and (3) 70

tilt the diagram 45° and we have ⇝
tilt the diagram and bend it slightly, obtaining

1,336 Fig. (7) 71

the photographer has lined up the two parts of this figure improperly in this edition; the left-hand half of the illustration should be lowered so that the trees are flush at the bottom -- this means that corresponding letters will be on the same line in both left and right parts of the illustration

1,336 line -9 72

of (7) ⇝ of the left-hand tree in (7)

1,349 line 16 73

node to ⇝ node with

1,353 line 9 74

only upward links are sufficient ⇝ upward links are sufficient by themselves

1.357 in (17) 75

delete "." outside the boxes (far consistency in style)

1.360 exercise 11 76

change script ℓ to italic l in five places (lines 5,6,6,23,25)

1.363 Theorem A part (a) 77

; ↝ .

1.375 line 4 78

remove hairline between "fin" and "("

1.395 line -3 79

o r it ↝ o r

1.396 line -2 80

Exercise ↝ exercise

1.405 exercise 12 81

Suppose ↝ [*20*] Suppose

1.406 line -14 82

partic lar ↝ particular

1.427 line -4 83

3.). ↝ 3).

1,429 Fig. 40 *84*

the shape of the box containing B6. should have rounded sides (like that of B2); on the other hand, the box that says "Error" should be rectangular

1,445 line -5 85

this displayed line should be raid half a space so that it is separated from line -4 by the same amount as it is separated Cram lino -6

1,447 **lines 4,6,7** *86*

audition ⤳ condition
emergencies ⤳ emergencies.
hence ⤳ Hence

1,450 line -7 87

two level ⤳ two-level

1,455 exercise 39 line 3 88

$N(n,m)$ ⤳ $N(n,m)/n$

1,457 line 18 89

2 ⤳ **2,2**

1,463 first line of quote 90

me that ⤳ ***me . . . that***

1.465 exercise 3 *91*

line 3: let r be ⇝ let m be
line 4: If $r = 0$, ⇝ If $m = 0$,
line 5: n/r ⇝ n/m
r and let m be ⇝ m and let n be
lines 6 and 7 (steps F4 and F5) *deleted*
line 8: **F6.** ⇝ F4.

1.468 better answer to exercise 3 *92*

3. $-1/27$, but the text hasn't defined it.

1.468 exercise 13 *93*

first sentence should become:
Add "$T \le 3(n-d)+k$" to assertions A3, A4, A5, A6, where k takes the respective values 2,3,3,1.

1.468 line 16 *94*

elements a and b ⇝ elements, $a < b$,

1.470 exercise 3 *95*

the value 3 in ... *two* n^2. ⇝ $n^2 = 3$ occurs for no n, and in the second place $n^2 = 4$ occurs for *two* n.

1.470 line 10 *96*

388. ⇝ 388; V. S. Linskiĭ, *Zh. Vych. Mat. i Mat. Fiz.* **2** (1957), 90-119.

1.470 new answer replacing answer 10 *97*

9.10. No, the applications of rule (d) assume that $n \ge 0$. (The result is correct for $n = -1$ but the derivation isn't.)

1,478 exercise 41 line 4 *98*

$1/4 \rightsquigarrow 1/R$ (twice)

1,485 exercise 31 *99*

We have $\rightsquigarrow$ [This sum was first obtained in closed form by J. F. Pfaff, ***Nova acta** acad.* ***scient. Petr.*** 11 (1797), 38-57.] We have

1,486 and extending to page 487 100

change B to B (Roman type) in the solutions to exercises 40, 41 (twice), 42, 48 (twice).

1,490 exercise 14 ***101***

$n+4 \rightsquigarrow n+1$

1,494 exercise 10 line 2 *102*

(25) $\rightsquigarrow$ *(17)*

1,494 exercise 15 ***103***

Line 1: $zG_{n-2}(z), \rightsquigarrow zG_{n-2}(z)+\delta_{n0},$
line 3 (the displayed formula): delete the period, then add a new line:
when $z \ne -1/4$; $G_n(-1/4) = (n+1)/2^n$ for $n \ge 0$.

1,498 bottom of page, a new answer to exercise 1.2.11.2-3: *104*

3. $|R_{2k}| < |B_{2k}|/(2k)!| \int_1^n |f^{(2k)}(x)|dx$. [C. H. Reinsch observes that $R_{2k} = \int_1^n (B_{2k+2} - B_{2k+2}(\{x\}))\, f^{(2k+2)}(x)dx/(2k+2)!$, and that $B_{2k+2} - B_{2k+2}(\{x\})$ always lies between 0 and $(2-2^{-2k-1})B_{2k+2}$. Therefore if $f^{(2k+1)}(x)$ but not $f^{(2k+3)}(x)$ tends monotonically to zero, (13) still holds for some θ with $0 < \theta < 2 - 2^{-2k-1}$.]

1,500 exercise 6 **105**

$O(n^{-3}) \rightsquigarrow O(n^{-3}))$

1.502 exercise 14 *106*

line 3: MOVE ⇝ MOVE
line 4: JSJ*+1 ⇝ JSJ *+1

1.502 exercise 17(b) *107*

(Using assembly . . . section.) ⇝
(A slightly faster, but quite preposterous, program uses 993 STZ's: JMP 3995; STZ 1,2; STZ 2,2; . . . ; STZ 993,2; J2N 3999; DEC2 993; J2NN 3001; ENN1 0,2; JMP 3000,1.)

1.502 exercise **18** add new sentence: *108*

(Unless the program itself appears in locations 0000-0015.)

1.502 **exercise** 20 *109*

Fukuoka) ⇝ Fukuoka.)

1.502 exercise 16 line 1 110

(49): ⇝ (49);

1.504 new line just before answer no. 23: *111*

For small byte size, the entries $\pm 6^{13}$ would not appear.

1.506 exercise 6 line 3 *112*

$\sqrt{n}$ ⇝ $\sqrt{N}$

1.517 line -13 *113*

e.g. the ⇝ e.g., the

1.519 exercise 22(d) 114

Since the α's are independently chosen, the $\leadsto$ The

1.520 exercise 23 *115*

line 1: $\int_0^1 \ldots$ (l n t). $\leadsto$ $\int_0^\infty \exp(-t-E_1(t))dt$, where $E_1(x) = \int_x^\infty e^{-t}dt/t$.

line 4: ln n / e^γ $\leadsto$ $e^{-\gamma}$ ln n

line 6: 8310...; $\leadsto$ 8 3 1 0 0 83724 41796+ [*Math. Comp.* 2 2 (1968), 411-415];

1.520 line 6 *116*

dev $\sqrt{1/m}$). $\leadsto$ dev $\sqrt{1/m}$), when $n \geq 2m$.

