☆ THE ART OF COMPUTER PROGRAMMING ☆

☆☆☆☆ ERRATA TO VOLUME 4 FASCICLE 6 ☆☆☆☆

This document is a transcript of the notes that I have been making in my personal copy of *The Art of Computer Programming*, Volume 4, Fascicle 6, since it was first printed in 2015.

Four levels of updates — "errors," "amendments," "plans," and "improvements" — appear, indicated by four different typographic conventions:

▸**Page 666** line 1 ———— 04 Jul 1776

Technical or typographical errors (aka bugs) are the most critical items, so they are flagged with a '▸' preceding the page number. The date on which I first was told about the bug is shown; this is the effective date on which I paid the finder's fee. The necessary corrections are indicated in a straightforward way. If, for example, the book says 'n' where it should have said '$n+1$', the change is shown thus:

$n \rightsquigarrow n+1$

Page 666 line 2 ———— 14 Jul 1789

Amendments to the text appear in the same format as bugs, but without the '▸'. These are things I wish I had known about or thought of when I wrote the original text, so I added them later. The date is the date I drafted the new text.

Page 666 line 3 . . . . 20 Nov 1917

Plans for the future represent a third kind of item. In such notes I sketched my intentions about things that I wasn't ready to flesh out further when I wrote them down. You can identify these items because they're written in slanted type, and preceded by a bunch of dots '. . . . .' leading to the date on which I recorded the plan in my files.

Page 666 line 4 ———— 10 Jan 1938

The fourth and final category — indicated by page and line number in smaller, slanted type — consists of minor corrections or improvements that most readers don't want to know about, because they are so trivial. You wouldn't even be seeing these items if you hadn't specifically chosen to print the complete errata list in all its gory details. Are you sure you wanted to do that?

The ultimate, glorious, future editions of Volumes 1–5 are works in progress. Please let me know of any improvements that you think I ought to make. Send your comments either by snail mail to D. E. Knuth, Computer Science, Gates Building 4B, Stanford University, Stanford CA 94305-9045, or by email to `taocp@cs.stanford.edu`*. (Use email for book suggestions only, please — all other correspondence is returned unread to the sender, or discarded, because I have no time to read ordinary email.) Although I'm working full time on*

Volume 4 these days, I will try to reply to all such messages within a year of receipt. Current news about The Art of Computer Programming is posted on

`http://www-cs-faculty.stanford.edu/~knuth/taocp.html`

and updated regularly.

— Don Knuth, March 2006

I thought there would be not much corrections.
I honestly wrote what I thought, but was most grievously mistaken.
I find the style incredibly bad, & most difficult to make clear & smooth.
. . . How I could have written so badly is quite inconceivable,
but I suppose it was owing to my whole attention being fixed
on general line of argument, & not on details.
All I can say is that I am very sorry.

— CHARLES DARWIN, letter to John Murray (14 June 1859)

An opportunity to revise is a great luxury.
There's always something that needs correcting or improving.
And it's not just a matter of my own second thoughts.
Many of the most important changes and additions
start with letters from readers.

— BRIAN HAYES (2017)

I still feel fine, but I'm making more or more little errors.

— DONALD E. KNUTH, letter to Matthew Ginsberg (28 April 2022)

☆☆☆☆☆☆☆☆☆☆☆☆☆☆ SATISFIABILITY ☆☆☆☆☆☆☆☆☆☆☆☆☆☆

Last updated 26 July 2022

▸**Important Changes Throughout!** ——— 28 Jul 2019

The Figures in Volume 4 Fascicle 6 have all received new numbers, now that Volume 4 Fascicle 5 is complete! Figures formerly numbered 33–56 are now numbered 76–99. Figures in the answers, formerly numbered A–5 through A–10, are now numbered A–7 through A–12.

▸**Page i** line −2 ——— 11 May 2016

Saõ Paulo ⇝ São Paulo

Page iii line 18 ——— 26 Jul 2019

combinatorial algorithms ⇝ combinatorial searching

▸**Page v** line 7 ——— 26 Jul 2019

Information Systems Laboratory ⇝ InfoLab

▸**Page v** line 20 ——— 17 Jan 2016

`~/fasc5a` ⇝ `~knuth/fasc5a`

Page v replacement for lines 18–23 ——— 16 Dec 2019

the word "improvement.") That section, together with the introductory material about backtracking and dancing links, can be found in Fascicle 5 of Volume 4.

Page vi line 4 ——— 04 Oct 2018

defined in Section 1.3.1′. ⇝ defined in Section 1.3.1′ of Volume 1, Fascicle 1.

Page vii lines 3 and 11 ——— 16 Dec 2019

00. ⇝ **15.**

Page vii lines 4, 10, 11, 12 ——— 31 May 2017

rows ⇝ options [four changes]
columns ⇝ items [one change]

Page viii line 5 ——— 26 Jul 2019

Basic Backtrack ⇝ Backtrack Programming

Page 3 line −11 ——— 19 Dec 2019

is equivalent to ⇝ can be replaced by

Page 3 line −9 ——— 19 Dec 2019

equivalent to two ⇝ equivalent to the AND of two

Page 4 line −8 ——— 15 Feb 2017

equally spaced 1s ⇝ *equally spaced* 1s (or both)

Page 5 bottom and page 6 top ——— 31 May 2017

row $\rightsquigarrow$ option [three changes]
rows $\rightsquigarrow$ options [four changes]
column $\rightsquigarrow$ item [three changes]
columns $\rightsquigarrow$ items [two changes]

▸**Page 9** line −18 ——— 10 Apr 2016

it can be put into the conjunctive normal form "piece by piece",

Page 14 line 15 ——— 17 Jul 2018

R_{23}^{46}, $R_{23}^{46,1}$ and $R_{23}^{46,2}$ $\rightsquigarrow$ R_{23}^{46}, $R_{23}^{46,1}$, and $R_{23}^{46,2}$

Page 16 insert new paragraph 10 lines before (32) ——— 22 Aug 2021

D. Morgenstern has found a much simpler formula that also matches Table 2:

$$f(x_1,\ldots,x_{20}) \;=\; \bar x_4 x_{10}\bar x_{12} \vee \bar x_6\bar x_{10}\bar x_{12} \vee x_9\bar x_{10}x_{11}.$$

But it's actually further than (27) from the "true" f that's revealed in exercise 53.

Page 19 line −5 ——— 08 Apr 2017

predictable pattern $\rightsquigarrow$ repetitive pattern

Page 28 line 4 after (57) ——— 17 May 2019

Furthermore `C(`l`)` $\rightsquigarrow$ Furthermore, if l is a literal whose value has not yet been fixed, `C(`l`)`

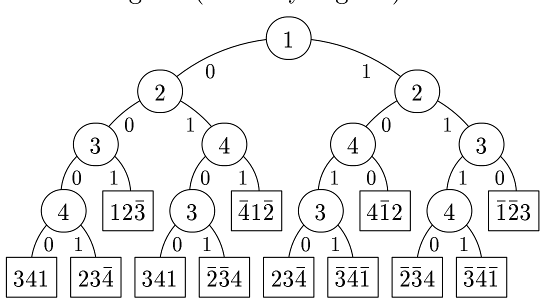
Page 30 line 23 ——— 24 May 2018

we can assume that $\rightsquigarrow$ we can adjust the data so that

▸**Page 32** line −11 ——— 22 Nov 2015

$\bar5\bar3\bar4$ $\rightsquigarrow$ $\bar5\bar3\bar7$

▸**Page 33** replacement for Fig. 82 (formerly Fig. 39) ——— 10 Feb 2016



Page 34 line 2 after (60) ——— 05 Jul 2016

moves. $\rightsquigarrow$ moves (the 4s and the 5s).

