

DEXTRO.ORG

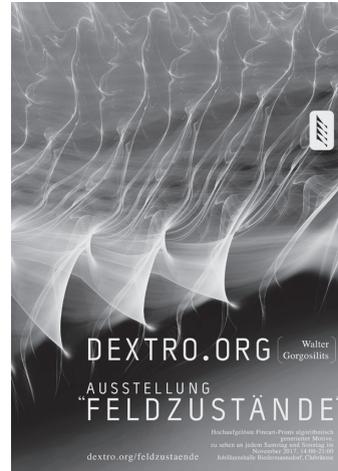
FELDZUSTÄNDE

K 456+ . + CODE . 2018/19

Katalog zu den Ausstellungen „Also Sprach Algorithmi“, Galerie im Turm der Bezirkshauptmannschaft Baden, und „Feldzustände“ („Also Sprach Algorithmi II“), Jubiläumshalle Biedermansdorf (Fineart Prints, A0 Format, 400 dpi)



dextro.org, 1994-2019
dextro@dextro.org



Catalog, originally intended to accompany the following exhibitions:

- 21 June 2016: „Also Sprach Algorismi“ (Images from the Depths of the Algorithmic Parallel Universe), Galerie im Turm der Bezirkshauptmannschaft Baden, Schwartzstraße 50, 2500 Baden
- 21 January 2017 (canceled): „Also Sprach Algorismi 2“, Dellago, Payergasse 10, 1160 Vienna
- November 2017: „Feldzustände“, Jubiläumshalle, 2362 Biedermannsdorf

Biedermannsdorf, 31 December 2018

Print: Flyeralarm, Patron: Oskar Obereder, Fonts: Times, Typestar

Both exhibitions showed fineart-prints in A0 format (at 400 dpi), printed by Digital Laut, Ziegelofengasse 29, Vienna 1050 (digitallaut.at). These prints are depicted here as they were shown, with the code displayed underneath the image.

Katalog, eigentlich gedacht für die Ausstellungen:

- 21. Juni - 3. Juli 2016: „Also Sprach Algorismi“ (Bilder aus den Tiefen des algorithmischen Paralleluniversums), Galerie im Turm der Bezirkshauptmannschaft Baden, Schwartzstraße 50, 2500 Baden
- 21. Januar - 21. März 2017 (abgesagt): „Also Sprach Algorismi 2“, Dellago, Payergasse 10, 1160 Wien
- November 2017: „Feldzustände“, Jubiläumshalle, 2362 Biedermannsdorf

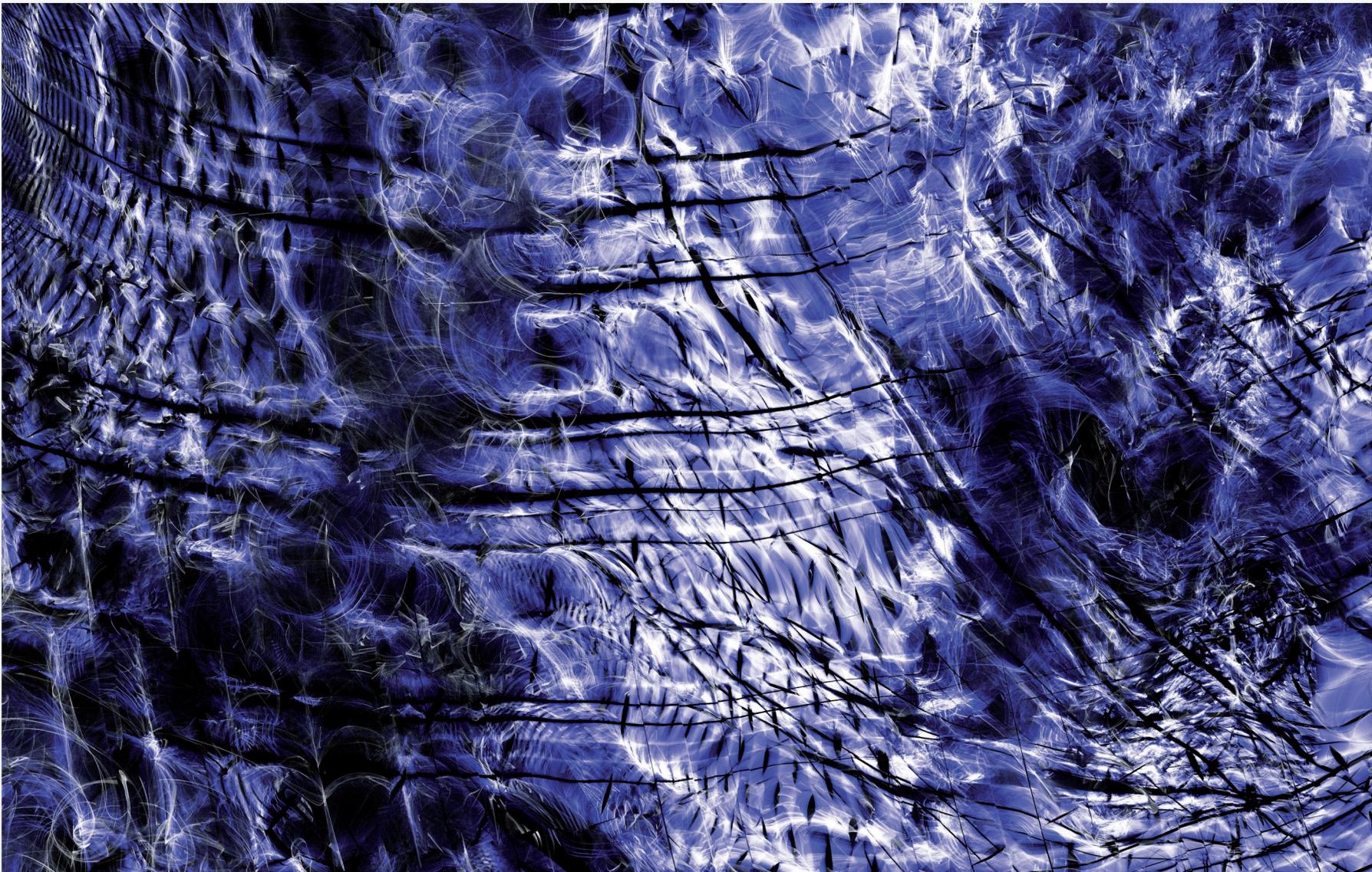
Biedermannsdorf, 31. Dezember 2018

Druck: Flyeralarm, Mäzen: Oskar Obereder, Fonts: Times, Typestar

Beide Ausstellungen zeigten Fineart Prints im Format A0 (bei 400 dpi), hergestellt von Digital Laut, Ziegelofengasse 29, 1050 Wien (digitallaut.at). Diese Prints sind hier so abgebildet, wie sie dort zu sehen waren, mit dem Code unter den Bildern.

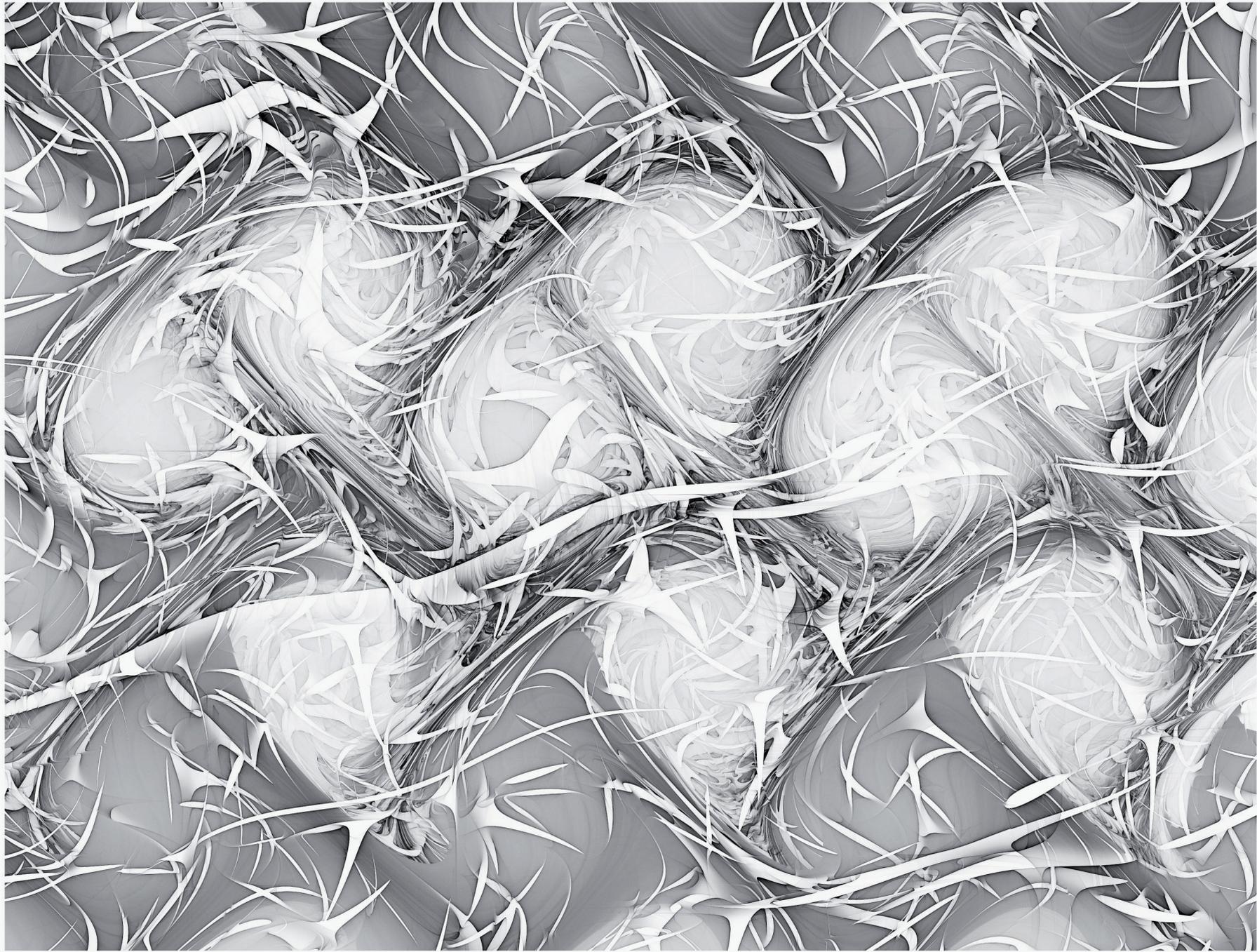


m.u.t.i.o.c. 