1.529 line 5 *117*

process would loop indefinitely; $\leadsto$
algorithm breaks dawn (possibly refers to buffer while I/O is in progress);

1.533 exercise 9 *118*

in reverse, we can get the inverse $\leadsto$
backwards, we can get the reverse of the inverse of the reverse

1.534 exercise 12 *119*

$0 < \alpha < 1$ $\leadsto$ $|\alpha| < 1$

1.534 line 12 *120*

$r_2(z)$ $\leadsto$ $r_2(z)x$

1.535 exercise **4(ii)** should have the following answer instead: *121*

(ii) **LDA** X,7:7(0:2).

1.541 new answer *122*

13. D. J. Kleitman has shown that $\lim_{n\to\infty} 2^{-n} \log f(n) = \lim_{n\to\infty} 2^{-n} \log \prod_{0\leq k\leq n}\binom{n}{k}!$. [To appear.]

1.542 **line -5** *123*

COUNT ⇝ COUNT

1.543 and also page 544, answer to exercise 24 *124*

replace lines 85-87 of the MIX program by
ST6 X, 1 (QLINK) QLINK[rI1] ← k.
Then renumber lines 88-118 to 86-116.
Finally *delete* "Note: When the ... as the loop." on p. 544.

1.543 lines 11-12 change to (with same indentation): *125*

T10. If P ≠ A, set QLINK[SUC(P)] ← **k**, P ← NEXT (P), and repeat this step.

1.549 exercise 16 *126*

line 2 : 29Σ ⇝ 27Σ (twice)
line 8: 6 ⇝ 4

1.549 **line** -4 insert new sentence (no new paragraph) *127*

[See exercise 5.2.3-29 for a faster algorithm.]

1.550 exercise 1 line 4 *128*

AVAIL ⇝ Y ← INFO(P); AVAIL

1.554 line 2 *129*

COL (P) ⇝ COL(P0)

1,556 change answer 18 (saving space for new answer 21): *130*

the first part up to "after the final" can be shortened as follows.
18. The three pivot steps, in respective columns 3,1,2, yield respectively
( ), ( ), ( );
(use the same matrices as before but squeeze onto one line)

1,556 exercise 20 *131*

A(1,1) ⤳ A[1,1]

1,556 new answer *132*

21. For example, M ← max(I,J), LOC(A[I, J]) = LOC(A[1,1]) + M(M−1) + I − J. (Such formulas have been proposed independently by many people. A. L. Rosenberg and H. R. Strong have suggested the following *k*-dimensional generalization: LOC(A[I_1, . . . , I_k]) = L_k where L_1 = LOC(A[1, . . . , 1]) + I_1 − 1, L_r = L_{r-1} + $(M_r-1)^r$ + $(M_r - I_r)(M_r^{\,r-1} - (M_r-1)^{r-1})$, where M_r = max(I_1, . . . , I_r). [*IBM Tech. Disclosure Bull.* 14 (1972), 3026–3028.])

1,558 exercise 15 *133*

remove blotches in first and second lines

1,560 exercise 12 line 2 *134*

A[*m*]. ⤳ A[*m*],

1,560 new answer 135

13. (Solution by S. Araujo.) Let steps T1 through T4 be unchanged, except that a new variable Q is initialized to A in step T1; Q will point to the last node visited, if any. Step T5 becomes two steps: '**T5.** [Right branch done?] If RLINK(P) = A or RLINK(P) = Q, go on to T6; otherwise set A ⇐ P, P ← RLINK(P) and return to T2. **T6.** [Visit P.] "Visit" NODE(P), set Q ← P, and return to T4.' A similar proof applies.

1,563 line 15 *136*

LOC(T). ⤳ LOUT).

1.566 exercise 1 line 1 *137*

consist $\rightsquigarrow$ consists

1.567 exercise 12 line 2 *138*

INFO(P2)-1 $\rightsquigarrow$ TREE(INFO(P2)-1)

1.573 exercise **18** line 5 ***139***

preorder $\rightsquigarrow$ postorder

1.574 exercise 7 ***140***

the diagrams for Case 1 have two arrowheads in the wrong direction ... the arrows should lead ***away from*** α' and ***towards*** β' both Before and After

1.574 **line -8** ***141***

332 $\rightsquigarrow$ 322

1.575 exercise 12 line 5 *142*

$a(i)$ set $a(i) \leftarrow c(i,j)$ and $b(i) \leftarrow j$; $\rightsquigarrow$
$a(j)$ set $a(j) \leftarrow c(i,j)$ and $b(j) \leftarrow i$;

1.577 exercise 16 *143*

line 2 : the existence of $\rightsquigarrow$ tracing out
lines 4,5: we have an oriented subsubtree $\rightsquigarrow$ the stated digraph is an oriented tree
line 5: configuration $\rightsquigarrow$ digraph
line 6 : subtree $\rightsquigarrow$ tree

1.578 last line *144*

D. E. Knuth, $\rightsquigarrow$ R. Dawson and I. J. Good, *Ann. Math. Stat.* 28 (1957), 946-956; D. E. Knuth,

1,580 exercise 24 line 2

C' ⇝ *C*

1,580 last line of exercise 23, add: *146*

[For $m = 2$ this result is due to C. Flye Sainte-Marie, *l'Intermédiaire des Mathématiciens* **1** (1894), 107-110.]

1,581 exercise 3 line 3 *147*

upper ⇝ right

1,584 exercise 10 *148*

height ⇝ weight (three times)

1,590 second-last line before exercise 6 *149*

this line isn't right-justified, add space after the semicolon

1,594 bottom line 150

exhausted. ⇝ exhausted. [See Guy L. Steele Jr., *CACM* **18** (1975), 495-508, and P. Wadler, *CACM* **19** (1976), to appear, for further information.]
[Note that there's no comma between Steele and Jr. in his name.]

1,594 lines 19-21 replace by 151

Several beautiful List-copying algorithms which make substantially weaker assumptions about List representation have been devised. See D. W. Clark, *CACM* **19** (1976), to appear, and J. M. Robson, *CACM* **19** (1976), to appear.