▸**Page 41** line 8 after (67) ——— 13 Aug 2018

$h(3)\,h(6)$ $\rightsquigarrow$ $h(4)\,h(6)$

▸**Page 41** lines 4–8 after (67) ——— 27 Apr 2019

$h(2)h(3)$ ⇝ $h(\bar2)h(\bar3)$, $h(3)h(5)$ ⇝ $h(\bar3)h(\bar5)$, etc.
$=$ ⇝ $\approx$

▸**Page 42** line 5 after (70) ——— 01 Oct 2019

$h(\bar4)h(\bar6)$ ⇝ $h(\bar3)h(\bar4)+h(\bar4)h(\bar6)$

▸**Page 43** in (71) ——— 21 Oct 2020

for $1 < j < E$ ⇝ for $1 \le j < E$

▸**Page 43** line 1 after (71) ——— 09 Jul 2020

proto truth, etc. ⇝ proto truths, etc.

▸**Page 44** in step X2 ——— 09 Jul 2020

Set $N = n - F$ and use (65) to compute a rough score $h(l)$ for each free literal l. ⇝ Set $N \leftarrow n - F$. For each free literal l, set `VAL[`l`]` $\leftarrow 0$, and use (65) to compute a rough score $h(l)$.

▸**Page 44** line 6 of step X3 ——— 14 Jan 2020

whose rating exceeds ⇝ whose rating is less than

Page 44 line 3 of step X7 ——— 07 Nov 2020

`BASE` $+ 2S \ge$ `PT`. ⇝ `BASE` $+ 2S \ge$ `PT` (beware of overflow).

▸**Page 46** line 1 of step Y3 ——— 30 Jan 2020

Set $l \leftarrow$ `LL[`j`]` and $T \leftarrow$ `BASE+LO[`j`]`. ⇝ Set $l \leftarrow$ `LL[`$\hat{\jmath}$`]` and $T \leftarrow$ `BASE+LO[`$\hat{\jmath}$`]`.

▸**Page 46** in step Y7 ——— 11 Nov 2020

[Make $\hat{l}_0$ false.] ⇝ [Assume also l.]
with $l \leftarrow \hat{l}_0$ and $T \leftarrow$ `DT`. ⇝ with $T \leftarrow$ `DT`.

▸**Page 46** in step Y8 ——— 30 Jan 2020

set `CONFLICT` $\leftarrow$ X13 and exit ⇝ set `BASE` $\leftarrow$ `LBASE`, `CONFLICT` $\leftarrow$ X13, and exit

Page 51 the line after (82) ——— 12 Aug 2019

[See A. C. Kaporis ⇝ [See M. Hajiaghayi and G. B. Sorkin, arXiv:math/0310193 [math.CO] (2003), 8 pages; A. C. Kaporis

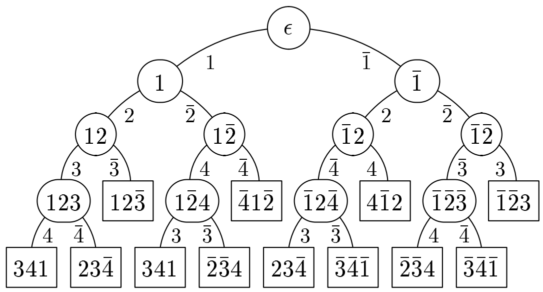
Page 51 line −9 ——— 05 Apr 2018

to appear ⇝ 59–68

▸**Page 52** line 6 after (84) ——— 11 Nov 2021

Section 7.4.1 ⇝ Section 7.4.4

▸**Page 55** replacement for Fig. 91 (formerly Fig. 48) ——— 10 Feb 2016



Page 57 line 10 ——— 22 Apr 2020

$\bar1\bar2\bar3, \bar12\bar4,$ $\leadsto$ $\bar12\bar4, \bar1\bar2\bar3,$

▸**Page 65** replacement for last line of Table 3 ——— 21 Jun 2022

(Here $**$ denotes the previously learned clause $\overline{50}\,\overline{26}\,27\,\overline{30}\,\overline{32}\,\overline{35}\,\overline{38}\,\overline{40}\,\overline{41}\,\overline{44}\,\overline{45}\,\overline{47}\,\overline{50}\,\overline{55}\,\overline{60}\,\overline{65}\,\overline{70}$.)

▸**Page 66** line -2 or -1 ——— 16 Jun 2019

(see exercise 261) $\leadsto$ (see exercise 265)

Page 66 new sentence at bottom of page ——— 15 Mar 2018

The "reason" for literal l's current value is kept in variable R_l.

Page 67 lines 10 and 22 ——— 06 May 2018

100 conflicts $\leadsto$ $M = 100$ conflicts
ρ^{-100} times $\leadsto$ $\rho^{-M} = \rho^{-100}$ times

▸**Page 68** line -2 ——— 21 Apr 2019

$L_F \leftarrow l',$ $\leadsto$ $L_F \leftarrow \bar l',\ \mathtt{VAL}(|l'|) \leftarrow 2d + (\bar l' \,\&\, 1),$

Page 74 in (124) ——— 16 Dec 2019

$m_j > 0$ and $\leadsto$

Page 80 line 14 ——— 17 May 2017

on variables that need $\leadsto$ on clauses whose variables need

Page 80 line 22 ——— 17 May 2017

tackled. $\leadsto$ tackled and when $N = \infty$.

▸**Page 86** line 2 of the proof of Theorem F ——— 21 Jun 2022

$\sum_{\alpha,\beta} \mu_G(\alpha) = \sum_\gamma \sum_{\alpha,\beta} \mu_G(\alpha)[\gamma = \alpha\beta]$ $\leadsto$
$\sum_{\alpha,\beta} \mu_G(\alpha)\alpha\beta = \sum_\gamma \sum_{\alpha,\beta} \mu_G(\alpha)\gamma[\gamma = \alpha\beta]$

▸**Page 86** bottom line ——— 02 Feb 2017

(this is, $\leadsto$ (that is,

Page 94 new paragraph at bottom of page ——— 28 Apr 2019

Even better results occur when step S8 is allowed to backtrack, resetting less-biased variables when problems arise. See R. Marino, G. Parisi, and F. Ricci-Tersenghi, *Nature Communications* **7**, 12996 (2016), 1–8.

Page 95 line −5 — 14 Mar 2021

so-called because ⇝ so called because

Page 96 bottom two lines — 03 Jul 2018

75] that introduced ... omitted ⇝ 75], which introduced important special cases that allow many of the pq potential clauses to be omitted

▸**Page 113** line 5 — 17 Jul 2018

$i_1 < j_1,\ i_2 < j_2,\ \ldots,\ i_t < j_t$ ⇝ $i_1 \le j_1,\ i_2 \le j_2,\ \ldots,\ i_t \le j_t$

Page 114 problem D5 — 17 Jul 2018

$\times'$ ⇝ $\times$

Page 118 line 27 — 17 May 2017

tests, ⇝ tests with $N = 50n$ and $p = .4$,

Page 121 lines −15 through −13 — 31 May 2017

the dancing links method of Section 7.2.2.1 ⇝ the backtrack method of exercise 7.2.2–21
7.2 Tμ ⇝ 4 Tμ
90 minutes ⇝ 50 minutes

Page 121 line −12 — 17 Jul 2018

trumps ⇝ trounces

Page 125 the line after (193) — 25 May 2019

these defaults ⇝ those defaults

▸**Page 126** line 16 — 17 Jul 2018

$(\alpha_i, \rho_i, \ldots, \psi_i)$ ⇝ $(\alpha, \rho, \ldots, \psi)$

Page 129 line 11 — 07 Feb 2021

conflict-driven ⇝ conflict driven

Page 130 line −18 — 07 Feb 2021

first order logic ⇝ first-order logic

▸**Page 131** line −14 — 09 Jul 2016

benchmark programs of 1992 ⇝ benchmark problems of 1992

▸**Page 132** line 23 — 23 Aug 2016

Independent Decreasing Sum ⇝ Independent Decaying Sum

Page 133 line 19 — 16 Jul 2020

In future, the next breakthrough ⇝ The next future breakthrough

Page 134 better wording in exercise 15 — 14 May 2016

that has an arbitrary order $n \ge 3$ ⇝ whose order is a given number $n \ge 3$

Page 135 new rating for exercise 24 — 15 Feb 2017

M32 ⇝ *M34*

▸**Page 135** line 3 of exercise 31 ——— 21 Apr 2022

$F_t(n) \leadsto F_t(r)$

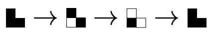
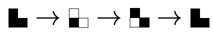
Page 136 end of exercise 38 ——— 10 May 2017

each side. $\leadsto$ each side?