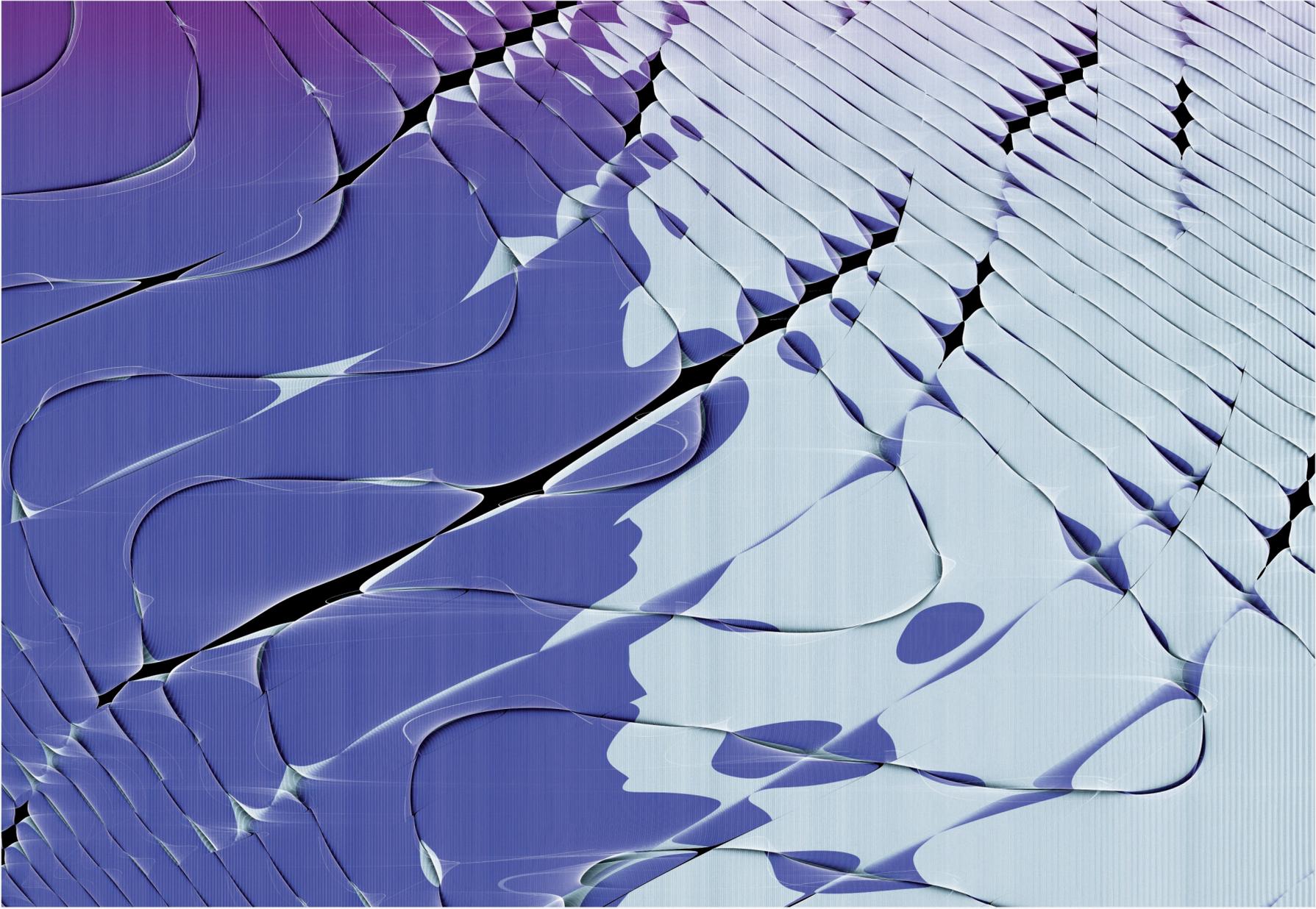


0000

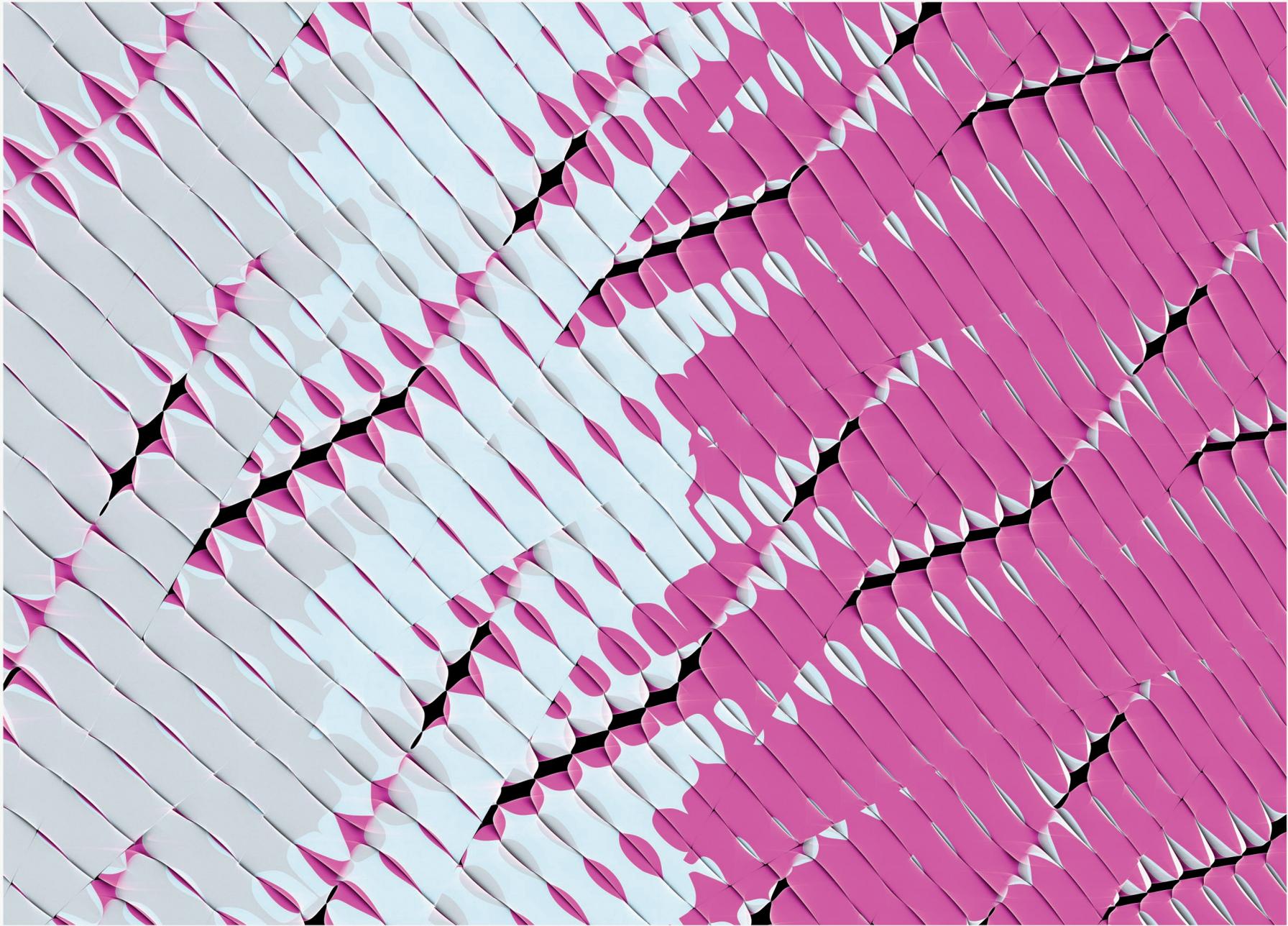
Il presente documento è un prodotto di ricerca e non deve essere considerato un'offerta o una sollecitazione di acquisto o di vendita di titoli o di servizi finanziari. Il presente documento è destinato ai soli destinatari per i quali è stato preparato e non deve essere distribuito o reso pubblico. Il presente documento non costituisce e non deve essere considerato un'offerta o una sollecitazione di acquisto o di vendita di titoli o di servizi finanziari. Il presente documento è destinato ai soli destinatari per i quali è stato preparato e non deve essere distribuito o reso pubblico. Il presente documento non costituisce e non deve essere considerato un'offerta o una sollecitazione di acquisto o di vendita di titoli o di servizi finanziari. Il presente documento è destinato ai soli destinatari per i quali è stato preparato e non deve essere distribuito o reso pubblico.



© 2008 The McGraw-Hill Companies. All rights reserved. This publication is protected by copyright. Any unauthorized distribution or reproduction of this work is strictly prohibited. For more information, contact the publisher. The McGraw-Hill Companies, 1221 Avenue of the Americas, New York, NY 10020-1346. ISBN 978-0-07-143288-9. Printed in the United States of America. 10 9 8 7 6 5 4 3 2 1



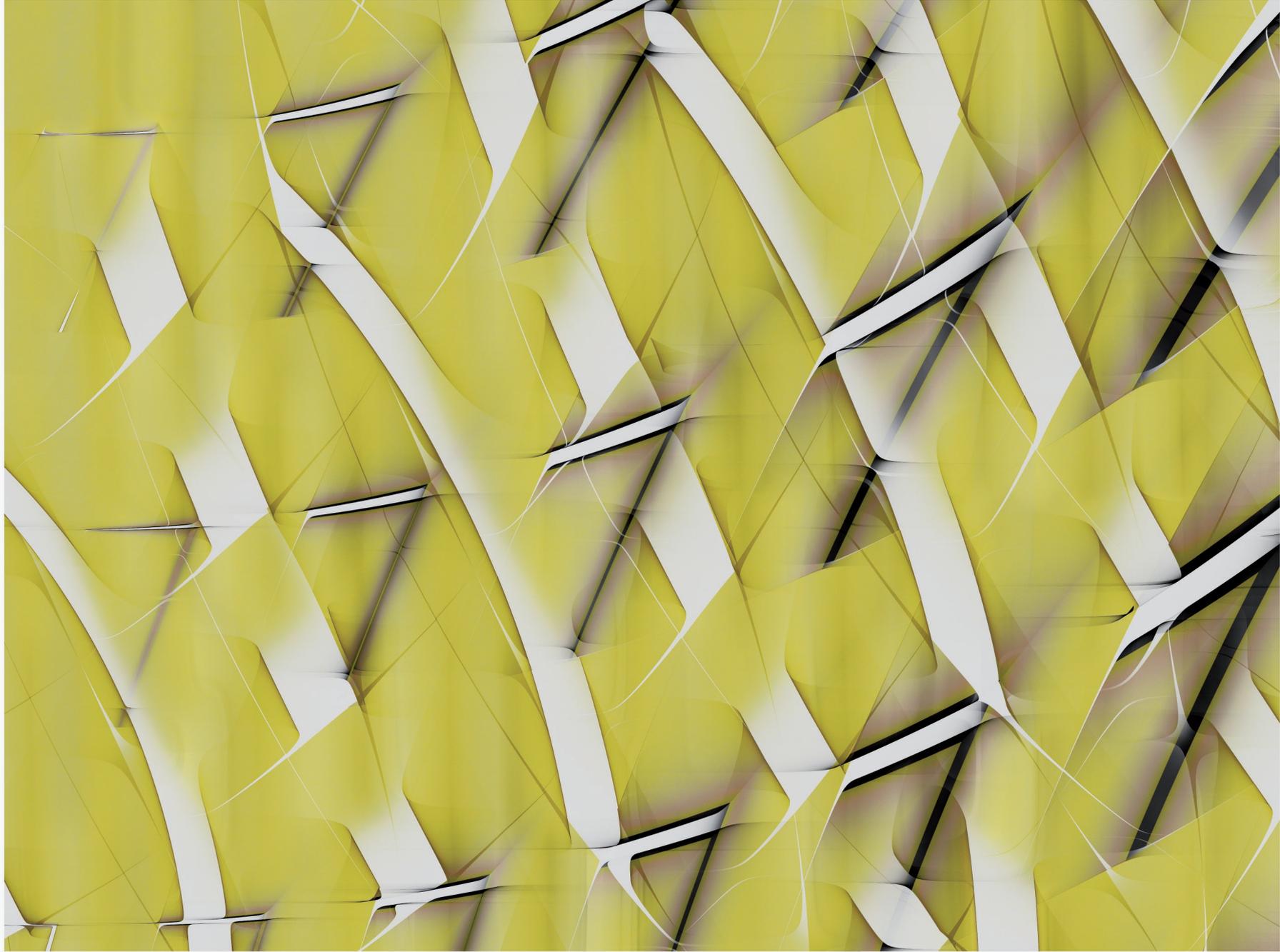
© 2015 Adobe Systems Incorporated. All rights reserved. Adobe, the Adobe logo, and Adobe Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Adobe Reader is a trademark of Adobe Systems Incorporated. All other trademarks are the property of their respective owners.