1,596 line 4 *152*

miniscule ⇝ minuscule

1.603 line before' exercise 34 *153*

165.] $\rightsquigarrow$ 165. Sco also B. Wcgbrcit, *Comp. J.* **15** (1972), 204-208; D. A. Zave, *Inf. Proc. Letters* 3 (1975), 167-169.]

1.608 **line -7** *154*

det (A) $\rightsquigarrow$ $\det(A)$

1.609 in several places 155

change $\cdots$ to . . . in the dcfinitians of x upper k, x lower k, n factorial, and Stirling numbers of bath kinds

1.610 bottom line *156*

give section rcfcrcncc 1.2.5 in right-hand column

1.610 definition of Beta function *157*

B $\rightsquigarrow$ B

1.614 line -20 (the entry for 1 degree of arc) 158

1154 $\rightsquigarrow$ *1155*

1.615 insert new paragraph after line 7: 159

See tha answer to exercise 1.3.3-23 for the 40-digit value of another fundamental constant.

1.617L last line *160*

9 $\rightsquigarrow$ 9.

1,617R 161

Araujo, Saulo, 560.

1,618R 162

Bendix G20, 120.

1,619L 163

Briggs, Henry, 26.

1,619L Bolzano entry 164

delete "theorem,"

1,619R 165

Carlyle, Thomas, xvi.

1,619R 166

Christie Mallowan, Dame Agatha Mary Clarissa (Miller), xix.

1,619R 167

Chu Shih-Chieh, 52, 58.

1,619R 168

Clark, Douglas Wells, 594.

1,619R Chebyshev's inequality entry 169

add p. 102

1,621L 170

Dawson, Reed, 578.

1,621R 171

Doyle, Sir Arthur Conan, 463.

1,622R 172

Even, Shimon, 239.

1,623L 173

Flye Sainte-Marie, Camille, 580.

1,623L 174

delete Fisher, David Allen

1,623R 175

Hamlet, Prince of Denmark, 228.

1,623R entry for Good, Irving John . 176

add p. 578

1,624R line -8 177

Exercise ↝ exercise

1,625R 178

Kleitman, Daniel J., 541.

1,625R Knopp entry 179

add p.494

1,625R Krogdahl entry 180

fix broken type

1,625R line 1 181

20 ⤳ 20.

1,626L 182

Linskiĭ, V. S., 470.

1,628R 183

Path length, 399–405.

1,629L 184

Pfaff, Johann Friedrich, 485.

1,629L 185

Philco S2000, 120.

1,629L 186

Poirot, Hercule, xix.

1,630L 187

RCA 601, 120.

1,630R 188

Rosenberg, Arnold Leonard, 556.

1,630R Robson entry 189

add p. 594

1,631L 190

Shakespeare, William, 228, 465.

1,631L 191

delete Shih-chich, Chu entry

1,631R 192

Steele Jr., Guy Lewis (=Quux), 594.

1,632L 193

Strong, Havcy Raymond, Jr., 556.

1,632R 194

Tarjan, Robert Endre, 239.

1,633R 195

Wadlar, Philip Lee, 594.

1,633R last line 196

delete "theorem," (saves one line)

1,634L 197

Wegbreit, Eliot Ben, 603.

1,634L 198

Wise, David Stephen, 434, 595.

1,634R 199

Zave, Derek Alan, 90, 603.

1,636 (namely the endpapers of the book) 200

delete "Table 1"
also make the change specified for page 136

3,V line 4 of the Preface 201

system $\leadsto$ systems

3,VII line 4 202

forcing himself $\leadsto$ being encouraged

3,IX line 10 203

answer $\leadsto$ answers

3,XII 204

raise this illustration about 3/8 inch

3.1 making the quotation format more consistent 205

line 5: The Prince ⤳ The Prince
line 10: MASON (The Case . . . 1951) ⤳
MASON, in *The Case of the Angry Mourner* (1951)

3.22 new exercises 206

▸**21.** [*M25*] (G. D. Knott.) Show that the permutation $a_1 \ldots a_n$ is obtainable with a stack, in the sense of exercise 2.2.1-5 or 2.3.1-6, if and only if $C_j \le C_{j+1}+1$ for $1 \le j < n$ in the notation of exercise 7.
22. [*M28*] (C. Meyer.) When m is relatively prime to n, we know that the sequence (m mod n)($2m$ mod n) . . . ($(n-1)m$ mod n) is a permutation of $\{1,2,\ldots,n-1\}$. Show that the number of inversions of this permutation can be expressed in terms of Dedekind sums (cf. Section 3.3.3).

3.23 line -9 207

45885 ⤳ 45855

3.64 lines 5-8 after (38) 208

Curiously ... situation to the ⤳ An interesting one-to-one correspondence between such permutations and binary trees, more direct than the roundabout method via Algorithm I that we have used here, has been found by D. Rotem [*Inf. Proc. Letters* **4** (1975), 58-61]; similarly there is a

3.67 insert new sentence after (53): 209

Actually the O terms here should have an extra $\mathcal{H}$ in the exponent, but our manipulations make it clear that this $\mathcal{H}$ would disappear if we had carried further accuracy.

3.72 exercise 28, three changes 210

the average is ⤳ the average l_n is
sorting" for some obscure reason, ⤳ sorting,"
$2\sqrt{n}$; ... $1.97\sqrt{n}$.) ⤳ $2\sqrt{n}$. L. A. Shepp and B. F. Logan have proved that $\liminf_{n\to\infty} l_n/\sqrt{n} \ge 2$ [to appear].)

3.79 figure 9 step 03 *211*

COUNT $[K_j]$ ⤳ COUNT $[K_j]$

3.107 addition to step B2 *212*

(If BOUND = 1, this means go directly to B4.)

3.107 line -5 *213*

the underline shouldn't be broken

3.108 comments for lines 14 and 15 of the program *214*

BOUND ⤳ BOUND

3.120 line 9 *215*

(December, 1974).) ⤳ (1974), 287-289.)