Page 136 line 3 of exercise 43 ——— 21 Jan 2018

$(z_{m+n} \ldots z_1)_2 = (z_1 \ldots z_{m+n})_2 \leadsto (z_{2n} \ldots z_1)_2 = (z_1 \ldots z_{2n})_2$

▸**Page 138** last line of exercise 70 ——— 01 Jun 2020

 $\leadsto$ 

Page 145 new rating for exercise 150 ——— 15 Dec 2019

21 $\leadsto$ *24*

Page 147 last lines of exercise 176 ——— 26 May 2022

together with $b_{1,1}$ ... by (17). $\leadsto$ together with $(b_{1,1}) \wedge (c_{1,2}) \wedge (d_{1,3})$ to set the colors of (b_1, c_1, d_1) to $(1, 2, 3)$.

Page 150 replacement for exercise 204 ——— 01 May 2019

▸**204.** [*28*] Figure 89 might suggest that 3SAT problems on n variables are always easy when there are fewer than $2n$ clauses. We shall prove, however, that *any* set of m ternary clauses on n variables can be transformed mechanically into another set of ternary clauses on $N = O(m)$ variables in which no variable occurs more than four times. The transformed problem has the same number of solutions as the original one; thus it isn't any simpler, although (with at most $4N$ literals) it has at most $\frac{4}{3}N$ clauses.

a) First replace the original m clauses by m new clauses $(X_1 \vee X_2 \vee X_3)$, ..., $(X_{3m-2} \vee X_{3m-1} \vee X_{3m})$, on $3m$ new variables, and show how to add $3m$ clauses of size 2 so that the resulting $4m$ clauses have exactly as many solutions as the original.

b) Construct ternary clauses that have a unique solution, yet no variable occurs more than four times.

c) Use (a) and (b) to prove the N-variable result claimed above.

Page 150 replacement for exercise 205 ——— 01 May 2019

205. [*26*] If F and F' are sets of clauses, let $F \sqcup F'$ stand for any other set obtained from $F \cup F'$ by replacing one or more clauses C of F by $x \vee C$ and one or more clauses C' of F' by $\bar{x} \vee C'$, where x is a new variable. Then $F \sqcup F'$ is unsatisfiable whenever F and F' are both unsatisfiable. For example, if $F = \{\epsilon\}$ and $F' = \{1, \bar{1}\}$, then $F \sqcup F'$ is either $\{2, 1\bar{2}, \bar{1}\bar{2}\}$ or $\{2, 1, \bar{1}\bar{2}\}$ or $\{2, 1\bar{2}, \bar{1}\}$.

a) Construct 16 unsatisfiable ternary clauses on 15 variables, where each variable occurs at most four times.

b) Construct an unsatisfiable 4SAT problem in which every variable occurs at most five times.

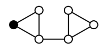
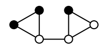
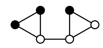
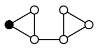
Page 151 new rating for exercise 215 ——— 09 Jul 2018

HM23 $\leadsto$ *HM28*

▸**Page 154** the displayed equation on line 13 ——— 17 Jul 2018

$\ell(v) \leadsto l(v)$

▸**Page 154** in exercise 245 ———— 30 Aug 2020

line 2 in part (a): G' =  and G'' =  $\rightsquigarrow$ G' =  and G'' = 

line 1 in part (c): if $|V'| = \mu(C)$ $\rightsquigarrow$ if $\bigcup_{v \in V'} \alpha(v) \vdash C$ and $|V'| = \mu(C)$
last line of part (d): $s/n + 8$ $\rightsquigarrow$ $s/m + 8$

Page 154 line 3 of exercise 246 ———— 03 Apr 2022

extended $\rightsquigarrow$ *extended*

Page 159 new rating for exercise 305 ———— 17 May 2017

M25 $\rightsquigarrow$ *HM29*

▸**Page 165** line 7 of exercise 356 ———— 17 Aug 2017

Theorems M and K $\rightsquigarrow$ Theorems M and S

▸**Page 166** line −5 of exercise 363 ———— 21 Mar 2020

The sets in (d) $\rightsquigarrow$ The sets in (f)

Page 167 new rating for exercise 372 ———— 21 Jul 2022

25 $\rightsquigarrow$ *30*

Page 167 last line of exercise 374 ———— 13 Jul 2022

literal $l \in C$ $\rightsquigarrow$ literal $\bar{x} \in C$

Page 167 new rating for exercise 376 ———— 21 Jul 2022

32 $\rightsquigarrow$ *36*

▸**Page 167** line 3 of exercise 379 ———— 25 Jan 2022

$(b \vee c \vee d)$ $\rightsquigarrow$ $(a \vee b \vee c)$

Page 168 in exercise 385(b) ———— 21 Jun 2022

in (7) $\rightsquigarrow$ in (7)

▸**Page 169** in exercise 386 ———— 09 Jun 2022

line 1: [*M25*] $\rightsquigarrow$ [*M31*]
line 6: returns to C2. The data structures `HEAP`, `OVAL`, and `ACT` $\rightsquigarrow$ returns to C5. The data structures `HEAP`, `HLOC`, `OVAL`, `ACT`
line 17: on a helpful round $\rightsquigarrow$ on a round that's helpful for C and l
bottom line: $\mu \ln n \ln \ln n$ $\rightsquigarrow$ $\mu n \ln n$

Page 170 line 1 of exercise 392(c) ———— 21 Jun 2022

SAT solver $\rightsquigarrow$ SAT solver

Page 172 lines 2 and 3 of exercise 404 ———— 21 Jun 2022

using either ... variable. $\rightsquigarrow$ using $d + 1 - a$ clauses of length ≤ 4 and no auxiliary variables.

▸**Page 173** line 2 of exercise 417 ———— 23 Feb 2016

via (174) $\rightsquigarrow$ via (173)

Page 174 last line of exercise 428 ——— 26 Mar 2016

M and N cannot be $\rightsquigarrow$ M and N cannot both be

Page 174 exercise 433 ——— 04 Jun 2022

[this exercise now is 'Recommended'; that is, it's now preceded by ▸]

Page 178 first line of exercise 474 ——— 18 Mar 2020

A signed permutation is a symmetry $\rightsquigarrow$ A signed permutation σ is a *symmetry*

Page 179 line 2 of exercise 481 ——— 01 Aug 2016

$(x\oplus y)y \rightsquigarrow (x\oplus y)\,y$

Page 179 replacement for final paragraph of exercise 483 ——— 25 May 2019

Experiment with this scheme by applying it to the Mycielski graphs of exercise 7.2.2.1–116.