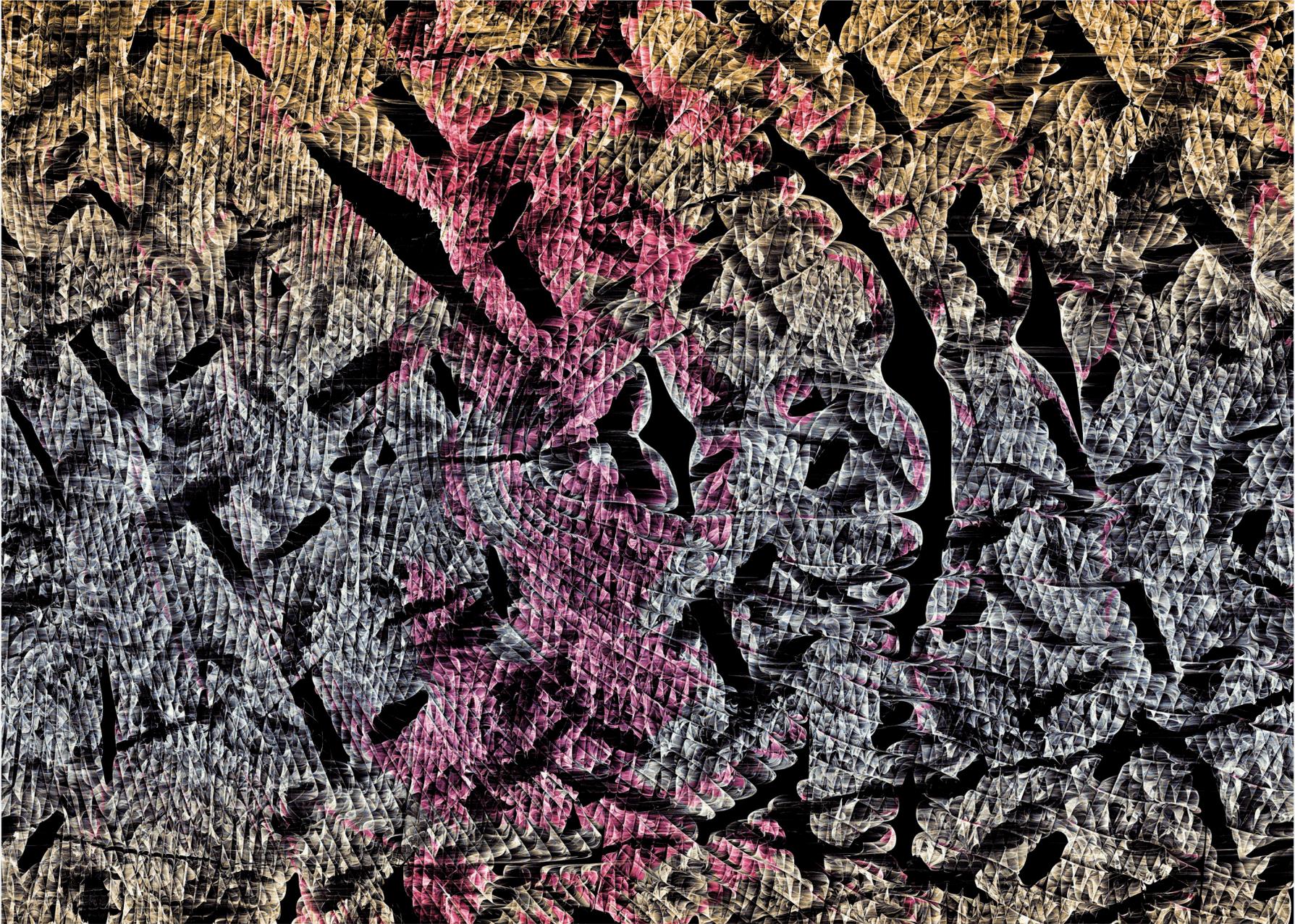
© 2011 The McGraw-Hill Companies. All rights reserved. This publication is protected by copyright. Any unauthorized use or distribution of this work is strictly prohibited. This work may be used for educational or promotional purposes, provided that the copyright notice is included on all copies. For more information, contact The McGraw-Hill Companies, 1221 Avenue of the Americas, New York, NY 10020-1345. ISBN 978-0-07-337551-2. Printed in the United States of America. 10 9 8 7 6 5 4 3 2 1



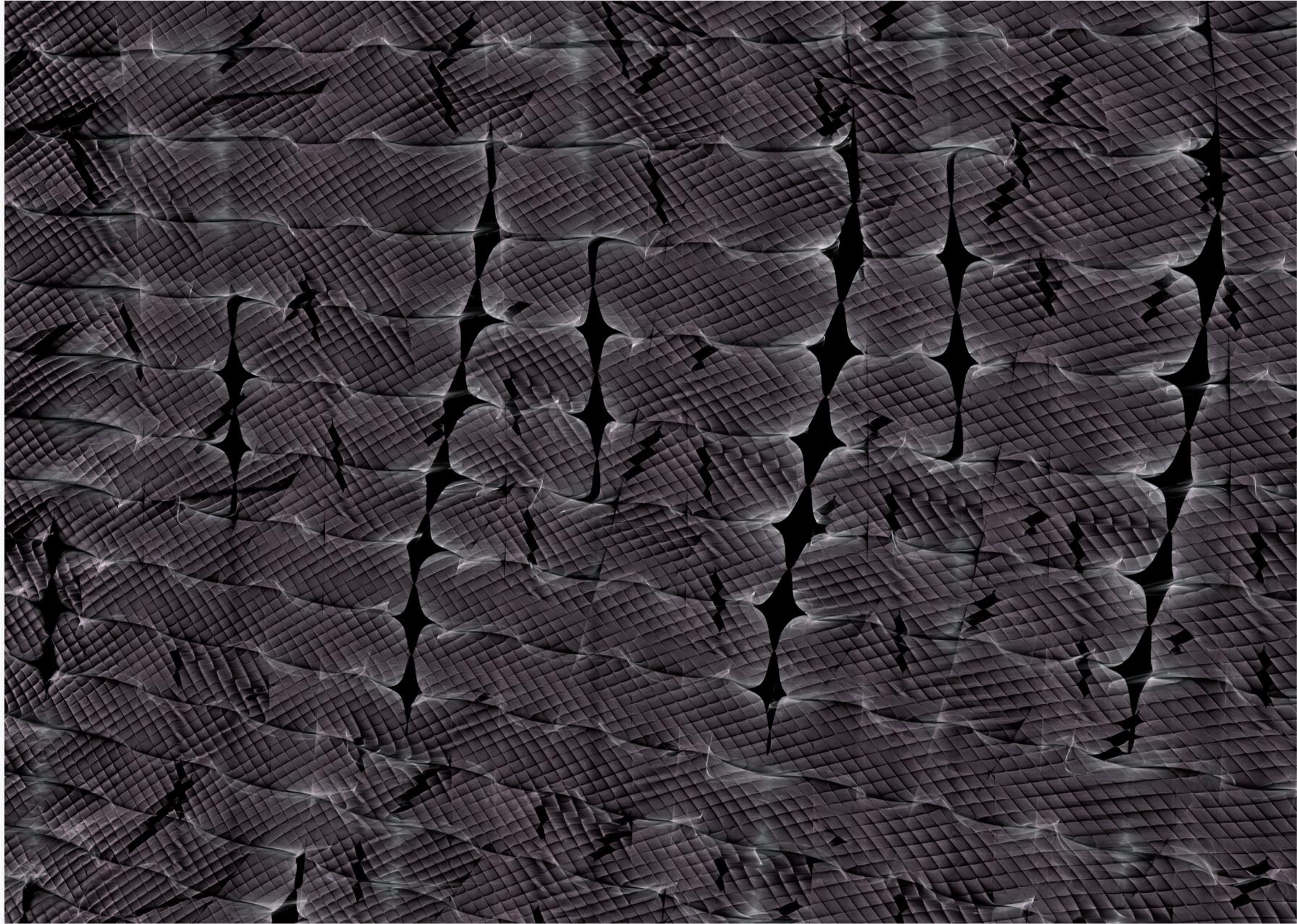
© 2011 The McGraw-Hill Companies. All rights reserved. This publication is protected by copyright. Any unauthorized use or distribution of this work is strictly prohibited. For more information, contact the publisher at 1221 Avenue of the Americas, New York, NY 10020-1095. ISBN 978-0-07-337551-2. Printed in the United States of America. 10 9 8 7 6 5 4 3 2 1



© 2013 Intel Corporation. Intel, the Intel logo, and other marks contained herein are trademarks of Intel Corporation or its subsidiaries in the United States and other countries. All other marks contained herein are the property of their respective owners. Intel, the Intel logo, and other marks contained herein are trademarks of Intel Corporation or its subsidiaries in the United States and other countries. All other marks contained herein are the property of their respective owners.



© 2014 The McGraw-Hill Companies. All rights reserved. This publication is protected by copyright. Any unauthorized use, reproduction, or distribution of this work is prohibited. For more information, contact the publisher. Printed in the United States of America. This book is printed on acid-free paper. 10 9 8 7 6 5 4 3 2 1



Small, illegible text located in the bottom right corner of the page, likely a page number or a small caption.



© 2017 ZZZ. All rights reserved. This document is the property of ZZZ and is intended for internal use only. It contains confidential information and is not to be distributed outside the organization. Any unauthorized use or disclosure of this information is strictly prohibited. For more information, please contact the ZZZ legal department.



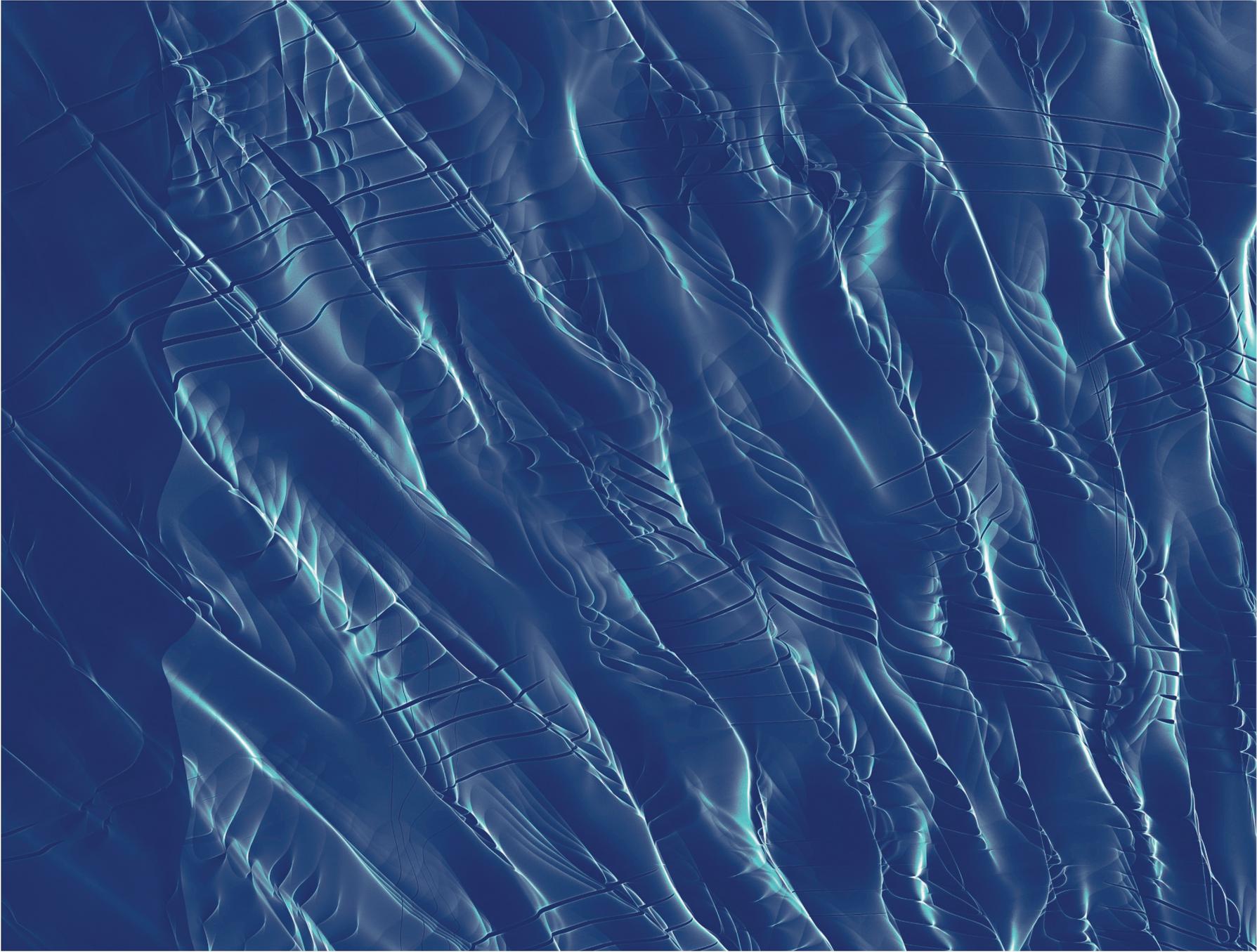
© 2004 Adobe Systems Incorporated. All rights reserved. Adobe, the Adobe logo, and the Adobe Type Kit logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Adobe Reader is a trademark of Adobe Systems Incorporated. All other trademarks are the property of their respective owners.