3.134 line 8 *216*

$\log_2$ ⤳ lg

3.135 exercise 15 line 2 *217*

subscripts an superscripts are in wrong font

3.141 line 1 *218*

items; ⤳ items,

3.148 line 3 *219*

r15 ⤳ r15

3,155 line -18 *220*

one ⇝ at least one

3,165 last line of Table 2 *221*

179 ⇝ 170

3,200 line -2 *222*

wise, oracle ⇝ dangerous, adversary

3,200 lines -13, -12, -9, -8 *223*

pronouncements ⇝ outcomes (four changes)

3,200 lines -7 thru -3 *224*

oracles ⇝ adversaries
oracle ⇝ adversary (five changes)

5.3.2 lines 7-23 must be replaced by new copy: 225

Constructing lower bounds. Theorem M shows that the "information theoretic" lower bound (2) can be arbitrarily far from the true lower bound; thus the technique used to prove Theorem M gives us another way to discover lower bounds. Such a proof technique is often viewed as the creation of an *adversary*, a pernicious being who tries to make algorithms run slowly. When an algorithm for merging decides to compare $A_i:B_j$, the adversary determines the fate of the comparison so as to force the algorithm down the more difficult path. If we can invent a suitable adversary, as in the proof of Theorem M, we can ensure that every valid merging algorithm will have to make a rather large number of comparisons. (Some people have used the words 'oracle' or 'demon' instead of 'adversary'; but it is preferable to avoid such terms in this context, since 'oracles' have quite a different connotation in the theory of recursive functions, and 'demons' appear in still a different guise within languages for artificial intelligence.)

We shall make use of *constrained adversaries*, whose power is limited with regard to the outcomes of certain comparisons. A merging method which is under the influence of a constrained adversary does not know about the constraints, so it must make the necessary comparisons even though their outcomes have been predestined. For example, in our proof of Theorem M we constrained all outcomes by condition (5), yet the merging algorithm was unable to make use of this fact in order to avoid any of the comparisons.

The constraints that we shall use in the following discussion apply to the left and right ends of the files. Left constraints are symbolized by

5.3.2 lines 7 and 16 *226*

questions ⇝ comparisons
be answered ⇝ result in

5.3.2 lines 9, 10, 18 227

oracle ⇝ adversary (four changes)

5.3.2 line 12 225

then we define ⇝ thus,
to be ⇝ is

5.3.2 line 15 229

our oracle ⇝ that our adversary

3.202 line 18 230

the oracle ↝ h o

3.202 lines 2, 11, 16, 20, -9, -6 231

oracle ↝ adversary (six changes)

3.203 line 1 232

ORACLE ↝ ADVERSARY

3.204 line 4 233

its ↝ his

3.207 exercise 10, line 2 234

oracle ↝ adversary

3.209 exercise 23 line 6 235

oracle ↝ adversary

3.210 line 2 236

oracle is asked ↝ adversary is about t o decide

3.210 line 3 237

The oracle ↝ H c .

3.210 lines 5, 11, 20, 24, **27, 30** 235

S a y ↝ Decide (six changes)

3.211 line -2 239

"oracle", ⇝ "adversary" as in Section 5.3.2,

3.212 240

line 13: finding an oracle ⇝ constructing an adversary
line 15: oracle declare ⇝ adversary cause
lines 17, 20, 23: oracle ⇝ adversary

3.215 replace the eight lines preceding Table 1 by: 241

may be subject to further improvement. The fact that $V_4(7) = 10$ shows that (11) is already off by 2 when $n = 7$.

A fairly good lower bound for the selection problem has been obtained by David G. Kirkpatrick [Ph.D. thesis, U. of Toronto, 1974], who constructed an adversary which proves that

$V_t(n) \geq$ [illegible] $+ t -$ [illegible] $\cdot \Sigma_{0<j<t-2} \lceil \lg((n+2-t)/(t+j)) \rceil, n \geq 2t-1.$ (12)

Kirkpatrick has also established the exact behavior when $t=3$ by showing that $V_3(n) = n + \lceil \lg((n-1)/2.5) \rceil + \lceil \lg((n-1)/4) \rceil$ for all $n \geq$ SO (cf. exercise 22).

3.217 242

line 17: A. Schönhage ⇝ M. Paterson, N. Pippenger, and A. Schönhage
line 18: has ⇝ have
line -1: (12) ⇝ (13)

3.218 243

line -7: (13) ⇝ (14)
line -5: $V_t(n)$ ⇝ $\overline{V}_t(n)$

3.220 244

line -21: a ***homogeneous*** ⇝ an ***oblivious***
iina -2 arid -1: a homogeneous ⇝ an oblivious
any homogeneous ⇝ any oblivious

3.220 lines 5-6 245

a suitable oracle.] ⤳ an adversary.]

3.220 substitute for exercise 22 246

22. [*24*] (David G. Kirkpatrick.) Show that when $4\cdot 2^k < n-1 \leq 5\cdot 2^k$, the upper bound (11) for $V_3(n)$ can be reduced by 1 as follows: (i) Form four "knockout trees" of size 2^k. (ii) Find the minimum of the four maxima, and discard all 2^k elements of its tree. (iii) Using the known information, build a single knockout tree of size $n-1-2^k$. (iv) Continue as in the proof of (11).

3.221 caption 247

A homogeneous ⤳ An oblivious

3.228 line 3 248

1972), Chapter 15] ⤳ 1973), 163-172]

3.231 upper left corner of Fig. 51 249

there's a dot missing on the second line of the diagram for $n=6$

3.232 line 3 new sentence 250

A. C. Yao and F. F. Yao have proved that $\hat{M}(2,n) = C(2,n) = \lceil \frac{3}{2}n \rceil$ and that $\hat{M}(m,n) \geq \frac{1}{2}n \lg(m+1)$ for $m \leq n$ [*JACM*, to appear].

3.259 line 12 251

16 is in the wrong bold-face font

3.259 line 13 252

RECORD (Q) ⤳ **RECORD (Q)**

3.265 line 10 *253*

delete "[*Hint:* ... 4.5.3L.]" since the proof of that theorem is being changed in the second edition of vol. 2

3.266 line 6 *254*

other *P.* ⇝ other *P.*]

3.294 line 15 *255*

to C5. ⇝ to C5 if *m* > 0.