▸**Page 183** row 16, column 39 of the big matrix in exercise 518 ——— 11 Jan 2016

$_0 \rightsquigarrow {}_2$

Page 183 line 2 of exercise 518(a) ——— 11 Jan 2016

$r \rightsquigarrow p \quad s \rightsquigarrow q \quad t \rightsquigarrow r$

▸**Page 184** line 2 of exercise 520 ——— 23 Dec 2019

feasability $\rightsquigarrow$ *feasibility*

▸**Page 184** line 2 of exercise 521 ——— 21 Jun 2022

Forget a learned clause $\rightsquigarrow$ Keep a learned clause

Page 185 lines 4 and 5 of answer 5 ——— 22 Aug 2019

Bloom in ..., shows $\rightsquigarrow$
Bloom in *J. London Math. Society* (2) **93** (2016), 643–663, shows

▸**Page 185** line 6 of answer 6 ——— 02 Jun 2017

By Theorem J, $\rightsquigarrow$ By Theorem L,

Page 186 replacement for lines 3–7 of answer 11 ——— 04 Feb 2016

Furthermore, the only way to match $j'1$ is to choose one of the satisfiability triples for clause j. Suppose $\bar{l}_k j$ belongs to the chosen triple; then we must also have chosen the true triples for literal l_k. Thus a perfect matching implies satisfiable clauses.

Page 186 in answer 12 ——— 31 May 2017

columns $\rightsquigarrow$ items [three changes]

Page 186 in answer 13 ——— 31 May 2017

rows $\rightsquigarrow$ options [two changes]

▸**Page 186** last line of answer 13 ——— 21 Jun 2022

3600 cells $\rightsquigarrow$ 3600 literals

Page 186 line 1 of answer 16 ——— 05 Apr 2021

when $n \geq 5 \rightsquigarrow$ for all $n \geq 3$

▸**Page 186** or 187 at end of answer 16 ——— 30 May 2019

Laurier, *Artificial Intelligence* **14** ⤳ Lauriere, *Artificial Intelligence* **10**

▸**Page 187** line 1 of answer 17 ——— 21 Jun 2022

Add clauses (19) ⤳ Add clauses (18) and (19)

Page 189 simplifying the formula for f in answer 24 ——— 15 Feb 2017

$r-l<n''?\ \ldots f(n'',0,n''))$ for $n\geq 2$ ⤳ $f(n',(n'+l)\dot{-}r,n')+f(n'',(n''+l)\dot{-}r,n''))$ for $n\geq 2$

▸**Page 190** line 2 of answer 33 ——— 28 Jul 2019

$v_{j-1}\vee v_{j+1}\vee v_k$ ⤳ $v_{j-1}\vee v_{j+1}\vee\cdots\vee v_k$

Page 191 line 2 of answer 35 ——— 23 Dec 2015

W_6 ⤳ W_5
C_{n-1} ⤳ C_n

▸**Page 192** in answer 43 ——— 21 Jan 2018

seem to be quite rare ... when $n=32$. ⤳ aren't at all rare. For example, $(x,y)=$ ($^{\#}$`C4466223`, $^{\#}$`E26E7647`) is one of 293 instances when $n=32$.

Page 192 in answer 46 ——— 18 Dec 2019

Indeed, there's a nice ... 2009.] ⤳ Indeed, M. Coriand has discovered a nice way to find all n-bit examples by considering only $O(2^{n/4})$ cases, because the left and right halves of a binary number are nearly forced by the left and right *quarters* of its square. The first eight square binary palindromes were found by G. J. Simmons, *JRM* **5** (1972), 11–19; see OEIS sequence A003166 for many further results.]

▸**Page 194** line 6 of answer 56 ——— 21 Jun 2022

on the left be u_k and v_l as ⤳ on the left be u_k and those on the right be v_l, as

Page 196 lines 17–18 ——— 15 Dec 2016

`arXiv:1410.2736` [cs.DS] (2014), 10 pages ⤳ *LNCS* **9136** (2015), 167–176

Page 199 lines −11 and −7 ——— 08 Apr 2017

tarnishing ⤳ tainting

Page 200 line 2 ——— 08 Apr 2017

size less than 16×18. ⤳ size 15×18 or 16×17.

Page 202 line −3 of answer 85 ——— 03 Mar 2021

(August 2013), #P16 ⤳ (2013), #P16, 1–19

▸**Page 204** bottom line and top lines of page 205 ——— 11 Jan 2016

nonzero top row ... $(n-k)$, which ⤳ nonzero first row and column has all entries zero except that $y_{1t}=-y_{t1}=y_{(k+1-t)1}=-y_{kt}=y_{k(k+1-t)}=-y_{(k+1-t)k}=y_{tk}=-y_{1(k+1-t)}$, for some t and k with $1<t\leq k/2\leq m/2$. So the answer is $2\sum_{k=4}^{m}\lfloor k/2-1\rfloor(m+1-k)(n+1-k)$, which

▸**Page 205** line 4 of answer 105(e) ——— 21 Jun 2022

for $n=(5,7,9)$ ⤳ for $m=n=(5,7,9)$

▸**Page 209** replacement for answer 128 ——— 10 Feb 2016

128. Active ring	$x_1x_2x_3x_4$	Units	Choice	Changed clauses
(1 2 3 4)	`- - - -`		$\bar{1}$	$21\bar{3}$
(2 3 4)	`0 - - -`		$\bar{2}$	$\bar{3}12, 32\bar{4}$
(3 4)	`0 0 - -`	$\bar{3}$	$\bar{3}$	$\bar{4}23, 431$
(4)	`0 0 0 -`	$4, \bar{4}$	Backtrack	
(3 4)	`0 - - -`		2	$\bar{3}\bar{2}4$
(3 4)	`0 1 - -`	$\bar{4}$	$\bar{4}$	$341, \bar{1}42$
(3)	`0 1 - 0`	$3, \bar{3}$	Backtrack	
(4 3)	`- - - -`		1	$4\bar{1}2, \bar{2}\bar{1}3$
(2 4 3)	`1 - - -`		$\bar{2}$	
(4 3)	`1 0 - -`	4	4	$32\bar{4}, 1\bar{4}\bar{2}$
(3)	`1 0 - 1`	$3, \bar{3}$	Backtrack	
(4 3)	`1 - - -`		2	$3\bar{1}\bar{2}$
(4 3)	`1 1 - -`	3	3	$4\bar{2}\bar{3}, 1\bar{3}2, \bar{4}\bar{3}\bar{1}$
(4)	`1 1 1 -`	$4, \bar{4}$	Backtrack	

▸**Page 211** line 7 of answer 137 ——— 04 Mar 2018

`LINK(`t`)` $\leftarrow p''$. $\rightsquigarrow$ `LINK(`t`)` $\leftarrow p''$, $p' \leftarrow t$.

Page 211 last line of answer 137 ——— 16 Aug 2020

`VAR` or `INX`.) $\rightsquigarrow$ `VAR` or `INX`. These efficient techniques based on swapping are examples of "sparse-set representations"; see 7.2.2–(16) thru (23).)

Page 212 append new line to end of answer 143 ——— 10 May 2022

ParamILS advises changing α from 3.5 to 0.001(!) in (195).

Page 212 line 3 of answer 145 ——— 27 Apr 2019

$= \rightsquigarrow \approx$

Page 212 recomputed values for the bottom three lines ——— 13 Aug 2018

(2.10, . . . , 24.17). $\rightsquigarrow$ (2.10, 9.10, 3.10, 6.60, 3.10, 13.60, 4.10, 11.10), (5.63, 3.37, 9.24, 2.57, 5.48, 5.67, 8.37, 4.87), (1.42, 10.00, 2.31, 10.42, 1.28, 17.69, 1.94, 16.07), (8.12, 1.43, 12.42, 1.30, 7.51, 2.41, 12.02, 1.81), (0.32, 14.72, 0.42, 16.06, 0.30, 26.64, 0.43, 24.84).