Small vertical text or fine print located on the right edge of the page, likely containing copyright information or a credit line.



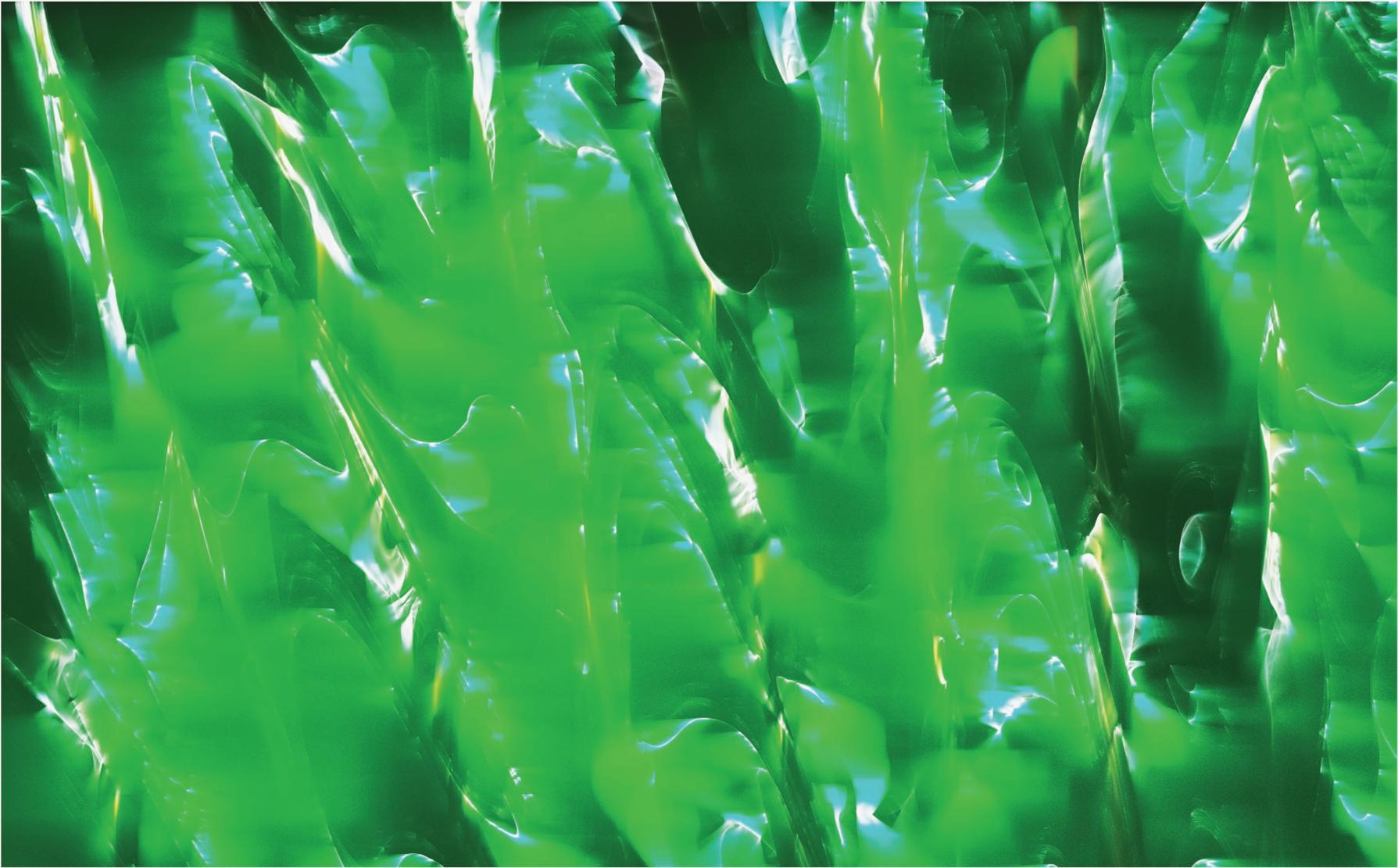
© 2011 The McGraw-Hill Companies. All rights reserved. This publication is protected by copyright. Any unauthorized use or distribution of this work is strictly prohibited. For more information, contact the publisher. The McGraw-Hill Companies, 1221 Avenue of the Americas, New York, NY 10020-1345. ISBN 978-0-07-309823-2. Printed in the United States of America. 10 9 8 7 6 5 4 3 2 1



Small, illegible text located at the bottom right corner of the page, likely a copyright notice or technical information.



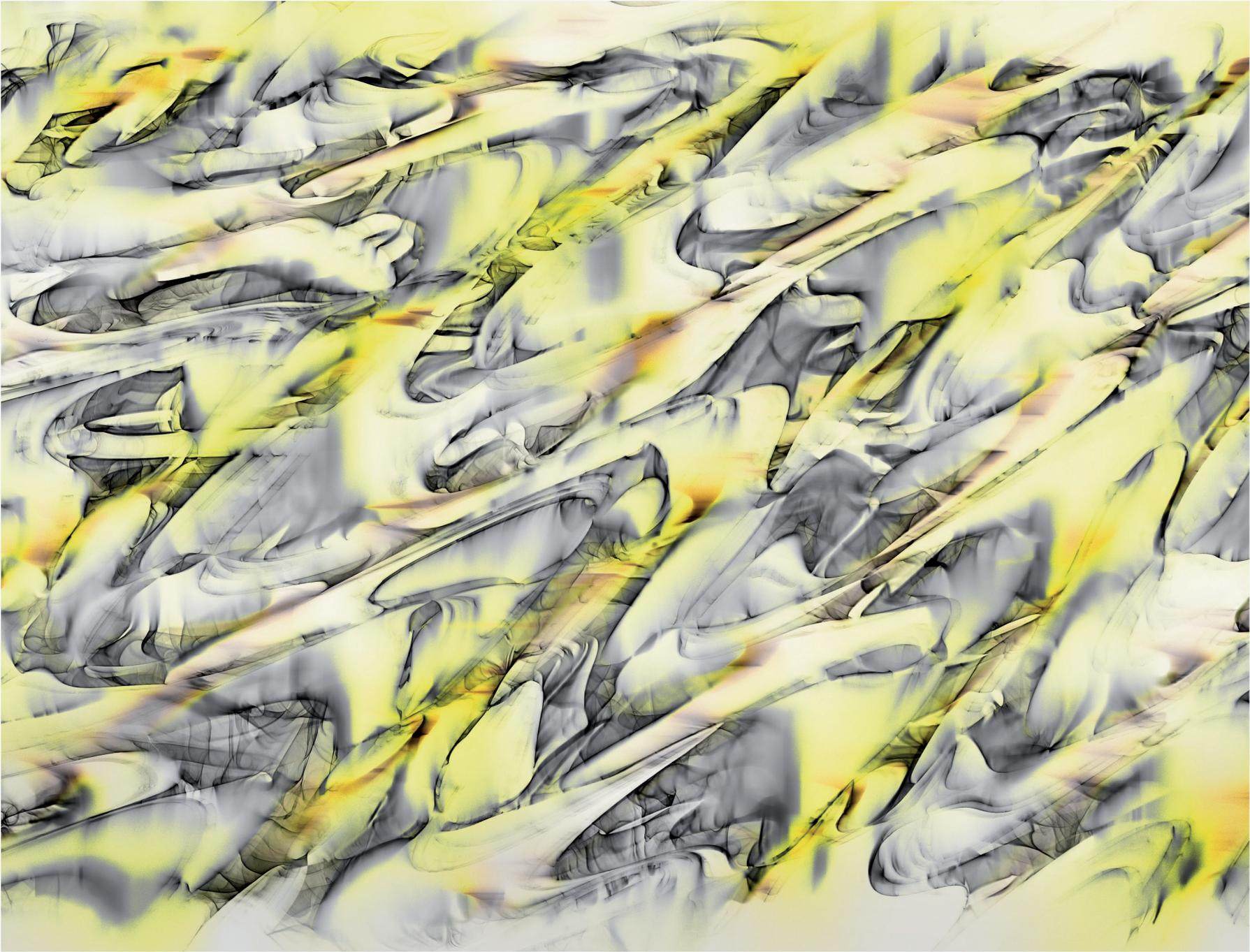
© 2013 The McGraw-Hill Companies. All rights reserved. This publication is protected by copyright. Any unauthorized distribution, reproduction, or use of any part of this work is prohibited. For more information, contact the publisher at 1221 Avenue of the Americas, New York, NY 10020-1095. ISBN 978-0-07-337551-2. Printed in the United States of America. 10 9 8 7 6 5 4 3 2 1



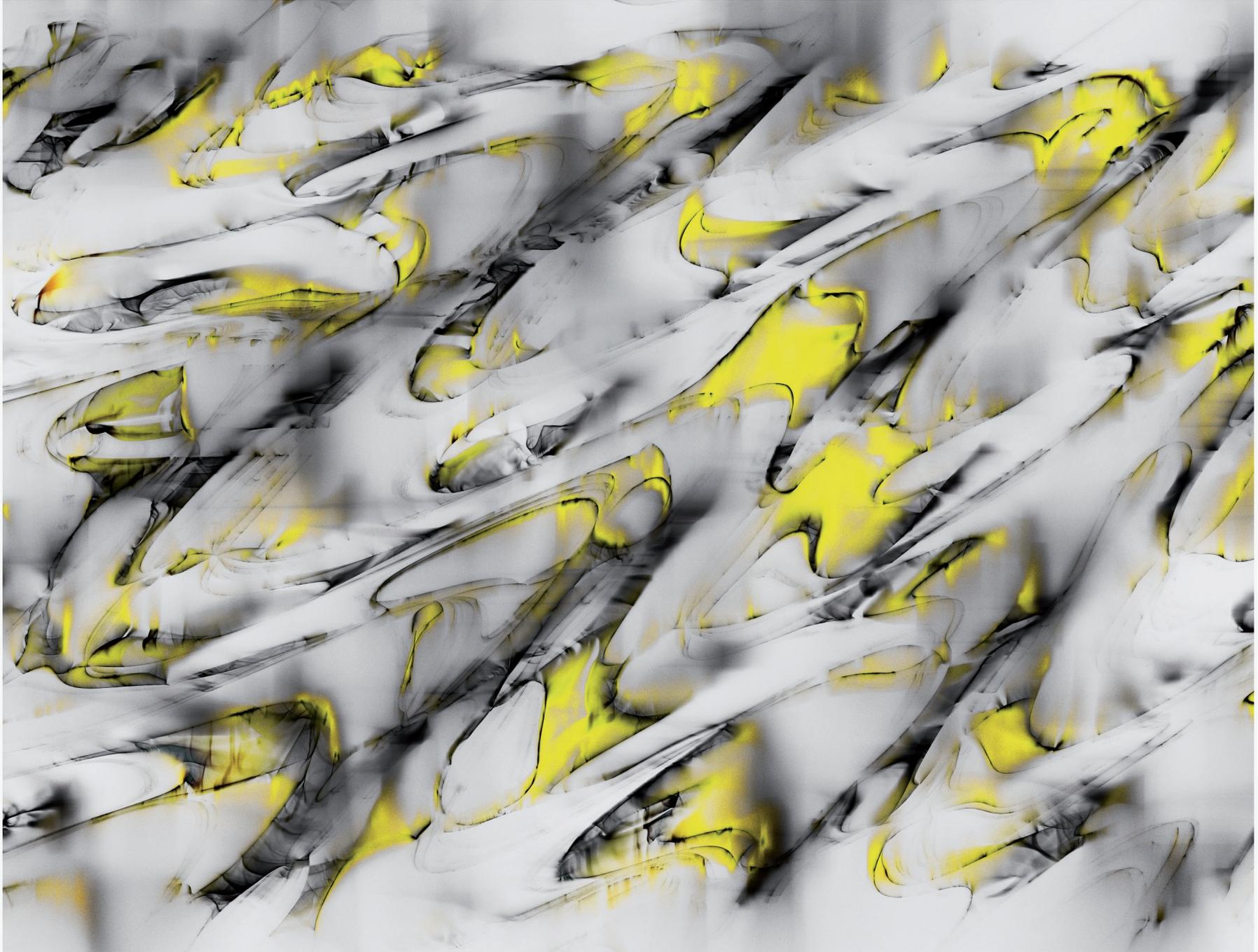
© 2014 by the author. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the publisher. This publication is intended for informational purposes only and is not intended to be used as a substitute for professional advice. The publisher disclaims any liability for any damages, including consequential, special, or exemplary damages, that may result from the use of the information contained herein. The publisher also disclaims any liability for any errors or omissions that may appear in this publication.