3.354 lines -16 and -15 256

SORT10 ⇝ SORT10

SORT01 ⇝ SORT01

3.374 bottom line 257

$\lg_2$ ⇝ lg (twice)

3.391 line -3 *258*

"Soundex" ⇝ contemporary form of the "Soundex"

3.397 *259*

line 17: formulated ⇝ popularized

lines 19–20: inversely ... Reading ⇝ approximately proportional to $1/n$. [*The Psycho-Biology of Language* (Boston, Mass.: Houghton Mifflin, 1935); *Human Behavior and the Principle of Least Effort* (Reading

3.408 extra annotation on line 08 of Program B 260

$\lfloor rA/2 \rfloor$. ⇝ $\lfloor rA/2 \rfloor$. (rX changes too)

3.410 line -7 261

only all $\rightsquigarrow$ only if all

3.410 line 13 262

between $\rightsquigarrow$ between and outside the extreme values of the

3.412 (6) 263

$1 \le j \rightsquigarrow 2 \le j$

3.417 line -10 and also line -18 264

800 $\rightsquigarrow$ 500 ---

3.417 line 18 265

memory. It $\rightsquigarrow$ memory. The difference between lg lg N and lg N is not substantial unless N is quite large, and typical files aren't sufficiently random either. Interpolation

3.417 new paragraph after line 14: *266*

Interpolation search is asymptotically superior to binary search; one step of binary search essentially replaces tha amount at watch, n, by $\frac{1}{2}n$, while one step of interpolation search essentially replaces n by $\sqrt{n}$ if the keys in the table are randomly distributed. Hence it can be shown that interpolation search takes about lg lg N steps on the average. (See exercise 22.)

3.417 replace lines -5 thru -11 by: 267

```
01 13 09 34 29 08 08 53 20          49 12 27
01 13 14 31 52 30                   490907 12
01 13 43 40 48                      48 49 41 15
01 13 48 40 30                      48 46 22 59 25 25 55 33 20
01 14042640                         48 36
```

3.418 bottom line 268

was $\rightsquigarrow$ seems to have been

3.419 269

line 11: 1 ⇝ 1,2
line 12: February, ⇝ February

3.419 line 2 270

the last part is in nearly perfect alphabetic order! ⇝
the alphabetic order in the last part is substantially better.

3.422 replate exercise 22 271

22. [*M43*] (A. C. Yao and F. F. Yao.) Show that an appropriate formulation of interpolation search requires asymptotically lg lg N comparisons, on the average, when applied to N independent uniform random keys that have been sorted. Furthermore, *all* search algorithms on such tables must make asymptotically lg lg N comparisons, on the average.

3.424 line -8 272

A). ⇝ A), since the necessary operations are trivial when ROOT = A.

3.428 line 17 273

Algorithm L ⇝ Algorithm T

3.431 lines 7 and 8 274

clearly constructed $n+1$ different deletions; ⇝
constructed $n+1$ different deletions, one for each j;

3.439 275

line 6: A fairly ⇝ An even more
line -7: time. ⇝ time. In fact, M. Fredman has shown that $O(n)$ units of time suffice, if the right data structures are used [*ACM Symp. Theory of Comp.* 7 (1975), 240-244].

3,439 and following pages 276

in the second edition of vol. 3 I must revise the subsection about the Hu-Tucker algorithm to take account of the new Garsia-Wachs algorithm. Meanwhile I could have improved my treatment of Hu-Tucker by leaving the external nodes out of the priority queues (cf. (23) on p. 444, an unnecessarily cumbersome approach).

3,439 replace lines 3-5 by: 277

that the resulting maximum subtree weight, $\max(w(0,k-1),w(k,n))$, is as small as possible. This approach can also be fairly poor, because it may choose a node with very small p_k to be the root; however, Paul J. Bayer has proved that the resulting tree will always have a weighted path length near the optimum (see exercise 36).

3,440 exercise 30 *278*

M46 ⤳ *M41*

%,%&u new version of exercise 36 279

36. [*M40*] (Paul J. Bayer.) Generalizing the upper bound of Theorem G, prove that the cost of any optimum binary search tree with nonnegative weights must be at most the total weight $S = \Sigma_{1\le i\le n}\, p_i + \Sigma_{0\le i\le n}\, q_i$ times $H + 2$, where

$$H = \Sigma_{1\le i\le n}(p_i/S)\lg(S/p_i) + \Sigma_{0\le i\le n}(q_i/S)\lg(S/q_i);$$

in fact, the top-down procedure which repeatedly chooses roots that minimize the maximum subtree weights will yield a binary search tree satisfying this bound. Show further that the cost of the optimum binary tree search tree is $\ge S$ times $H - \lg(2H/e)$.

3,444 diagrams (2) 280

put extra little vertical lines above the topmost nodes (B and X, respectively), for consistency with (1)

3,456 **line 2** 281

K ⤳ *K*

3.460 replace lines -4 and -3 **by:** 282

indicate that the average number of comparisons needed to insert the *N*th item is approximately 1.01 lg *N* • 0.1 except when *N* is mall.

3.461 bottom line of Table 1 283

2.0 ⇝ 2.78

3.462 **Eq. (14)** ***284***

$p/(1-p) \approx 1.851.$ ⇝ $1/(1-p) \approx 2.851.$

3.462 **line -12** 282

k - 1 is $p/(1-p)$. ⇝ k is $1/(1-p)$.

3.463 ***286***

line 1: lg N + 0.25 ⇝ 1.01 lg N + 0.1
line 2: 11.17 lg N + 4.8 ⇝ 11.3 lg N + 3
line 6: 6.5 lg N + 4.1 ⇝ 6.6 lg N + 3

3.463 Figure 24 287

below the third node from the left, the 1 has a bar across it, making it look like a 4 hy mistake

3.464 **line -9** 288

R(P) ⇝ RANK(P)

3.465 **line 16** 289

RANK(R). ⇝ RANK **(R)**. Co to **C10.**

3.468 line -4 290

(unpublished) ⇝ [see Aho, Hopcroft, and Ullman, *The Design and Analysis of Computer Algorithms* (Reading, Mass.: Addison-Wesley, 1974), Chapter 4]

3.468 lines 11-12 should be replaced by: *291*

trees which arise when we allow the height difference of subtrees to be at most k. Such structures may be called HB[k] trees (meaning "height-balanced"), so that ordinary balanced trees represent the special case HB[1]. Empirical tests on HB[k] trees have been discussed by P. L. Karlton et al., *CACM* 19 (1976), 23-28.

3.471 new exercise 292

31. [*34*] (M. L. Fredman.) Invent a representation of linear lists with the property that insertion of a new item between positions $m-1$ and m, given m, takes $O(\log m)$ units of time.

3.479 new paragraph before the exercises 293

Andrew Yao has proved that the average number of nodes after random insertions without the overflow feature will be $N/(m \ln 2) + O(N/m^2)$, for large N and m, so the storage utilization will be approximately $\ln 2 \approx 69.3$ percent [*Acta Informatica*, to appear].