Page 213 recomputed values for lines 2 and 3 ——— 27 Apr 2019

$$(0.1017, 20.6819, 0.1027, 21.6597, 0.1021, 32.0422, 0.1030, 33.0200) \quad \text{and} \quad (8.0187, 0.1712, 11.9781, 0.1361, 11.9781, 0.2071, 15.9374, 0.1718).$$

▸**Page 213** in answer 150 ——— 01 Oct 2019

line 1: 147 $\rightsquigarrow$ $\bar{1}\bar{4}\bar{7}$
line 3: 369 $\rightsquigarrow$ $\bar{3}\bar{6}\bar{9}$

▸**Page 214** line 4 of answer 152 ——— 14 Jan 2020

b and c both true $\rightsquigarrow$ x_2 and x_3 both true

Page 214 in answer 153 ——— 11 Nov 2020

at the top (see Section 5.2.3). $\rightsquigarrow$ at the top. (See Section 5.2.3; but start indices at 0, with $r($`CAND[`k`]`$) \le \min(r($`CAND[`$2k{+}1$`]`$), r($`CAND[`$2k{+}2$`]`$))$.)

▸**Page 214** line 6 of answer 155 ——— 11 Nov 2021

Section 7.4.1 $\rightsquigarrow$ Section 7.4.1.2

Page 221 replacement for last line of answer 195 ——— 12 Aug 2019

see A. Coja-Oghlan and K. Panagiotou, *Advances in Math.* **288** (2016), 985–1068.]

Page 223 replacement for answer 204(b,c,d) ——— 26 Apr 2019

(b) (Solution by E. Wynn.) The following 44 clauses in 35 variables are satisfiable if and only if each variable is false: $a_i \vee \bar{b}_i \vee c_i$, $\bar{a}_i \vee \bar{b}_i \vee c_i$, $\bar{b}_i \vee \bar{c}_i \vee \bar{d}_i$, $\bar{c}_i \vee d_i \vee e_i$, $d_i \vee \bar{e}_i \vee f_i$, $\bar{e}_i \vee \bar{f}_i \vee \bar{g}_i$, $\bar{f}_i \vee g_i \vee h_i$, $\bar{f}_i \vee g_i \vee \bar{h}_i$, for $i \in \{1,2\}$; $b_1 \vee b_2 \vee \bar{A}$, $A \vee B \vee \bar{C}$, $A \vee \bar{B} \vee D$, $A \vee \bar{D} \vee E$, $\bar{B} \vee \bar{D} \vee \bar{E}$, $B \vee C \vee \bar{F}_1$, $C \vee \bar{E} \vee G_1$, $C \vee F_1 \vee \bar{G}_1$; $F_j \vee G_j \vee \bar{F}_{j+1}$, $F_j \vee G_j \vee \bar{G}_{j+1}$, for $1 \le j \le 6$; $D \vee E \vee \bar{a}_1$, $F_2 \vee G_2 \vee \bar{a}_2$, $F_3 \vee G_3 \vee \bar{h}_1$, $F_4 \vee G_4 \vee \bar{h}_2$, $a_1 \vee h_1 \vee \bar{d}_1$, $a_2 \vee h_2 \vee \bar{d}_2$, $F_5 \vee G_5 \vee \bar{g}_1$, $F_6 \vee G_6 \vee \bar{g}_2$.

(c) Add the clauses of (b), and the clauses $F_j \vee G_j \vee \bar{F}_{j+1}$, $F_j \vee G_j \vee \bar{G}_{j+1}$ for $7 \le j \le \lceil 3m/2 \rceil + 3$, to the $4m$ clauses of (a). We can stick the literals $\{F_7, G_7, \dots\}$, which are always false, into the 2-clauses without using any variable five times, obtaining at most ($7m$+39) 3-clauses in $N \approx 7m + 30$ variables.

Page 223 and 224, replacement for answer 205 ——— 01 May 2019

205. Let $F_0 = \{\epsilon\}, \ldots \rightsquigarrow$

205. (a) After $F_0 = \{\epsilon\}$, $F_1 = F_0 \sqcup F_0$, $F_2 = F_0 \sqcup F_1$, $F_3 = F_0 \sqcup F_2$, $F_4 = F_3 \sqcup F_3'$, $F_5 = F_4 \sqcup F_4''$, always putting the new variable into the four shortest possible clauses, we get $F_5 = \{345,\ 2\bar{3}4,\ 1\bar{2}\bar{3},\ \bar{1}\bar{2}\bar{3},\ 3'\bar{4}5,\ 2'\bar{3}'\bar{4},\ 1'\bar{2}'\bar{3}',\ \bar{1}'\bar{2}'\bar{3}',\ 3''4''\bar{5},\ 2''\bar{3}''4'',\ 1''\bar{2}''\bar{3}'',\ \bar{1}''\bar{2}''\bar{3}'',\ 3'''\bar{4}''\bar{5},\ 2'''\bar{3}'''\bar{4}'',\ 1'''\bar{2}'''\bar{3}''',\ \bar{1}'''\bar{2}'''\bar{3}'''\}$.

(b) Let $F_0 = \{\epsilon\}, \ldots$

Page 224 line −2 ——— 08 May 2017

SODA **22** (2011), 664–674 $\rightsquigarrow$ *JACM* **63** (2016), 43:1–43:32

Page 225 in answer 212 ——— 31 May 2017

line 2: of the rows $\rightsquigarrow$ with options
line 3: matching; see the discussion of sudoku in Section 7.2.2.1. $\rightsquigarrow$ matching.

▸**Page 226** replacement for lines 8 and 9 of answer 215 ——— 09 Jul 2018

$$\frac{P_{L+k}}{P_L} = \exp\Bigl(-\frac{\gamma k^2}{n} + O\Bigl(\frac{k}{n}\Bigr) + O\Bigl(\frac{k^3}{n^2}\Bigr)\Bigr), \quad \gamma = \frac{(\ln 2)^2}{6\alpha}\Bigl(1 + \frac{2}{t_\alpha^3}\Bigr), \quad \text{when } L = 2t_\alpha n;$$

by trading tails, the expected total tree size is $\sqrt{\pi n/\gamma}\,P_L\bigl(1 + O(1/\sqrt{n})\bigr)$.

▸**Page 228** last line of answer 231 ——— 19 Apr 2020

$G_{mm} \diamond I_{mm} = \epsilon$. $\rightsquigarrow$ $G_{mm} \diamond I_m = \epsilon$.

▸**Page 229** replacement for first two lines of answer 233 ——— 19 Apr 2020

233. $C_9 = C_6 \diamond C_8$, $C_{10} = C_1 \diamond C_9$, $C_{11} = C_3 \diamond C_{10}$, $C_{12} = C_7 \diamond C_{10}$, $C_{13} = C_4 \diamond C_{11}$, $C_{14} = C_2 \diamond C_{12}$, $C_{15} = C_{13} \diamond C_{14}$, $C_{16} = C_5 \diamond C_{15}$, $C_{17} = C_6 \diamond C_{15}$, $C_{18} = C_8 \diamond C_{15}$,

▸**Page 229** line 2 of answer 235 ——— 21 Jun 2022

k stages $\rightsquigarrow$ m stages

▶**Page 229** in answer 237 ——— 13 Aug 2020

line 2: m pigeons to $m-1$ $\rightsquigarrow$ $m+1$ pigeons to m
line 2: $6(m-1)(m-2)$ new clauses $\rightsquigarrow$ $6m(m-1)$ new clauses
line 4: $2(m-1)(m-2)$ new variables $\rightsquigarrow$ $2m(m-1)$ new variables
line -3: reduce m to $m-1$ $\rightsquigarrow$ reduce $m+1$ to m

▶**Page 231** in answer 245 ——— 30 Aug 2020

lines 1–2: When conditioning on e_{uv}, simply ... also complement $\rightsquigarrow$ When conditioning on $\bar{e}_{uv}$, simply delete the edge u — v from G. When conditioning on e_{uv}, also complement
line 6: $\alpha(G) \mid \{\bar{b}, e\}$ $\rightsquigarrow$ $\alpha(G) \mid \{b, \bar{e}\}$ and $\alpha(G) \mid \{\bar{b}, \bar{e}\}$ $\rightsquigarrow$ $\alpha(G) \mid \{b, e\}$

Page 231 lines -4 and -3 of answer 245 ——— 21 Apr 2020

are part of the Stanford GraphBase $\rightsquigarrow$ can be found by the algorithms of the Stanford GraphBase

▶**Page 231** lines 9 and 10 of answer 246 ——— 21 Apr 2020

$[s_v]^{\ell(v)\oplus 1}$, meaning $s_v = \ell(v)$. $\rightsquigarrow$ $[s_v]^{l(v)\oplus 1}$, meaning $s_v = l(v)$.