© 2014 Adobe Systems Incorporated. All rights reserved. Adobe, the Adobe logo, and the Adobe Creative Cloud logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. All other marks contained herein are the property of their respective owners.



© 2010 Adobe Systems Incorporated. All rights reserved. Adobe, the Adobe logo, and the Adobe Creative Cloud logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. All other marks contained herein are the property of their respective owners.



Small vertical text or fine print on the right edge of the page, likely containing copyright information or technical details.

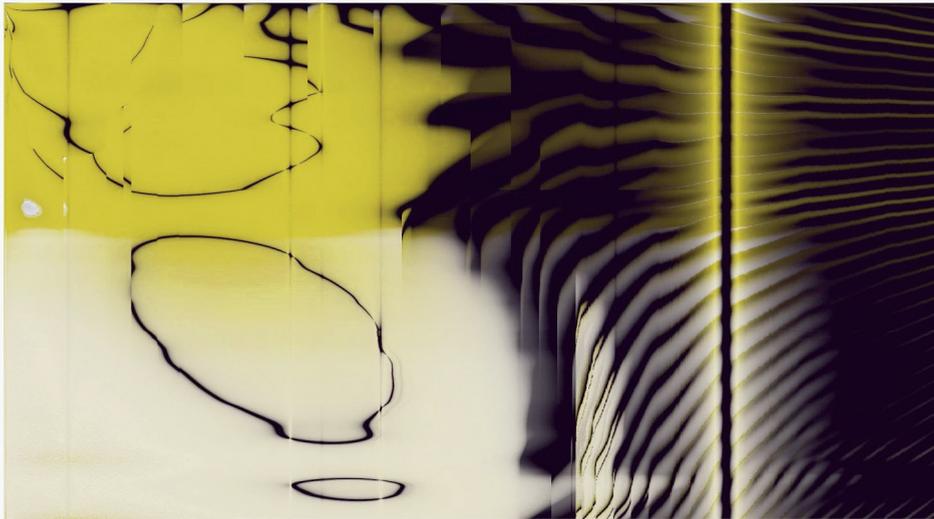


Small, illegible text block at the bottom of the top-left image, likely a technical specification or copyright notice.



Small, illegible text block at the bottom of the top-right image, likely a technical specification or copyright notice.

+



Small, illegible text block at the bottom of the bottom-left image, likely a technical specification or copyright notice.



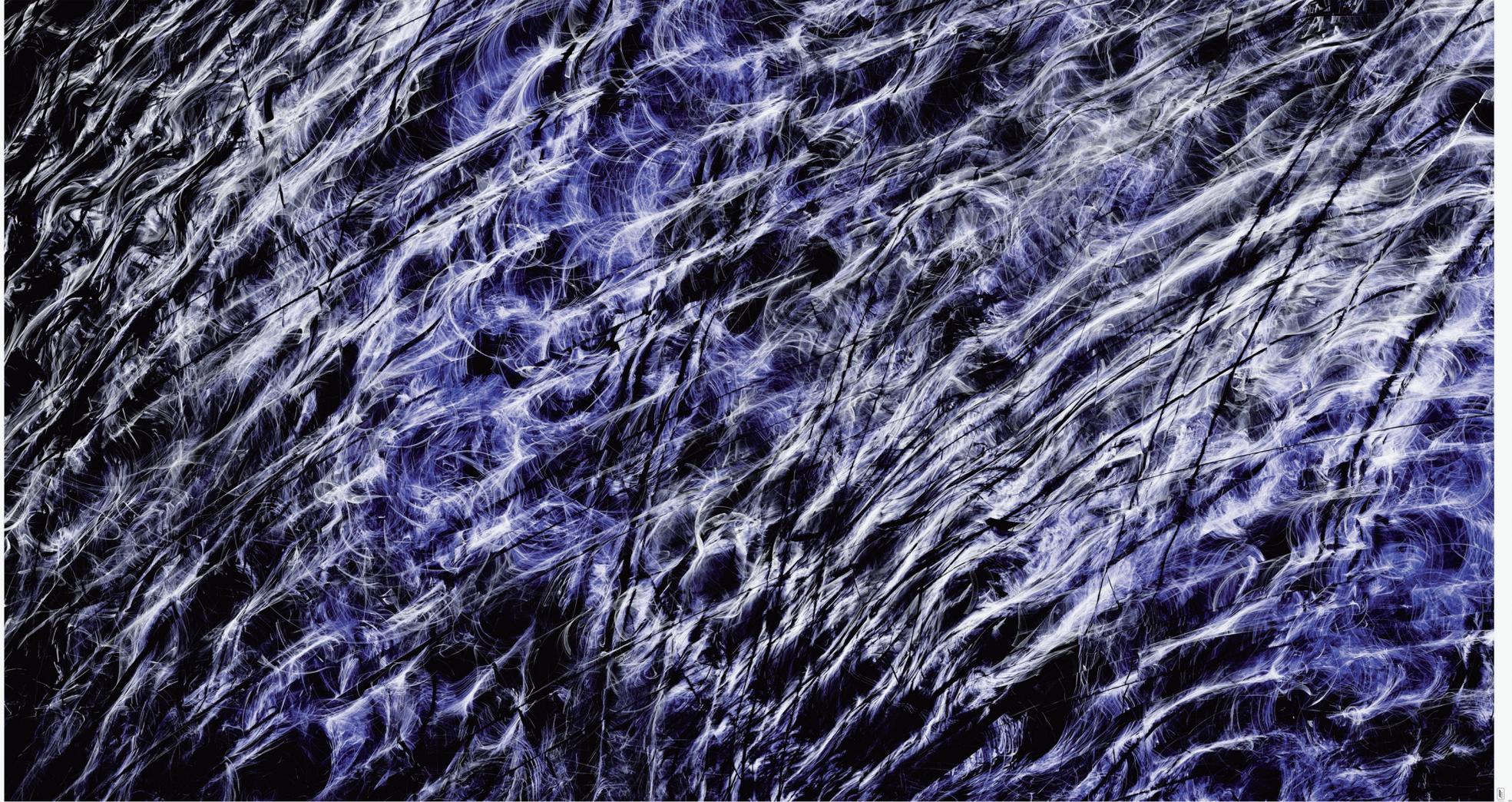
Small, illegible text block at the bottom of the bottom-right image, likely a technical specification or copyright notice.



© 2010 Bentley Systems, Incorporated. Bentley, the "B" Bentley logo, MicroStation, and MicroStation V8i are either registered trademarks or trademarks of Bentley Systems, Incorporated or Bentley Software, Inc. All other marks contained herein are the property of their respective owners. Bentley reserves the right to change the specifications and features of the software without notice. Bentley does not warrant the accuracy or completeness of the information provided in this document. Bentley is not responsible for any errors or for any consequences arising from the use of the information contained herein. Bentley disclaims any liability for damages, including consequential, special, or exemplary damages, arising from the use of the information contained herein. Bentley is not responsible for any delays or interruptions in the service of the software. Bentley is not responsible for any loss of data or information. Bentley is not responsible for any unauthorized use of the software. Bentley is not responsible for any third-party claims against the user. Bentley is not responsible for any software bugs or errors. Bentley is not responsible for any hardware damage. Bentley is not responsible for any network connectivity issues. Bentley is not responsible for any system downtime. Bentley is not responsible for any data corruption. Bentley is not responsible for any security breaches. Bentley is not responsible for any illegal use of the software. Bentley is not responsible for any misuse of the software. Bentley is not responsible for any unauthorized disclosure of confidential information. Bentley is not responsible for any violation of applicable laws or regulations. Bentley is not responsible for any breach of contract. Bentley is not responsible for any breach of warranty. Bentley is not responsible for any breach of license. Bentley is not responsible for any breach of privacy. Bentley is not responsible for any breach of intellectual property rights. Bentley is not responsible for any breach of trade secrets. Bentley is not responsible for any breach of confidentiality. Bentley is not responsible for any breach of security. Bentley is not responsible for any breach of data integrity. Bentley is not responsible for any breach of data availability. Bentley is not responsible for any breach of data accuracy. Bentley is not responsible for any breach of data consistency. Bentley is not responsible for any breach of data reliability. Bentley is not responsible for any breach of data security. Bentley is not responsible for any breach of data privacy. Bentley is not responsible for any breach of data protection. Bentley is not responsible for any breach of data governance. Bentley is not responsible for any breach of data compliance. Bentley is not responsible for any breach of data ethics. Bentley is not responsible for any breach of data transparency. Bentley is not responsible for any breach of data accountability. Bentley is not responsible for any breach of data responsibility. Bentley is not responsible for any breach of data ownership. Bentley is not responsible for any breach of data control. Bentley is not responsible for any breach of data access. Bentley is not responsible for any breach of data use. Bentley is not responsible for any breach of data disposal. Bentley is not responsible for any breach of data retention. Bentley is not responsible for any breach of data archiving. Bentley is not responsible for any breach of data backup. Bentley is not responsible for any breach of data recovery. Bentley is not responsible for any breach of data migration. Bentley is not responsible for any breach of data integration. Bentley is not responsible for any breach of data interoperability. Bentley is not responsible for any breach of data compatibility. Bentley is not responsible for any breach of data portability. Bentley is not responsible for any breach of data transfer. Bentley is not responsible for any breach of data exchange. Bentley is not responsible for any breach of data communication. Bentley is not responsible for any breach of data distribution. Bentley is not responsible for any breach of data dissemination. Bentley is not responsible for any breach of data publication. Bentley is not responsible for any breach of data presentation. Bentley is not responsible for any breach of data visualization. Bentley is not responsible for any breach of data analysis. Bentley is not responsible for any breach of data interpretation. Bentley is not responsible for any breach of data understanding. Bentley is not responsible for any breach of data knowledge. Bentley is not responsible for any breach of data wisdom. Bentley is not responsible for any breach of data insight. Bentley is not responsible for any breach of data intelligence. Bentley is not responsible for any breach of data expertise. Bentley is not responsible for any breach of data proficiency. Bentley is not responsible for any breach of data skill. Bentley is not responsible for any breach of data capability. Bentley is not responsible for any breach of data competence. Bentley is not responsible for any breach of data performance. Bentley is not responsible for any breach of data productivity. Bentley is not responsible for any breach of data efficiency. Bentley is not responsible for any breach of data effectiveness. Bentley is not responsible for any breach of data quality. Bentley is not responsible for any breach of data quantity. Bentley is not responsible for any breach of data value. Bentley is not responsible for any breach of data utility. Bentley is not responsible for any breach of data benefit. Bentley is not responsible for any breach of data advantage. Bentley is not responsible for any breach of data opportunity. Bentley is not responsible for any breach of data potential. Bentley is not responsible for any breach of data possibility. Bentley is not responsible for any breach of data probability. Bentley is not responsible for any breach of data likelihood. Bentley is not responsible for any breach of data chance. Bentley is not responsible for any breach of data risk. Bentley is not responsible for any breach of data hazard. Bentley is not responsible for any breach of data danger. Bentley is not responsible for any breach of data threat. Bentley is not responsible for any breach of data peril. Bentley is not responsible for any breach of data jeopardy. Bentley is not responsible for any breach of data loss. Bentley is not responsible for any breach of data damage. Bentley is not responsible for any breach of data destruction. Bentley is not responsible for any breach of data annihilation. Bentley is not responsible for any breach of data erasure. Bentley is not responsible for any breach of data deletion. Bentley is not responsible for any breach of data removal. Bentley is not responsible for any breach of data obliteration. Bentley is not responsible for any breach of data extinction. Bentley is not responsible for any breach of data annihilation. Bentley is not responsible for any breach of data erasure. Bentley is not responsible for any breach of data deletion. Bentley is not responsible for any breach of data removal. Bentley is not responsible for any breach of data obliteration. Bentley is not responsible for any breach of data extinction.