3.480 line 11 294

long, ⇝ long, but always a multiple of 5 characters,

3.481 line -10 295

tree. ⇝ trie.

3.490 line -4 296

HOUSE ⇝ HOUSE (twice)

3,490 line 5 — 297

the nodes of the tree ⇝ the tree is nonempty and that its nodes

3,492 line 19 — *298*

follows. ⇝ follows:

3,500 exercise 4 — *299*

there will be a new illustration, with positions numbered from 1 to 49 instead of 1 to 55.
The respective entries will be:

---	(20)	---	WAS	THAT	(18)	OF
BE	THE	HIS	WHICH	WITH	THIS	---
(4)	ON	I	HE	A	OR	(19)
(3)	TO	HAD	---	(14)	BUT	(1)
(17)	FOR	BY	IN	FROM	ANO	NOT
(1)	HER	ARE	IS	IT	AS	AT
(7)	---	HAVE	(3)	---	YOU	---

line 2 : 55 ⇝ 49

lines after new illustration: 20,1,14,..., 2 within ⇝ 20,19,3,14,1,17,1,7,3,20,18,4 within

3,503 line 2 — 300

that ⇝ that, if $n \geq 2$,

3,504 exercise 39 — 301

M47 ⇝ *M43*

3,513 line -5 add new sentence after **"of M."** — 302

(A precise formula is worked out in exercise 34.)

3,520 program line 13 — 303

empty ⇝ nonempty

3.521 304

delete lines 15-18 ($m+1$ not really needed after all)

3.522 line -1 305

antially $\rightsquigarrow$ stantidly

3.527 line 13 306

similar $\rightsquigarrow$ weaker

3.528 three lines after (34) 307

purposes. $\rightsquigarrow$ purposes. In fact, Leo Guibas and Endre Szemerédi have succeeded in proving the difficult theorem that double hashing is asymptotically equivalent to uniform probing, in the limit as $M \to \infty$. [To appear.]

3.529 just after (37), insert new sentence: *308*

By convention we also set $f(0,0) = 1$.

3.537 new formula for (58) 309

$$C_N = 1 + (e^{-b\alpha} b^b \alpha^b)/2b!)\ (2 + (\alpha-1)b + (\alpha^2+(\alpha-1)^2(b-1))R(\alpha,b)) + O(1/M).$$

3.541 line -2 310

until Morris's... 1968, $\rightsquigarrow$ until the late 1960's,

3.542 *311*

line 1: The only ... among $\rightsquigarrow$ The first published appearance of the word seems to have been in H. Hellerman's book *Digital Computer System Principles* (New York: McGraw-Hill, 1967), p. 152; the only previous occurrence among

line 6: 1968 $\rightsquigarrow$ 1967

3.542 exercise 5 lines 3 and 4 *312*

ten or less ⇝ at most ten

3.543 exercise 10 *313*

$M48$ ⇝ $M43$

3.546 line -4 *314*

$M' > M$ ⇝ $M' \geq M$

3.547 exercise 45 *315*

$M48$ ⇝ $M43$

3.549 exercise 66 *316*

66. ⇝ ▸**66.**

3.549 new exercise *317*

67. [*M25*] (Andrew Yao.) Prove that all fixed-permutation single-hashing schemes in the sense of exercise 62 satisfy the inequality $C'_N \geq \frac{1}{2}(1 + 1/(1-\alpha))$. [*Hint:* Show that an unsuccessful search takes exactly k probes with probability $p_k \leq (M-N)/M$.]

3.554 lines -10, -8, -6 *318*

LONGITUDE ⇝ LONGITUDE (three places)

3.555 *319*

in the second edition I will be revising Section 6.5 again, deleting the material on post-office trees, paying more attention to Bentley's ***k-d*** trees, and discussing the search procedure of Burkhard and its analysis by Dubost and Trousse (cf. Stanford CS report of Sept. 1975)

3.555 line 13, add: 320

[*CACM* 18 (1975), 509-516.]

3.555 line 8 321

3 (to appear) ⤳ 4 (1974), 1 - 1 0

3.561 the numbers in (5) should be respectively: 322

.07948358; .00708659; .00067094; .00006786; .00000728; .00000082.

3.571 quotation 323

Alice's Adventures in Wonderland ⤳ *Alice's* Adventures *in Wonderland*

3.576 lines 1-3 324

So WC may . . . (p-1)/2. ⤳
In general if f is any divisor of $p-1$ and d any divisor of $\gcd(f,n)$, we can similarly determine $(n/d) \bmod f$ by looking up the value of $b^{(p-1)/f}$ in a table of length f/d. If $p-1$ has the prime factors $q_1 < q_2 < \cdots \le q_t$ and if q_t is small, we can therefore compute n rapidly by finding the digits from right to left in its mixed-radix representation, for radices $q_1, \ldots, q_t$. (This idea in tluc to R. L. Silver.)

3.579 exercise 6 325

the l's in the exponents ride too high (twice)

3.580 exercise 13 326

b_{m-1}, ⤳ b_{m-1}, b_{m+1},

3.581 exercise 20 327

Zolnowsky (to appear). ⤳ Zolnowsky, *Discrete Math.* 9 (1974), 293-298.

3.581 new answer *328*

22. $\lfloor mj/n \rfloor - \lfloor mi/n \rfloor - \lfloor m(j-i)/n \rfloor = 0$ or 1; and it is 0 iff $mj \bmod n > mi \bmod n$. Hence the number of inversions is $\sum_{0<i\le j<n} (\lfloor mj/n \rfloor - \lfloor mi/n \rfloor - \lfloor m(j-i)/n \rfloor) = \sum_{0<r<n} \lfloor mr/n \rfloor (r - (k-r) - (k-r-1))$, which can be transformed to $\frac{1}{4}(n-1)(n-2) - \frac{1}{4}n\sigma(m,n,0)$. [*J. für die reine und angew. Math.* 198 (1957), 162-166.]

3.592 exercise 19 329

delete lines 3-7 of this answer.
line 8: The answer $\leadsto$ (This formula
now add a new paragraph:

Note: A general formula for the number of ways to place the integers $\{1,2,\ldots,n\}$ into an array which is the "difference" of two tableau shapes $(n_1,\ldots,n_m) \setminus (l_1,\ldots,l_m)$, where $0 \le l_i \le n_i$ and $n = \sum n_i - \sum l_i$, has been found by W. Feit [*Proc. Amer. Math. Soc.* 4 (1953), 740-744]. This number is $n! \det (1/((n_j-j) - (l_i-i))!)$.