▶**Page 231** line 14 of answer 246 ——— 21 Apr 2020

$0 \le v \le d$ $\rightsquigarrow$ $0 \le k \le d$

▶**Page 232** line 3 ——— 21 Apr 2020

$s_v = \ell(v)$ $\rightsquigarrow$ $s_v = l(v)$

▶**Page 233** line 10 of answer 257 ——— 15 Mar 2018

where $\mathit{red}(l)$ is $\rightsquigarrow$ where $\mathit{red}(\bar{l})$ is

▶**Page 234** line 8 of answer 263 ——— 21 Apr 2019

$\mathtt{TLOC}(|l_1|)$ $\rightsquigarrow$ $\mathtt{TLOC}(|l_0|)$

▶**Page 234** line 14 of answer 263 ——— 21 Apr 2019

$\mathit{lev}(|b_j|) < d'$ $\rightsquigarrow$ $\mathit{lev}(b_j) < d'$

▶**Page 235** second line of (**) ——— 23 Dec 2021

$l_0 \leftarrow R_H$ $\rightsquigarrow$ $l_0 \leftarrow L_H$

▶**Page 236** line -2 of answer 271 ——— 15 Mar 2018

$\mathtt{VAL}(|l_j|) \le 2d' + 1$ $\rightsquigarrow$ $0 \le \mathtt{VAL}(|l_j|) \le 2d' + 1$

▶**Page 239** line 8 of answer 288 ——— 15 Mar 2018

If $v < 2$ $\rightsquigarrow$ Set $v \leftarrow \mathtt{VAL}(|l_k|)$. If $v < 2$

▶**Page 240** line 2 of answer 293 ——— 05 Dec 2019

do this: If $\rightsquigarrow$ do this: Set $M_{\rm f} \leftarrow M_{\rm f} + v_{\rm f}$. If

Page 241 line 2 ——— 14 Sep 2016

$\Pr(T \ge N) \le n/N$ $\rightsquigarrow$ $\Pr(T \ge N \mid \text{the algorithm terminates}) \le n/N$

▶**Page 241** line 5 of answer 299, in the denominator ——— 14 Sep 2016

$\lceil x^{3/2} \rceil$ $\rightsquigarrow$ $\lceil x \rceil^{3/2}$

Page 242 in answer 302 ——— 22 Jun 2020

just as we need to know the literals ... independent of the others.) $\leadsto$
just as we need to know the literals of any given clause.

W4. [Choose l.] Set $g \leftarrow [U \geq p]$, $c \leftarrow \infty$, $j \leftarrow z \leftarrow 0$, and do the following while $j < k$: Set $j \leftarrow j+1$. Then if $c_{|l_j|} < c$ and either $c_{|l_j|} = 0$ or $g = 1$, set $c \leftarrow c_{|l_j|}$ and $z \leftarrow 0$. Then if $c_{|l_j|} \leq c$, set $z \leftarrow z+1$, and if $zU < 1$ also set $l \leftarrow l_j$. (Here each random fraction U should be independent of the others.)

Page 243 line 10 of answer 304 ——— 17 May 2017

$a \leq n-2$, $\leadsto$ $a \leq n-2$, $a \neq n/2$,

▸**Page 243** lines 23–30 of answer 304 ——— 17 May 2017

Functions t_m and u_m ... when n is odd. $\leadsto$
Functions t_k and u_k can be defined as in MPR–105; but the new recurrences are $t_{k+1} = (t_k - pz^2 t_{k-1})/q$ and $u_{k+1} = (u_k - pz^2 u_{k-1})/q$. Hence

$$T(w) = \frac{q - pw}{q - w + pz^2w^2}; \qquad U(w) = \frac{q - (1-qz)w}{q - w + pz^2w^2}.$$

Differentiating with respect to z, then setting $z = 1$, now yields

$$t_k'(1) = \frac{2pq(1-(p/q)^k)}{(q-p)^2} - \frac{2pk}{q-p}, \qquad u_k'(1) = \frac{(2p-(p/q)^k)q}{(q-p)^2} - \frac{2p(k-1/2)}{q-p}.$$

It follows that $g_a'(1) = a/(q-p) - 2pq((p/q)^{m-a} - (p/q)^m)/(q-p)^2$ for $0 \leq a \leq n/2$ when n is even, $a/(q-p) - q((p/q)^{m-a} - (p/q)^m)/(q-p)^2$ when n is odd.

Page 249 last line of answer 333 ——— 29 May 2020

[*Manuscripta* ... 241.] $\leadsto$ [*Discrete Mathematics* **14** (1976), 215–239; *Manuscripta Mathematica* **19** (1976), 211–243. See also R. Fröberg, *Mathematica Scandinavica* **37** (1975), 29–39.]

▸**Page 253** line -2 ——— 21 Mar 2020

is 2 $\leadsto$ is $2n$

Page 257 lines 3 and 4 of answer 374 ——— 31 May 2017

additional links as in the exact cover algorithms of Section 7.2.2.1: $\leadsto$ additional links:

▸**Page 258** line 1 of answer 374(b) ——— 15 Jul 2020

a literal l with $\leadsto$ a literal l in C with

Page 258 replacement for answer 374(c) ——— 13 Jul 2022

(c) Use (b), but set $p \leftarrow \mathtt{F}(l = \bar{x}?\ x\colon l)$, and use $\big((\mathtt{SIG}(c)\ \&\ {\sim}\mathtt{SIG}(\bar{x}))\mid \mathtt{SIG}(x)\big)$ in place of $\mathtt{SIG}(c)$. Also modify the detailed test, by inserting 'if $u = \bar{x}$ then $u \leftarrow x$' just after each occurrence of '$u \leftarrow \mathtt{L}(q)$'.

▸**Page 259** in answer 376 ——— 21 Jul 2022

line 5: nonzero; $\leadsto$ `FF`, `FT`, or `EQ`;
line 10: $\mathtt{TALLY}(l) = 0$ $\leadsto$ $\mathtt{TALLY}(\bar{l}) = 0$
line 26: subsumption stack $\leadsto$ exploitation stack
lines 33–34: set nonzero when ... output. $\leadsto$ set nonzero when x is eliminated or its elimination has been abandoned.

Page 261 line 1 of answer 385 ________ 18 Mar 2022

$\overline{C} \wedge l \vdash_1 \epsilon \rightsquigarrow \overline{C} \wedge l \vdash_1 \epsilon$ when $l \in C$

▸**Page 261** in answer 386 ________ 18 Mar 2022

line 2: level d. $\rightsquigarrow$ level d during the helpful round.
line 7: $(\bar{c}, \bar{d}, \bar{l}) \rightsquigarrow (\bar{c}, \bar{l})$
line 8: $\frac{1}{10}\,\frac{1}{6}\,\frac{1}{4},\ \frac{1}{10}\,\frac{1}{6}\,\frac{1}{4} \rightsquigarrow \frac{1}{10}\,\frac{1}{6}\,\frac{1}{4},\ \frac{1}{10}\,\frac{1}{4}$

▸**Page 262** in answer 386 ________ 18 Mar 2022

line 2: a suitable decision $\rightsquigarrow$ suitable decisions
line 5: is a geometric $\rightsquigarrow$ is upper-bounded by a geometric

▸**Page 262** in answer 389 ________ 02 Mar 2020

line 1: This is a $\rightsquigarrow$ This is similar to a
line 2: `H` — `E`, $\rightsquigarrow$ `H` — `E`, `E` — ␣,
line 3: `N` — `G`. The $\rightsquigarrow$ `N` — `G`; however, we allow $v' = w'$ when $v \neq w$, and labels must match. The

Page 262 second paragraph of answer 389 ________ 19 Jan 2019

needs less than $\rightsquigarrow$ needs fewer than

▸**Page 262** and 263, new answer 390 (but last paragraph remains as before) 09 Feb 2016

390. Let $d(u,v)$ be the distance between vertices u and v. Then $d(v,v) = 0$ and

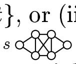
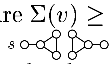
$$d(u,v) \le j+1 \iff d(u,w) \le j \text{ for some } w \in N(v) = \{w \mid w \longrightarrow v\}$$

if $u \neq v$. In parts (a), (d), we introduce variables v_j for each vertex v and $0 \le j \le k$. In part (c) we do this for $0 \le j < n$. But parts (b), (e), (f) use just n variables, $\{v \mid v \in V\}$.