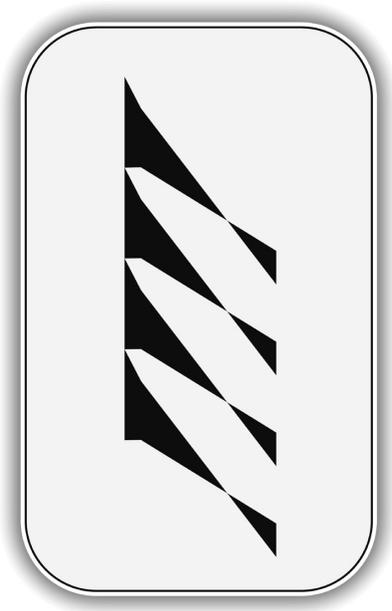
Small vertical text on the right side of the page, likely a page number or a small caption.



© 2011 The McGraw-Hill Companies. All rights reserved. Printed in the United States of America. This publication is protected by copyright. Any unauthorized distribution or reproduction of this work is prohibited. For more information, contact the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923. 0-09-100000-0



© 2011 The McGraw-Hill Companies. All rights reserved. This publication is protected by copyright. Any unauthorized distribution, reproduction, or use of this work is prohibited. For more information, contact The McGraw-Hill Companies, 1221 Avenue of the Americas, New York, NY 10020-1345. ISBN 978-0-07-337551-3. Printed in the United States of America. 10 9 8 7 6 5 4 3 2 1




```
<num2; i++) {xli[i] = random(a)/10 + 460; ylij[i] = random(b)/5 + 180; for (int i = 0; i < num1/2; i++) {x1 = pxli[2*i]; y1 = pyli[2*i]; x2 = pxli[2*i+1]; y2 = pyli[2*i+1]; dx = abs(x1 - x2); dy = abs(y1 - y2); d1 = sqrt(dx*dx + dy*dy); if (d1 == 0) {d1 = 1; d1lij[i] = d1;}} void draw(m = m + 1; if (m == 30) {m = 0; m3 = m3 + 1; for (int i = 0; i < num2; i++) {xli[i] = random(a)/10 + 460; ylij[i] = random(b)/5 + 180;}} for (int i = 0; i < num1/2; i++) {x1 = pxli[2*i]; y1 = pyli[2*i]; x2 = pxli[2*i+1]; y2 = pyli[2*i+1]; dx = abs(x1 - x2); dy = abs(y1 - y2); d1 = sqrt(dx*dx + dy*dy); if (d1 == 0) {d1 = 1; d1lij[i] = d1;}} for (int n = 0; n < num2; n++) {xx = xli[n]; yy = ylij[n]; xxc = xx; yy = yy; dx = abs(xx - x1); dy = abs(yy - y1); d2 = sqrt(dx*dx + dy*dy); dx = abs(xx - x2); dy = abs(yy - y2); d3 = sqrt(dx*dx + dy*dy); xx = xx - (xx - x2)/d1; yy = yy - (yy - y2)/d1; fff = d2/10; xpp = xx + cos(d3*PI/10)*fff; yyp = yy + sin(d3*PI/10)*fff; xx2 = min(6, tan(d2/3)*200); yy2 = min(6, sin(d2/3)*200); yy3 = min(6, atan(d3/4)*200); ff = min(xx2, d3); if (ff == 0.001); xx = xx + (xx - xpp)/ff; ff = max(yy2, yy3); if (ff == 0) {ff = 0.001; yy = yy + (yy - yyp)/ff; if (i == num1/2 - 1) {fff = (yy - 200); if ((fff < -20) || (fff > 200)) {xx = random(a)/10 + 460; yy = random(b)/5 + 180; ffy = (xx - 470); if ((ffy < -20) || (ffy > 200)) {xx = random(a)/10 + 460; yy = random(b)/5 + 180;}} stroke(255-m*8, m*8, m*8, ble); point(fix*12*8 - Sx*3840, fix*12*8 - Sy*2800);} xli[n] = xx; ylij[n] = yy;}} if (m3 == 640){saveFrame(x, y, "#####.tif"); m3 = 0; m = 0; background(0); Sx = Sx + 1; if (Sx == 4) {Sx = 0; Sy = Sy + 1; if (Sy == 4) {Sy = 0;}}}}
```

p18: H006:

```
float m = 0; float a = 1024; float b = 968; float m5; float m4 = 0; float m3 = 0; float x1, x2, y1, y2, dxp, dyp, d1, d2, d3, dx, dy, xxc, yy, dx, dy, xx2, yy2, xx3, yy3, xpp, ypp, ff; int num1 = 30; int num2 = 120000; float[] pxcli = {238, 894, 465, 805, 342, 448, 900, 884, 658, 237, 351, 365, 759, 171, 922, 301, 19, 915, 28, 490, 574, 182, 853, 388, 684, 231, 974, 127, 43, 908}; float[] pycli = {639, 477, 214, 228, 329, 503, 55, 292, 679, 238, 443, 239, 52, 204, 658, 313, 767, 191, 59, 252, 81, 624, 322, 545, 358, 450, 736, 250, 34, 4}; float[] pxli = pxcli; float[] pyli = pycli; float xli = new float[num2]; float yli = new float[num2]; float WW = 0; float WW3 = 0; float slow = 10; int X = (7+94*2)*25; int DET3 = 1; int DET_X, DET_Y; void setup() {size(1920, 1080, P2D); smooth(); background(0); frameRate(255, 5); stroke(255, 5); frameRate(400); for (int i = 0; i < xli.length; i++) {xli[i] = random(a); ylij[i] = random(b);} while (m4 < 4) {m4 = m4 + 1; WW = WW + 0.001; WW3 = WW3 + 0.001; for (int i = 0; i < pxli.length; i++) {pxcli[i] = pxcli[i] + (cos(WW*PI/180) + 1)*2/2/slow; pycli[i] = pycli[i] + (sin(WW*PI/180) + 1)*2/2/slow;}} m5 = -600 + WW3 - (sin(WW*90*PI/180) - 1); for (int i = 0; i < xli.length; i++) {xli[i] = random(a); ylij[i] = random(b);} void draw() {m = m + 1; stroke(min(255, 126+5*m), max(0, 255-6*m), max(0, 205-5*m), 5); if (m == 50) {m = 0; m3 = m3 + 1; for (int i = 0; i < xli.length; i++) {xli[i] = random(a); ylij[i] = random(b);} m = 0; for (int i = 0; i < num1/2; i++) {x1 = pxli[2*i]; y1 = pyli[2*i]; x2 = pxli[2*i+1]; y2 = pyli[2*i+1]; dpx = abs(x1 - x2); dpy = abs(y1 - y2); d1 = sqrt(dpx*dpx + dpy*dpy) + 1; for (int n = 0; n < num2; n++) {xx = xli[n]; yy = ylij[n]; xxc = xx; yy = yy; dx = abs(xx - x1); dy = abs(yy - y1); d2 = sqrt(dx*dx + dy*dy); dx = abs(xx - x2); dy = abs(yy - y2); d3 = sqrt(dx*dx + dy*dy) + 1; xx = xx - (xx - x2)/d1; yy = yy - (yy - y2)/d1; xx2 = tan(d2/m5)*100; yy2 = tan(d2/m5)*250; xx3 = tan(d3/m5)*200; xpp = xx + cos(xx2*PI/60)*d2; ypp = yy + sin(yy2*PI/60)*d2; ff = min(xx2, xx3); if (ff == 0) {ff = 0.0001; xx = xx + (xx - xpp)/ff; ff = max(yy2, yy3); if (ff == 0) {ff = 0.0001; yy = yy + (yy - ypp)/ff; if (i == num1/2 - 1) {point(xx*1.8 - 160, yy*1.8 - 30);} if ((xx < 0) || (xx > a)) {xx = random(a);} if ((yy < 0) || (yy > b)) {yy = random(b);} xli[n] = xx; ylij[n] = yy;}}}}
```

p19: H009:

```
float m = 0; float a = 1024; float b = 968; float m5; float m3 = 0; float m4 = 0; float x1, x2, y1, y2, dxp, dyp, d1, d2, d3, dx, dy, xxc, yy, dx, dy, xx2, yy2, xx3, yy3, xpp, ypp, ff; int num1 = 30; int num2 = 60000; float[] pxcli = {19, 915, 28, 490, 574, 182, 853, 388, 684, 231, 974, 127, 43, 908, 238, 894, 465, 805, 342, 448, 900, 884, 658, 237, 351, 365, 759, 171, 922, 301}; float[] pycli = {639, 477, 214, 228, 329, 503, 55, 292, 679, 238, 443, 239, 52, 204, 658, 313, 767, 191, 59, 252, 81, 624, 322, 545, 358, 450, 736, 250, 34, 4}; float[] pxli = pxcli; float[] pyli = pycli; float xli = new float[num2]; float yli = new float[num2]; float WW = 0; float slow = 10; int X = (7+94*2)*25; int DET3 = 1; int DET_X, DET_Y; void setup() {size(1920, 1080, P2D); smooth(); background(0); frameRate(400); stroke(255, 5); stroke(255, 5); frameRate(400); for (int i = 0; i < xli.length; i++) {xli[i] = random(a); ylij[i] = random(b);} while (m4 < 4) {m4 = m4 + 1; WW = WW + 0.001; WW3 = WW3 + 0.001; for (int i = 0; i < pxli.length; i++) {pxcli[i] = pxcli[i] + (cos(WW*PI/180) + 1)*2/2/slow; pycli[i] = pycli[i] + (sin(WW*PI/180) + 1)*2/2/slow;}} m5 = -650 - (sin(WW*90*PI/180) - 1); void draw() {m = m + 1; stroke(min(255, 126+5*m), max(0, 255-6*m), max(0, 205-5*m), 5); if (m == 50) {m = 0; m3 = m3 + 1; for (int i = 0; i < xli.length; i++) {xli[i] = random(a); ylij[i] = random(b);} m = 0; for (int i = 0; i < num1/2; i++) {x1 = pxli[2*i]; y1 = pyli[2*i]; x2 = pxli[2*i+1]; y2 = pyli[2*i+1]; dpx = abs(x1 - x2); dpy = abs(y1 - y2); d1 = sqrt(dpx*dpx + dpy*dpy) + 1; for (int n = 0; n < num2; n++) {xx = xli[n]; yy = ylij[n]; xxc = xx; yy = yy; dx = abs(xx - x1); dy = abs(yy - y1); d2 = sqrt(dx*dx + dy*dy); dx = abs(xx - x2); dy = abs(yy - y2); d3 = sqrt(dx*dx + dy*dy) + 1; xx = xx - (xx - x2)/d1; yy = yy - (yy - y2)/d1; xx2 = tan(d2/m5)*100; yy2 = tan(d2/m5)*250; xx3 = tan(d3/m5)*200; xpp = xx + cos(xx2*PI/60)*d2; ypp = yy + sin(yy2*PI/60)*d2; ff = min(xx2, xx3); if (ff == 0) {ff = 0.0001; xx = xx + (xx - xpp)/ff; ff = max(yy2, yy3); if (ff == 0) {ff = 0.0001; yy = yy + (yy - ypp)/ff; if (i == 1472/4, (yy*1.8 - 30)*2.44 - 1200)*0.7;}} if ((xx < 0) || (xx > a)) {xx = random(a);} if ((yy < 0) || (yy > b)) {yy = random(b);} xli[n] = xx; ylij[n] = yy;}}}}
```