3.597 line -4 330

$4.5N^2 + 2.5N - 6$. $\leadsto$ $(4.5N^2 + 2.5N - 6)u$.

3.599 addendum to exercise 15 *331*

It is interesting to note that $C(w,z) = F(-wz,z)/F(-w,z)$, where $F(z,q) = \sum_{n\ge 0} z^n q^{n^2} / \prod_{1\le k\le n}(1-q^k)$ is the generating function for partitions $p_1 + \cdots + p_n$ into n parts, where $p_j \ge p_{j+1}+2$ for $1 \le j < n$ and $p_n > 0$ (cf. exercise 5.1-16).

3.602 exercise 31 line 03 332

INPUT-N,4 $\leadsto$ -INPUT-N,4

3.605 addition to answer 2 333

[Algorithm 5.2.3S does exactly xch(π) exchanges, see exercise 5.2.3-4.]

3.605 line 12 334

,to appear]. $\leadsto$ 1 1 (1975), 29-35].

3.611 bottom two (clobbered) lines should start respectively thus: 335

43. As $a \to 0+$,
$\Gamma(1)) / a \to \Gamma'(1) = -\gamma$,

3.612 line -4 336

ζ is in wrong font (see line -2 for correct ζ)

3.633 exercise 13 337

397-404 $\rightsquigarrow$ 263-269

3.633 line-s -8, -7, -6, -3 33s

oracle $\rightsquigarrow$ adversary

3.634 lines 2, 17 339

oracle $\rightsquigarrow$ adversary

3.635 exercise 9 340

comparisons.) $\rightsquigarrow$ comparisons, yet the procedure is not optimal.)

3.636 exercise 14 341

line 1 : found in $\rightsquigarrow$ found in $U_t(n) \leq$
also add nnw sentence: (Kirkpatrick's adversary actually proves that (12) is a lower bound for $U_t(n+1) - 1$.)

3.637 line 2 342

oracle $\rightsquigarrow$ adversary

3.637 new answer 343

22. In general when $2^r \cdot 2^k < n+2-t \le (2^r+1)\cdot 2^k$ and $t < 2^r \le 2t$, this procedure starting with $t+1$ knockout trees of size 2^k will yield $\lfloor (t-1)/2 \rfloor$ fewer comparisons than (11), since at least this many of the comparisons used to find the minimum in (ii) can be "reused" in (iii).

3.640 exercise 36 last line 344

to appear.] ⇝ 333-339.]

3.640 insert new paragraph before line -2: 345

G. Baudet and D. Stevenson have observed that exercises 37 and 38 combine to yield a simple sorting method with $(n \lg n)/k + O(n)$ comparison cycles on k processors: First sort k subfiles of size $\le \lceil n/k \rceil$, then merge them in k passes using the "odd-even transposition merge" of order k. (To appear.)

3.651 exercise 2 line 4 346

D8 ⇝ C8

3.664 new answer 347

10. See *Proc. ACM Symp. Theory of Computing* 6 (1974), 216-229.

3.665 exercise 3 for section 5.5, last line 348

variables. ⇝ variables, without transforming the records in any way.

3.667 line -6 349

Strauss ⇝ Straus

3.672 exercise 7 line 3 350

80]. ⇝ 80; see also L. Guibas, *Acta Informatica* 4 (1975), 293-298.]

3.673 351

line -9 (displayed nodes):

$r_1 \leadsto r_0$

$r_2 \leadsto r_1$

$s_1 \leadsto s_0$

$s_2 \leadsto s_1$

line -8:

$r_1 \leadsto r_0$

$s_1 \leadsto s_0$

$k > 1 \leadsto k > 0$

lines -6 and -5: the right subtrees of . . . and the result $\leadsto$ the result

3.674 new answer 352

30. This has been proved by Russell Wessner [to appear].

3.674 replace answer to 36 by: 353

36. See *MAC Tech. Memo.* 69 (M.I.T., November 1975), 41 pp.

3.676 exercise 19 354

the fourth rectangle in the left-hand figure is too short -- it should be extended so that its bottom line is at the same level as the bottom of the first and third rectangles

3.677 answer 20, the line following the tree should become: 355

It may be difficult to insert a new node at the extreme left of this tree.

3.678 answer 30 line 4 356

left subtree of that $\leadsto$ subtree rooted at that

3.678 new answer 357

29. Partial solution by A. Yao: With $N \geq 6$ keys the lowest level will contain an average of $\frac{2}{7}(N+1)$ one-key nodes, $\frac{1}{7}(N+1)$ two-key nodes. The average total number of nodes lies between $0.70N$ and $0.79N$, for large N. [*Acta Informatica*, to appear.]

3.678 new answer 358

31. Use a nearly balanced tree, with additional upward links for the leftmost part, plus a stack of postponed balance factor adjustments along this path. (Each insertion does a bounded number of these adjustments.)

3.680 exercise 4 line 3 359

IONIC ⇝ TRASH
seven ⇝ *six*
insert new sentence on last line: [This remarkable 49-place packing is due to J. Scot Fishburn, who showed that 48 places do not suffice.]

3.682 new answer to exercise 11 (extends to p. 683) 360

11. No; eliminating a node with only one empty subtree will "forget" one bit in the keys of the nonempty subtree. To delete a node, it should be replaced by one of its ***terminal*** descendants, e.g., by searching to the right whenever possible.

3.683 exercise 12 361

line 3: Algorithm 6.2.2D. ⇝ the algorithm suggested in the previous answer.
last line: $\frac{3}{8}$ ⇝ $\frac{1}{4}$

3.686 exercise 34 line 1 362

$B_k 2^{j(k-1)}$ ⇝ $B_k/2^{j(k-1)}$

3.686 exercise 34, new answer to part (b) 363

(b) In the $1/(e^x-1)$ part, it suffices to consider values of j with $x \le 2 \ln n$. For $1 \le x \le 2 \ln n$ we have $\sum_{1<k<n/x} (1-kx/n)^{n-1} = \sum_{k\ge 1} e^{-kx} + O(x^2 e^{-x}/n)$. For $x \le 1$ we have $\sum_{0\le k<n} \binom{n}{k} B_k (x/n)^k = \sum_{k\ge 0} B_k\, x^k/k! + O(x^2/n)$.