(a) Clauses $(s_0) \wedge \bigwedge_{v \in V \setminus s}\bigl((\bar{v}_0) \wedge \bigwedge_{j=0}^{k-1}(\bar{v}_{j+1} \vee \bigvee_{w \in N(v)} w_j)\bigr)$ are satisfied only if $v_j \le [d(s,v) \le j]$; hence the additional clause (t_k) is also satisfied only if $d(s,t) \le k$. Conversely, if $d(s,t) \le k$, all clauses are satisfied by setting $v_j \leftarrow [d(s,v) \le j]$.

(b) There's a path from s to t if and only if there's a subset $H \subseteq V$ such that $s \in H$, $t \in H$, and every vertex v of the induced graph $G \mid H$ has degree $2 - [v = s] - [v = t]$. [The vertices on a shortest path from s to t yield one such H. Conversely, given H, we can find vertices $v_j \in H$ such that $s = v_0 \longrightarrow v_1 \longrightarrow \cdots \longrightarrow v_k = t$.]

We can represent that criterion via clauses on the binary variables $v = [v \in H]$ by asserting $(s) \wedge (t)$, together with clauses to ensure that $\Sigma(v) = 2 - [v = s] - [v = t]$ for all $v \in H$, where $\Sigma(v) = \sum_{w \in N(v)} w$ is the degree of v in $G \mid H$. The number of such clauses for each v is at most $6|N(v)|$, because we can append $\bar{v}$ to each clause of (18) and (19) with $r = 2$, and $|N(v)|$ additional clauses will rule out $\Sigma(v) < 2$. Altogether there are $O(m)$ clauses, because $\sum_{v \in V} |N(v)| = 2m$.

[Similar but simpler alternatives, such as (i) to require $\Sigma(v) \in \{0, 2\}$ for all $v \in V \setminus \{s,t\}$, or (ii) to require $\Sigma(v) \ge 2$ for all $v \in H \setminus \{s,t\}$, do *not* work: Counterexamples are (i) s  t and (ii) s  t. Another solution, more cumbersome, associates a Boolean variable with each *edge* of G.]

(c) Let s be any vertex; use (a) with $k = n-1$, plus (v_{n-1}) for all $v \in V \setminus s$.

(d) Clauses $(s_0) \wedge \bigwedge_{j=0}^{k-1} \bigwedge_{v \in V} \bigwedge_{w \in N(v)} (\bar{v}_j \vee w_{j+1})$ are satisfied only if we have $v_j \ge [d(s,v) \le j]$; hence the additional unit clause $(\bar{t}_k)$ cannot also be satisfied when $d(s,t) \le k$. Conversely, if $d(s,t) > k$ we can set $v_j \leftarrow [d(s,v) \le j]$.

(e) $(s) \wedge \bigl(\bigwedge_{v \in V} \bigwedge_{w \in N(v)} (\bar{v} \vee w)\bigr) \wedge (\bar{t})$.

(f) Letting s be any vertex, use $(s) \wedge \bigl(\bigwedge_{v \in V} \bigwedge_{w \in N(v)} (\bar{v} \vee w)\bigr) \wedge \bigl(\bigvee_{v \in V \setminus s} \bar{v}\bigr)$.

▸**Page 263** line −3 of answer 391 — 29 May 2022

$= k \rightsquigarrow = k - 2^{l-1}$.

▸**Page 264** line −3 of answer 396 — 31 Jan 2020

$(\bar{u}^3 \vee \bar{v}^3) \wedge (u^1 \vee v^1) \vee (\overline{\hat{u}^3} \vee \overline{\hat{v}^3}) \wedge (\hat{u}^1 \vee \hat{v}^1) \rightsquigarrow (\bar{u}^3 \vee \bar{v}^3) \wedge (u^1 \vee v^1) \wedge (\overline{\hat{u}^3} \vee \overline{\hat{v}^3}) \wedge (\hat{u}^1 \vee \hat{v}^1)$

Page 265 line −3 of answer 397 — 08 May 2022

but with $\rightsquigarrow$ but combined with

▸**Page 265** line −3 of answer 399 — 26 Apr 2019

521–523 $\rightsquigarrow$ 521–522

Page 266 new answer for exercise 404 — 21 Jun 2022

404. $\bigwedge_{j=0}^{d-a}(\bar{x}^j \vee x^{j+a} \vee \bar{y}^j \vee y^{j+a})$. (As usual, omit literals with superscript 0 or d.)

▸**Page 267** lines −5 and −4 — 17 Jul 2018

$\delta \geq w_{ij}$ and $\delta \geq w_{i'j'} \rightsquigarrow \delta \leq w_{ij}$ and $\delta \leq w_{i'j'}$

Page 269 append to answer 415 — 15 Nov 2018

[The clauses (169) are due to K. Sakallah, *Handbook of Satisfiability* (2009), Chapter 10, (10.32).]

▸**Page 269** line 2 of answer 417 — 23 Feb 2016

as in (174) $\rightsquigarrow$ as in (173)

▸**Page 269** line 3 of answer 418 — 23 Feb 2016

via (174) $\rightsquigarrow$ via (173)

Page 269 line 1 of answer 423 — 04 Apr 2016

[But a *forcing* $\rightsquigarrow$ [But Abío, Gange, Mayer-Eichberger, and Stuckey have shown [*LNCS* **9676** (2016), 1–17] that weak forcing is always achieved if $(\bar{a}_j \vee a_l \vee a_h)$ is added to (173). Furthermore, a *forcing*

Page 270 line 7 of answer 426 — 05 Apr 2016

variable elimination $\rightsquigarrow$ the elimination of an auxiliary variable

▸**Page 270** line 6 of answer 428 — 30 Mar 2020

for $1 \leq t < N \rightsquigarrow$ for $0 \leq t < N$

Page 271 line 6 of answer 436 — 04 Apr 2016

$\bigvee\{q_{k-1} \mid (q,a,q') \in T\}$); together with (vi) $\rightsquigarrow \bigvee\{q_{k-1} \mid (q,a,q') \in T\}$), for $a \in A$, $q' \in Q$. And (vi)

▸**Page 271** replacement for lines 18–23 of answer 436 — 04 Apr 2016

For example, the language L_2 of exercise 434 yields $20n+4$ clauses with $8n+3$ auxiliary variables: $F = \bigwedge_{k=1}^{n}((\bar{t}_{k00} \vee \bar{x}_k) \wedge (\bar{t}_{k00} \vee 0_k) \wedge (\bar{t}_{k11} \vee x_k) \wedge (\bar{t}_{k11} \vee 1_k) \wedge (\bar{t}_{k12} \vee x_k) \wedge (\bar{t}_{k12} \vee 2_k) \wedge (\bar{t}_{k02} \vee \bar{x}_k) \wedge (\bar{t}_{k02} \vee 2_k) \wedge (\bar{0}_{k-1} \vee t_{k00} \vee t_{k11}) \wedge (\bar{1}_{k-1} \vee t_{k12}) \wedge (\bar{2}_{k-1} \vee t_{k02}) \wedge (\bar{0}_k \vee t_{k00}) \wedge (\bar{1}_k \vee t_{k11}) \wedge (\bar{2}_k \vee t_{k02} \vee t_{k12}) \wedge (x_k \vee t_{k00} \vee t_{k02}) \wedge (\bar{x}_k \vee t_{k11} \vee t_{k12}) \wedge (\bar{t}_{k00} \vee 0_{k-1}) \wedge (\bar{t}_{k11} \vee 0_{k-1}) \wedge (\bar{t}_{k12} \vee 1_{k-1}) \wedge (\bar{t}_{k02} \vee 2_{k-1})) \wedge (\bar{1}_0) \wedge (\bar{2}_0) \wedge (\bar{0}_n) \wedge (\bar{1}_n)$.