p20: H008:

```
float aa = 1024; float bb = 768; int num = 40000; float[] wli = new float[num]; float[] xli = new float[num]; float[] yli = new float[num]; int m3 = 0; int m4 = 0; int m5 = 0; int m5d = 1; float W = 0; float qv = 1031.3 + sin((m5/2)*PI/180)*10 + 5; float qv = 1114 + cos(((m5+180)/2)*PI/180); float qhc = 1031.3 + sin(((m5-m5d)/2)*PI/180)*10 + 5; float qc = 1114 + cos((m5-m5d+180)/2)*PI/180; float x1, y1, w, xx, yy, d1, d2, d3, d4, d5, xpp, ypp, xx2, yy2, xx3, yy3, x2, y2, ff, rr; float fek, fok, fik, fak; void setup() {size(1920, 1200, P2D); background(0); frameRate(900); smooth(); stroke(255, 5); for (int n = 0; n < num; n++) {wli[n] = random(360); xli[n] = random(aa/3 + 170); ylij[n] = random(bb/2 + 10); fek = 1; fok = 180; fik = 1000; fak = 10; void draw() {m3 = m3 + 1; if (m3 == 200) {ff = 0; for (int n = 0; n < num; n++) {xli[n] = random(aa/3 + 170); ylij[n] = random(bb/2 + 10);} m3 = 0; for (int n = 0; n < num; n++) {wli[n] = wli[n]; xli[n] = xli[n]; ylij[n] = ylij[n]; yy = abs(xx - qb); yy = abs(yj - qv); if (d1 < 1) {d1 = 1; xx = abs(xx - 414); yy = abs(yy - 233); d1 = sqrt(xx*xx + yy*yy); if (d2 < 2) {d1 = 2; xx = abs(xx - 378); yy = abs(yy - 529); d2 = sqrt(xx*xx + yy*yy); if (d3 < 3) {d1 = 3; xx = abs(xx - 159); yy = abs(yy - 452); d3 = sqrt(xx*xx + yy*yy); if (d4 < 4) {d1 = 4; xx = abs(xx - 332); yy = abs(yy - 491); d4 = sqrt(xx*xx + yy*yy); if (d5 < 5) {d1 = 5; xx = abs(xx - 159); yy = abs(yy - 452); d5 = sqrt(xx*xx + yy*yy); if (d6 < 6) {d1 = 6; xx = abs(xx - 159); yy = abs(yy - 452); d6 = sqrt(xx*xx + yy*yy); if (d7 < 7) {d1 = 7; xx = abs(xx - 159); yy = abs(yy - 452); d7 = sqrt(xx*xx + yy*yy); if (d8 < 8) {d1 = 8; xx = abs(xx - 159); yy = abs(yy - 452); d8 = sqrt(xx*xx + yy*yy); if (d9 < 9) {d1 = 9; xx = abs(xx - 159); yy = abs(yy - 452); d9 = sqrt(xx*xx + yy*yy); if (d10 < 10) {d1 = 10; xx = abs(xx - 159); yy = abs(yy - 452); d10 = sqrt(xx*xx + yy*yy); if (d11 < 11) {d1 = 11; xx = abs(xx - 159); yy = abs(yy - 452); d11 = sqrt(xx*xx + yy*yy); if (d12 < 12) {d1 = 12; xx = abs(xx - 159); yy = abs(yy - 452); d12 = sqrt(xx*xx + yy*yy); if (d13 < 13) {d1 = 13; xx = abs(xx - 159); yy = abs(yy - 452); d13 = sqrt(xx*xx + yy*yy); if (d14 < 14) {d1 = 14; xx = abs(xx - 159); yy = abs(yy - 452); d14 = sqrt(xx*xx + yy*yy); if (d15 < 15) {d1 = 15; xx = abs(xx - 159); yy = abs(yy - 452); d15 = sqrt(xx*xx + yy*yy); if (d16 < 16) {d1 = 16; xx = abs(xx - 159); yy = abs(yy - 452); d16 = sqrt(xx*xx + yy*yy); if (d17 < 17) {d1 = 17; xx = abs(xx - 159); yy = abs(yy - 452); d17 = sqrt(xx*xx + yy*yy); if (d18 < 18) {d1 = 18; xx = abs(xx - 159); yy = abs(yy - 452); d18 = sqrt(xx*xx + yy*yy); if (d19 < 19) {d1 = 19; xx = abs(xx - 159); yy = abs(yy - 452); d19 = sqrt(xx*xx + yy*yy); if (d20 < 20) {d1 = 20; xx = abs(xx - 159); yy = abs(yy - 452); d20 = sqrt(xx*xx + yy*yy); if (d21 < 21) {d1 = 21; xx = abs(xx - 159); yy = abs(yy - 452); d21 = sqrt(xx*xx + yy*yy); if (d22 < 22) {d1 = 22; xx = abs(xx - 159); yy = abs(yy - 452); d22 = sqrt(xx*xx + yy*yy); if (d23 < 23) {d1 = 23; xx = abs(xx - 159); yy = abs(yy - 452); d23 = sqrt(xx*xx + yy*yy); if (d24 < 24) {d1 = 24; xx = abs(xx - 159); yy = abs(yy - 452); d24 = sqrt(xx*xx + yy*yy); if (d25 < 25) {d1 = 25; xx = abs(xx - 159); yy = abs(yy - 452); d25 = sqrt(xx*xx + yy*yy); if (d26 < 26) {d1 = 26; xx = abs(xx - 159); yy = abs(yy - 452); d26 = sqrt(xx*xx + yy*yy); if (d27 < 27) {d1 = 27; xx = abs(xx - 159); yy = abs(yy - 452); d27 = sqrt(xx*xx + yy*yy); if (d28 < 28) {d1 = 28; xx = abs(xx - 159); yy = abs(yy - 452); d28 = sqrt(xx*xx + yy*yy); if (d29 < 29) {d1 = 29; xx = abs(xx - 159); yy = abs(yy - 452); d29 = sqrt(xx*xx + yy*yy); if (d30 < 30) {d1 = 30; xx = abs(xx - 159); yy = abs(yy - 452); d30 = sqrt(xx*xx + yy*yy); if (d31 < 31) {d1 = 31; xx = abs(xx - 159); yy = abs(yy - 452); d31 = sqrt(xx*xx + yy*yy); if (d32 < 32) {d1 = 32; xx = abs(xx - 159); yy = abs(yy - 452); d32 = sqrt(xx*xx + yy*yy); if (d33 < 33) {d1 = 33; xx = abs(xx - 159); yy = abs(yy - 452); d33 = sqrt(xx*xx + yy*yy); if (d34 < 34) {d1 = 34; xx = abs(xx - 159); yy = abs(yy - 452); d34 = sqrt(xx*xx + yy*yy); if (d35 < 35) {d1 = 35; xx = abs(xx - 159); yy = abs(yy - 452); d35 = sqrt(xx*xx + yy*yy); if (d36 < 36) {d1 = 36; xx = abs(xx - 159); yy = abs(yy - 452); d36 = sqrt(xx*xx + yy*yy); if (d37 < 37) {d1 = 37; xx = abs(xx - 159); yy = abs(yy - 452); d37 = sqrt(xx*xx + yy*yy); if (d38 < 38) {d1 = 38; xx = abs(xx - 159); yy = abs(yy - 452); d38 = sqrt(xx*xx + yy*yy); if (d39 < 39) {d1 = 39; xx = abs(xx - 159); yy = abs(yy - 452); d39 = sqrt(xx*xx + yy*yy); if (d40 < 40) {d1 = 40; xx = abs(xx - 159); yy = abs(yy - 452); d40 = sqrt(xx*xx + yy*yy); if (d41 < 41) {d1 = 41; xx = abs(xx - 159); yy = abs(yy - 452); d41 = sqrt(xx*xx + yy*yy); if (d42 < 42) {d1 = 42; xx = abs(xx - 159); yy = abs(yy - 452); d42 = sqrt(xx*xx + yy*yy); if (d43 < 43) {d1 = 43; xx = abs(xx - 159); yy = abs(yy - 452); d43 = sqrt(xx*xx + yy*yy); if (d44 < 44) {d1 = 44; xx = abs(xx - 159); yy = abs(yy - 452); d44 = sqrt(xx*xx + yy*yy); if (d45 < 45) {d1 = 45; xx = abs(xx - 159); yy = abs(yy - 452); d45 = sqrt(xx*xx + yy*yy); if (d46 < 46) {d1 = 46; xx = abs(xx - 159); yy = abs(yy - 452); d46 = sqrt(xx*xx + yy*yy); if (d47 < 47) {d1 = 47; xx = abs(xx - 159); yy = abs(yy - 452); d47 = sqrt(xx*xx + yy*yy); if (d48 < 48) {d1 = 48; xx = abs(xx - 159); yy = abs(yy - 452); d48 = sqrt(xx*xx + yy*yy); if (d49 < 49) {d1 = 49; xx = abs(xx - 159); yy = abs(yy - 452); d49 = sqrt(xx*xx + yy*yy); if (d50 < 50) {d1 = 50; xx = abs(xx - 159); yy = abs(yy - 452); d50 = sqrt(xx*xx + yy*yy); if (d51 < 51) {d1 = 51; xx = abs(xx - 159); yy = abs(yy - 452); d51 = sqrt(xx*xx + yy*yy); if (d52 < 52) {d1 = 52; xx = abs(xx - 159); yy = abs(yy - 452); d52 = sqrt(xx*xx + yy*yy); if (d53 < 53) {d1 = 53; xx = abs(xx - 159); yy = abs(yy - 452); d53 = sqrt(xx*xx + yy*yy); if (d54 < 54) {d1 = 54; xx = abs(xx - 159); yy = abs(yy - 452); d54 = sqrt(xx*xx + yy*yy); if (d55 < 55) {d1 = 55; xx = abs(xx - 159); yy = abs(yy - 452); d55 = sqrt(xx*xx + yy*yy); if (d56 < 56) {d1 = 56; xx = abs(xx - 159); yy = abs(yy - 452); d56 = sqrt(xx*xx + yy*yy); if (d57 < 57) {d1 = 57; xx = abs(xx - 159); yy = abs(yy - 452); d57 = sqrt(xx*xx + yy*yy); if (d58 < 58) {d1 = 58; xx = abs(xx - 159); yy = abs(yy - 452); d58 = sqrt(xx*xx + yy*yy); if (d59 < 59) {d1 = 59; xx = abs(xx - 159); yy = abs(yy - 452); d59 = sqrt(xx*xx + yy*yy); if (d60 < 60) {d1 = 60; xx = abs(xx - 159); yy = abs(yy - 452); d60 = sqrt(xx*xx + yy*yy); if (d61 < 61) {d1 = 61; xx = abs(xx - 159); yy = abs(yy - 452); d61 = sqrt(xx*xx + yy*yy); if (d62 < 62) {d1 = 62; xx = abs(xx - 159); yy = abs(yy - 452); d62 = sqrt(xx*xx + yy*yy); if (d63 < 63) {d1 = 63; xx = abs(xx - 159); yy = abs(yy - 452); d63 = sqrt(xx*xx + yy*yy); if (d64 < 64) {d1 = 64; xx = abs(xx - 159); yy = abs(yy - 452); d64 = sqrt(xx*xx + yy*yy); if (d65 < 65) {d1 = 65; xx = abs(xx - 159); yy = abs(yy - 452); d65 = sqrt(xx*xx + yy*yy); if (d66 < 66) {d1 = 66; xx = abs(xx - 159); yy = abs(yy - 452); d66 = sqrt(xx*xx + yy*yy); if (d67 < 67) {d1 = 67; xx = abs(xx - 159); yy = abs(yy - 452); d67 = sqrt(xx*xx + yy*yy); if (d68 < 68) {d1 = 68; xx = abs(xx - 159); yy = abs(yy - 452); d68 = sqrt(xx*xx + yy*yy); if (d69 < 69) {d1 = 69; xx = abs(xx - 159); yy = abs(yy - 452); d69 = sqrt(xx*xx + yy*yy); if (d70 < 70) {d1 = 