3.686 line -9 364

$+f(n),$ ⇝ $+f(n)+2/n,$
$k\le 1$ ⇝ $k\ge 1$

3.687 new answer 365

39. See Miyakawa, Yuba, Sugito, and Hoshi, *SIAM J. Computing*, to appear.

3.691 line **12** 366

and $\rightsquigarrow$ with O/O . 1 when $k = N = M-1$, and

3.694 *367*

in the second edition I will revise several of these answers, using Mike Paterson's simplified new approach to such analyses

3.694 exercise 39 *368*

line 6 (third line of displayed formulas): delete "$j>1$," (on this line only)
line 6 (fourth line of displayed formulas): $\binom{j}{2} \rightsquigarrow \Sigma_{j\geq 1}\binom{j}{2}$ (two places)

3.696 new answer 369

46, Yes. See L. Guibas (to appear).

3.699 **new answer** 370

67. Let $q_k = p_k+p_{k+1}+\cdots$; then $C'_N = \Sigma_{k>1}\, q_k$ and $q_k \geq \max(0, 1-(k-1)(M-N)/M)$.

3.700 **line 1** *371*

$\Sigma_t\, p_t P_t.$ $\rightsquigarrow$ $\Sigma_t\, p_t P_t$, minus the probability that a particular record is a "true drop", namely $\binom{N-q}{r-q} / \binom{N}{r}$, where $N = \binom{n}{k}$.

3.702 **line -20 last column** 372

1154 $\rightsquigarrow$ *1155*

3.703 after line 7, a new paragraph: 373

A few interesting constants without common names have arisen in connection with the analysis of sorting and searching algorithms; 40-digit values of these constants appear in the answers to exercises 5.2.3-27, 5.2.4-13, and 6.3-27.

3.706 left column 374

det (A) ⇝ det(A)

3.706 line -2 375

... ⇝ ...

3.707 definition of factorial 376

$1 \cdot 2 \cdot \cdots \cdot n$ ⇝ $1 \cdot 2 \cdot \ldots \cdot n$

3.707 definitions of x lower k and Stirling numbers of both kinds 377

... ⇝ ...

3.707 line -12 378

5.1.3. ⇝ 5.1.3

3.720L 379

Adversaries, 200-204, 209, 211-212, 220.

3.720L Aho entry 380

add p. 468

3.711L 381

Baudet, Gerard, 640.

3.711L 382

Bayer, Paul Joseph, 439, 450.

3.712R 383

Dedekind sums, 22.

3.712R 384

Demons, 200.

3.713R 385

Feit, Walter, 592.

3.714L 386

Fishburn, John Scot, 680.

3.714L Fredman entry 387

add pp. 439, 471

3.714R 388

Grasseli ⇝ Grasselli

3.714R Guibas entry 389

add pp. 528, 672, 696

3,725L 390

Hellerman, Herbert, 542.

3,725L *391*

Hoshi, Mamoru, 687.

3,725L 392

delete the entry for "Homogeneous comparisons"

3,725L Hyafil entry 393

delete p. 215

3,725L two new entries 394

H B[k] trees, 468. Height-balanced trees, 468, *see* Balanced trees.

3,726L Knockout tournament entry 395

add pp. 214, 220

3,726L Linear list representation entry 396

468. ∼→ 471.

3,726L two new entries 397

Karlton, Philip Lewis, 468.
Kirkpatrick, David Galer, 215, 220, 636.

3,726R 398

Logan, Benjamin Franklin, Jr., 594.

3.717R 399

Meyer, Curt, 22.

3.717R 400

Miyakawa, Masahiro, 687.

3.718L 401

Oblivious algorithms, 220-221.

3.718L 402

Oracles, 200, *see* Adversaries.

3.718L Parallel **comput** at ion entry 403

add p. 640

3.718R 404

Paterson, Michael Stewart, 217.

3.719L 405

Pippenger, Nicholas John, 217.

3.719L line -1 406

78 $\rightsquigarrow$ vi, 78

3.719R 407

Rotem, Doron, 64.

3.720L *408*

Shepp, Lawrence Alan, 594.

3.720L line -7 *409*

223, ⇝ 223, 405 (exercise 22),

3.720R *410*

Silver, Roland Lazarus, 576.

3.720R Simultaneous comparisons entry *411*

add p. 640

3.721L *412*

Stevenson, David, 640.

3.721L new subentry under Sorting *413*

history of, 382-388, 417-418.

3.721R *414*

Strauss ⇝ Straus

3.721R *415*

Sugito, Yoshio, 687.

3.721R *416*

Szemerédi, Endre, 528.

3,721R 417

Tape searching, 400-401, 405.

3,721R line 25 41s

Slawomir ⇝ Sławomir

3,722L *419*

Treesort, see Tree selection sort, Heapsort.

3,722L 420

Two-dimensional trees, 555, 570.

3,722L Turski entry 421

Wladysław ⇝ Władysław

3,722L Ullman entry 422

add p. 468

3,722L new subentry under Trie search *423*

generalized, 565.

3,722R 424

Wessner, Russell, 674.

3,722R 425

Yao, Foong Frances, 232, 422.

3.722R Wrench entry 426

add p. 686

3.722R Yao, Andrew entry 427

add pp. 232, 422, 479, 549, 678

3.723L 425

Yuba, Toshitsugu, 687.

3.723R 429

Zeta function, 612, 666.

3.723R just before 2-3 trees entry 430

2D trees, 555, 570.

3.726 (namely the endpapers of the book) *431*

delete "Table 1"
also change 1 to italic *1* in box number 35

8.0 changes to MIX booklet 432

p30, Fig. 3: Step P3 should say "500 found?"
p34, Fig. 4: third card should say L EQU 500
p43, line 1 : 6667 ⇝ 66667
p43, line 2: 193,334 ⇝ 133,334
p44, problem 16, line 2: row...diagonal ⇝ row and column
p44, problem 16, line 8: 10 ⇝ 9
and change "record" to "block" everywhere in the discussion of MIX I/O operators.

, 9.1 changes to the book Surreal Numbers 433

p99, line 2: (4) ⇝ (3)
p111, lines 4 and 5, interchange the inside of the braces:
$(\{x-x^2,\ x-x^2+x^3-x^4,\ \ldots\},$
$\{x,\ x-x^2+x^3,\ x-x^2+x^3-x^4+x^5,\ \}).$
p117, problem 18, lines 3 and 4 should be:
X_L has a greatest element or is null if and only if
X_R has a least element or is null.