▸**Page 272** lines 6 and 10 of answer 440 — 03 Apr 2016

$QR_{hik} \rightsquigarrow QP_{hik}$ (twice)

Page 274 end of answer 448 ——— 17 Aug 2019

Horsley has shown ... preparation). $\rightsquigarrow$ Horsley proved in 2015 that $Z(m,n) = 3n + \lfloor 1 - m/14 \rfloor$ in such cases.

▶**Page 275** line 1 of answer 455 ——— 10 Jan 2017

1101 $\rightsquigarrow$ 1011

▶**Page 275** line 1 of answer 457 ——— 21 Jun 2022

$51.3 \times 10^{16} \rightsquigarrow 51.3 \times 10^{15}$

Page 276 new line for answer 465 ——— 14 May 2018

(See also the more general result in Theorem 7.2.2.1S.)

▶**Page 276** line 8 of answer 466 ——— 29 Mar 2022

$\max(i, m-i) \le j \le n/2 \rightsquigarrow 1 \le j \le \min(i, m-i, n/2)$

▶**Page 279** lines 4 and 5 of answer 479(d) ——— 17 Jul 2018

$S_3 = x_{11} \rightsquigarrow S_2 = x_{11}$ and $S_2 = x_{12} \rightsquigarrow S_3 = x_{12}$

▶**Page 280** line 17 ——— 09 May 2019

exercises 477–480 $\rightsquigarrow$ exercises 477–479

Page 281 replacement for second paragraph of answer 483 ——— 25 May 2019

It's not difficult to color the Mycielski graph M_c with c colors (which is the minimum), *without* any symmetry breaking. For example, the 191-vertex graph M_{12} leads to 2,446,271 clauses in 36852 variables (total length 4.9 million); yet 12-color solutions are found by Algorithms C, W, and L respectively in 2.6, 523, and 12200 megamems. The symmetry breaking clauses actually would *retard* that calculation, because those clauses are much longer. On the other hand, when we try to succeed with only $c-1$ colors, those clauses are extremely helpful: The runtime needed by Algorithm C to show that M_6 isn't 5-colorable goes down from 124 Gμ to 32 Mμ! Furthermore, Algorithm L does better here: Its runtime for that problem goes down from 7.5 Gμ to 28 Mμ.

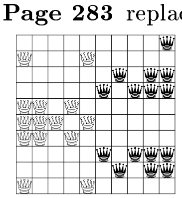
▶**Page 281** line 4 of answer 484 ——— 22 Mar 2020

$\bar{s}_{t,k} \rightsquigarrow \bar{s}_{t,l}$

▶**Page 281** line 2 of answer 485 ——— 22 Mar 2020

$q_{t',l} \rightsquigarrow q_{t',k}$

Page 283 replacement for Fig. A–11(e) (formerly Fig. A–9(e)) ——— 27 Apr 2019



Page 283 replacement for the paragraph after the illustration ——— 27 Apr 2019

The maximum army sizes for $3 \le n \le 13$ are known to be (1, 2, 4, 5, 7, 9, 12, 14, 17, 21, 24); see OEIS sequence A250000. An extra black queen can actually be included in the cases $n = 3$, 4, 6, 8, 10, 11, and 13. Solutions appear in Fig. A–11; the construction shown in Fig. A–11(d) generalizes to armies of $2q(q+1)$ queens whenever $n = 4q+1$, while the one in part (c) belongs to another family of constructions that achieve the higher asymptotic density $\frac{7}{48}n^2$.

Page 283 replacement for the final lines of answer 488 ——— 12 Aug 2019

respectively. [See Martin Gardner ... $\rightsquigarrow$ respectively. [B. M. Smith, K. E. Petrie, and I. P. Gent obtained similar results using CSP methods in *LNCS* **3011** (2004), 271–286.]

(This problem was posed by S. Ainley in his *Mathematical Puzzles* (1977), problem C1. He mentioned solutions for $n \le 30$ that have never yet been beaten, although he obtained them by hand. See also Martin Gardner, *Math Horizons* **7**, 2 (November 1999), 2–16, for generalizations to coexisting armies of sizes r and s. D. M. Kane has proved, among other things, that the maximum value of s, if $r = 3q^2 + 3q + 1$, is asymptotically $n^2 - (6q+3)n + O(1)$ [arXiv:1703.04538 [math.CO] (2017), 19 pages].)

▸**Page 283** line 5 of answer 491 ——— 21 Mar 2020

is a written as $\rightsquigarrow$ is written as

▸**Page 284** line 1 of answer 494 ——— 17 Jul 2018

Exercise 504(b) $\rightsquigarrow$ Exercise 475(d)

▸**Page 286** line 1 of answer 513 ——— 17 Mar 2020

choose the values $\rightsquigarrow$ chose the values

Page 288 lines 3 and 7 ——— 09 Feb 2016

$S_{1kk} \wedge \bar{Z}_{11} \wedge Z_{12}$ becomes $(S_{1kk}) \wedge (\bar{Z}_{11}) \wedge (Z_{12})$ and $S_{553} \wedge \bar{Z}_{17} \rightsquigarrow (S_{553}) \wedge (\bar{Z}_{17})$

Page 288 in answer 516 ——— 04 Feb 2021

line 2: there exists $\rightsquigarrow$ we can know
line -2: no randomized $\rightsquigarrow$ no knowable randomized

Page 288 replacement for answer 517(a) ——— 11 Jan 2016

517. (a) (Solution by Günter Rote.) Replace the jth ternary clause $(l_j \vee l'_j \vee l''_j)$ by three ternary equations $l_j + a_j + c_j = 1$, $\bar{l}'_j + a_j + b_j = 1$, $\bar{l}''_j + c_j + d_j = 1$, where a_j, b_j, c_j, and d_j are new variables.

Page 290 or 291 (in third paragraph of answer 525) ——— 21 Jun 2022

$5n$ rows and $3n$ columns, ... in each row $\rightsquigarrow$ $3n$ items and $5n$ options, with five options for each item and three items in each option

Page 291 last lines of answer 525 ——— 09 Feb 2022

(On the other hand ... unsatisfiable.) $\rightsquigarrow$ (The problem of this kind that defeated all the SAT solvers in the 2014 competition corresponds to an instance of 3D matching that is solved almost instantaneously by the dancing links method: Algorithm 7.2.2.1X needs fewer than 60 Mμ to prove it unsatisfiable. On the other hand, if we encode that 3D matching problem with $3n$ quinary at-least-one and $3n \cdot 10$ binary at-most-one clauses, as in (13), instead of using only $2n$ for at-least-one and $n \cdot 10$ for at-most-one, Algorithm L will be almost as good as dancing links.)

Page 293 and following ———— 19 Mar 2006

Miscellaneous changes to the existing index of Volume 4 Fascicle 6 are collected here, including corrections and amendments to the old entries as well as new entries that are occasioned by the new material. Thus, the lines of the full index that have changed serve also as an index to the present document. However, when a correction or amendment has caused an old index entry to be deleted, the deletion is usually not indicated.