70; xx = abs(xx - 159); yy = abs(yy - 452); d70 = sqrt(xx*xx + yy*yy); if (d71 < 71) {d1 = 71; xx = abs(xx - 159); yy = abs(yy - 452); d71 = sqrt(xx*xx + yy*yy); if (d72 < 72) {d1 = 72; xx = abs(xx - 159); yy = abs(yy - 452); d72 = sqrt(xx*xx + yy*yy); if (d73 < 73) {d1 = 73; xx = abs(xx - 159); yy = abs(yy - 452); d73 = sqrt(xx*xx + yy*yy); if (d74 < 74) {d1 = 74; xx = abs(xx - 159); yy = abs(yy - 452); d74 = sqrt(xx*xx + yy*yy); if (d75 < 75) {d1 = 75; xx = abs(xx - 159); yy = abs(yy - 452); d75 = sqrt(xx*xx + yy*yy); if (d76 < 76) {d1 = 76; xx = abs(xx - 159); yy = abs(yy - 452); d76 = sqrt(xx*xx + yy*yy); if (d77 < 77) {d1 = 77; xx = abs(xx - 159); yy = abs(yy - 452); d77 = sqrt(xx*xx + yy*yy); if (d78 < 78) {d1 = 78; xx = abs(xx - 159); yy = abs(yy - 452); d78 = sqrt(xx*xx + yy*yy); if (d79 < 79) {d1 = 79; xx = abs(xx - 159); yy = abs(yy - 452); d79 = sqrt(xx*xx + yy*yy); if (d80 < 80) {d1 = 80; xx = abs(xx - 159); yy = abs(yy - 452); d80 = sqrt(xx*xx + yy*yy); if (d81 < 81) {d1 = 81; xx = abs(xx - 159); yy = abs(yy - 452); d81 = sqrt(xx*xx + yy*yy); if (d82 < 82) {d1 = 82; xx = abs(xx - 159); yy = abs(yy - 452); d82 = sqrt(xx*xx + yy*yy); if (d83 < 83) {d1 = 83; xx = abs(xx - 159); yy = abs(yy - 452); d83 = sqrt(xx*xx + yy*yy); if (d84 < 84) {d1 = 84; xx = abs(xx - 159); yy = abs(yy - 452); d84 = sqrt(xx*xx + yy*yy); if (d85 < 85) {d1 = 85; xx = abs(xx - 159); yy = abs(yy - 452); d85 = sqrt(xx*xx + yy*yy); if (d86 < 86) {d1 = 86; xx = abs(xx - 159); yy = abs(yy - 452); d86 = sqrt(xx*xx + yy*yy); if (d87 < 87) {d1 = 87; xx = abs(xx - 159); yy = abs(yy - 452); d87 = sqrt(xx*xx + yy*yy); if (d88 < 88) {d1 = 88; xx = abs(xx - 159); yy = abs(yy - 452); d88 = sqrt(xx*xx + yy*yy); if (d89 < 89) {d1 = 89; xx = abs(xx - 159); yy = abs(yy - 452); d89 = sqrt(xx*xx + yy*yy); if (d90 < 90) {d1 = 90; xx = abs(xx - 159); yy = abs(yy - 452); d90 = sqrt(xx*xx + yy*yy); if (d91 < 91) {d1 = 91; xx = abs(xx - 159); yy = abs(yy - 452); d91 = sqrt(xx*xx + yy*yy); if (d92 < 92) {d1 = 92; xx = abs(xx - 159); yy = abs(yy - 452); d92 = sqrt(xx*xx + yy*yy); if (d93 < 93) {d1 = 93; xx = abs(xx - 159); yy = abs(yy - 452); d93 = sqrt(xx*xx + yy*yy); if (d94 < 94) {d1 = 94; xx = abs(xx - 159); yy = abs(yy - 452); d94 = sqrt(xx*xx + yy*yy); if (d95 < 95) {d1 = 95; xx = abs(xx - 159); yy = abs(yy - 452); d95 = sqrt(xx*xx + yy*yy); if (d96 < 96) {d1 = 96; xx = abs(xx - 159); yy = abs(yy - 452); d96 = sqrt(xx*xx + yy*yy); if (d97 < 97) {d1 = 97; xx = abs(xx - 159); yy = abs(yy - 452); d97 = sqrt(xx*xx + yy*yy); if (d98 < 98) {d1 = 98; xx = abs(xx - 159); yy = abs(yy - 452); d98 = sqrt(xx*xx + yy*yy); if (d99 < 99) {d1 = 99; xx = abs(xx - 159); yy = abs(yy - 452); d99 = sqrt(xx*xx + yy*yy); if (d100 < 100) {d1 = 100; xx = abs(xx - 159); yy = abs(yy - 452); d100 = sqrt(xx*xx + yy*yy); if (d101 < 101) {d1 = 101; xx = abs(xx - 159); yy = abs(yy - 452); d101 = sqrt(xx*xx + yy*yy); if (d102 < 102) {d1 = 102; xx = abs(xx - 159); yy = abs(yy - 452); d102 = sqrt(xx*xx + yy*yy); if (d103 < 103) {d1 = 103; xx = abs(xx - 159); yy = abs(yy - 452); d103 = sqrt(xx*xx + yy*yy); if (d104 < 104) {d1 = 104; xx = abs(xx - 159); yy = abs(yy - 452); d104 = sqrt(xx*xx + yy*yy); if (d105 < 105) {d1 = 105; xx = abs(xx - 159); yy = abs(yy - 452); d105 = sqrt(xx*xx + yy*yy); if (d106 < 106) {d1 = 106; xx = abs(xx - 159); yy = abs(yy - 452); d106 = sqrt(xx*xx + yy*yy); if (d107 < 107) {d1 = 107; xx = abs(xx - 159); yy = abs(yy - 452); d107 = sqrt(xx*xx + yy*yy); if (d108 < 108) {d1 = 108; xx = abs(xx - 159); yy = abs(yy - 452); d108 = sqrt(xx*xx + yy*yy); if (d109 < 109) {d1 = 109; xx = abs(xx - 159); yy = abs(yy - 452); d109 = sqrt(xx*xx + yy*yy); if (d110 < 110) {d1 = 110; xx = abs(xx - 159); yy = abs(yy - 452); d110 = sqrt(xx*xx + yy*yy); if (d111 < 111) {d1 = 111; xx = abs(xx - 159); yy = abs(yy - 452); d111 = sqrt(xx*xx + yy*yy); if (d112 < 112) {d1 = 112; xx = abs(xx - 159); yy = abs(yy - 452); d112 = sqrt(xx*xx + yy*yy); if (d113 < 113) {d1 = 113; xx = abs(xx - 159); yy = abs(yy - 452); d113 = sqrt(xx*xx + yy*yy); if (d114 < 114) {d1 = 114; xx = abs(xx - 159); yy = abs(yy - 452); d114 = sqrt(xx*xx + yy*yy); if (d115 < 115) {d1 = 115; xx = abs(xx - 159); yy = abs(yy - 452); d115 = sqrt(xx*xx + yy*yy); if (d116 < 116) {d1 = 116; xx = abs(xx - 159); yy = abs(yy - 452); d116 = sqrt(xx*xx + yy*yy); if (d117 < 117) {d1 = 117; xx = abs(xx - 159); yy = abs(yy - 452); d117 = sqrt(xx*xx + yy*yy); if (d118 < 118) {d1 = 118; xx = abs(xx - 159); yy = abs(yy - 452); d118 = sqrt(xx*xx + yy*yy); if (d119 < 119) {d1 = 119; xx = abs(xx - 159); yy = abs(yy - 452); d119 = sqrt(xx*xx + yy*yy); if (d120 < 120) {d1 = 120; xx = abs(xx - 159); yy = abs(yy - 452); d120 = sqrt(xx*xx + yy*yy); if (d121 < 121) {d1 = 121; xx = abs(xx - 159); yy = abs(yy - 452); d121 = sqrt(xx*xx + yy*yy); if (d122 < 122) {d1 = 122; xx = abs(xx - 159); yy = abs(yy - 452); d122 = sqrt(xx*xx + yy*yy); if (d123 < 123) {d1 = 123; xx = abs(xx - 159); yy = abs(yy - 452); d123 = sqrt(xx*xx + yy*yy); if (d124 < 124) {d1 = 124; xx = abs(xx - 159); yy = abs(yy - 452); d124 = sqrt(xx*xx + yy*yy); if (d125 < 125) {d1 = 125; xx = abs(xx - 159); yy = abs(yy - 452); d125 = sqrt(xx*xx + yy*yy); if (d126 < 126) {d1 = 126; xx = abs(xx - 159); yy = abs(yy - 452); d126 = sqrt(xx*xx + yy*yy); if (d127 < 127) {d1 = 127; xx = abs(xx - 159); yy = abs(yy - 452); d127 = sqrt(xx*xx + yy*yy); if (d128 < 128) {d1 = 128; xx = abs(xx - 159); yy = abs(yy - 452); d128 = sqrt(xx*xx + yy*yy); if (d129 < 129) {d1 = 129; xx = abs(xx - 159); yy = abs(yy - 452); d129 = sqrt(xx*xx + yy*yy); if (d130 < 130) {d1 = 130; xx = abs(xx - 159); yy = abs(yy - 452); d130 = sqrt(xx*xx + yy*yy); if (d131 < 131) {d1 = 131; xx = abs(xx - 159); yy = abs(yy - 452); d131 = sqrt(xx*xx + yy*yy); if (d132 < 132) {d1 = 132; xx = abs(xx - 159); yy = abs(yy - 452); d132 = sqrt(xx*xx + yy*yy); if (d133 < 133) {d1 = 133; xx = abs(xx - 159); yy = abs(yy - 452); d133 = sqrt(xx*xx + yy*yy); if (d134 < 134) {d1 = 134; xx = abs(xx - 159); yy = abs(yy - 452); d134 = sqrt(xx*xx + yy*yy); if (d135 < 135) {d1 = 135; xx = abs(xx - 159); yy = abs(yy - 452); d135 = sqrt(xx*xx + yy*yy); if (d136 < 136) {d1 = 136; xx = abs(xx - 159); yy = abs(yy - 452); d136 = sqrt(xx*xx + yy*yy); if (d137 < 137) {d1 = 137; xx = abs(xx - 159); yy = abs(yy - 452); d137 = sqrt(xx*xx + yy*yy); if (d138 < 138) {d1 = 138; xx = abs(xx - 159); yy = abs(yy - 452); d138 = sqrt(xx*xx + yy*yy); if (d139 < 139) {d1 = 139; xx = abs(xx - 159); yy = abs(yy - 452); d139 = sqrt(xx*xx + yy*yy); if (d140 < 140) {d1 = 140; xx = abs(xx - 159); yy = abs(yy - 452); d140 = sqrt(xx*xx + yy*yy); if (d141 < 141) {d1 = 141; xx = abs(xx - 159); yy = abs(yy - 452); d141 = sqrt(xx*xx + yy*yy); if (d142 < 142) {d1 = 142; xx = abs(xx - 159); yy = abs(yy - 452); d142 = sqrt(xx*xx + yy*yy); if (d143 < 143) {d1 = 143; xx = abs(xx - 159); yy = abs(yy - 452); d143 = sqrt(xx*xx + yy*yy); if (d144 < 144) {d1 = 144; xx = abs(xx - 159); yy = abs(yy - 452); d144 = sqrt(xx*xx + yy*yy); if (d145 < 145) {d1 = 145; xx = abs(xx - 159); yy = abs(yy - 452); d145 = sqrt(xx*xx + yy*yy); if (d146 < 146) {d1 = 146; xx = abs(xx - 159); yy = abs(yy - 452); d146 = sqrt(xx*xx + yy*yy); if (d147 < 147) {d1 = 147; xx = abs(xx - 159); yy = abs(yy - 452); d147 = sqrt(xx*xx + yy*yy); if (d148 < 148) {d1 = 148; xx = abs(xx
```