

Paper Computer Science Experiment

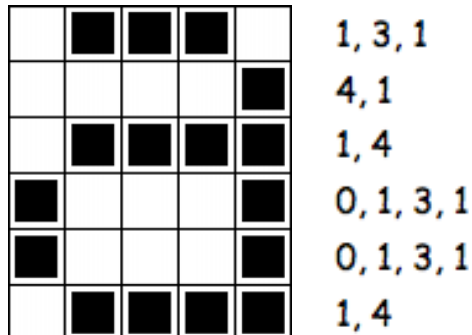
4

<p>Great Principles</p>  <p>of Computing</p>	Computation (Informationsspeicherung)
 <p>Thema</p>	Digitale Repräsentation von Grafiken und Bildern
 <p>Unterrichtsform</p>	Einzel- und Partnerarbeit
 <p>Voraussetzung</p>	In einem Zeichenprogramm, einem Videogame oder einer Multimediaanwendung müssen Computer Bilder speichern können. Wie können Computer diese Grafiken speichern, wenn sie doch nur Zahlen kennen? Wie machen das Faxgeräte?
 <p>Material</p>	<ul style="list-style-type: none">• Kopiervorlagen
 <p>Zeitdauer</p>	Erklärungen durch Lehrkraft plus Einzel- und Partnerarbeit 10 – 15 Min.



Vorgehen

Computermonitore sind in ein feines Raster gegliedert, deren Punkte man Pixel nennt ("picture elements", auf deutsch Bildpunkte). Bei einem schwarz-weiss Bild (wie sie Faxe verwenden) ist jedes Pixel entweder schwarz oder weiss. Der Buchstabe "a" wurde vergrössert, um die Pixelstruktur deutlich zu machen. Wenn ein Computer ein schwarz-weiss Bild speichert, muss er nur wissen, welche Punkte schwarz, und welche weiss sind.



Ein Bild wird mithilfe von Zahlen dargestellt. Die erste Zeile besteht aus einem weissen, drei schwarzen und wieder einem weissen Pixel. Die erste Zeile wird folglich als 1, 3, 1 gespeichert.

Quelle: http://csunplugged.org/sites/default/files/activity_pdfs_full/unplugged-02-image_representation.pdf

Die allererste Zahl gibt immer die Anzahl weisser Pixel an. Beginnt die Zeile mit einem schwarzen Bildpunkt, muss die Zeile mit einer Null beginnen. Sind zudem die Dimensionen der Grafik bekannt (im obigen Beispiel 5 x 6 Pixel), so muss der Zeilenumbruch nicht kodiert werden und das oben stehende Bild kann folgendermassen digitalisiert werden:

1, 3, 5, 1, 1, 5, 3, 2, 3, 1, 1, 4

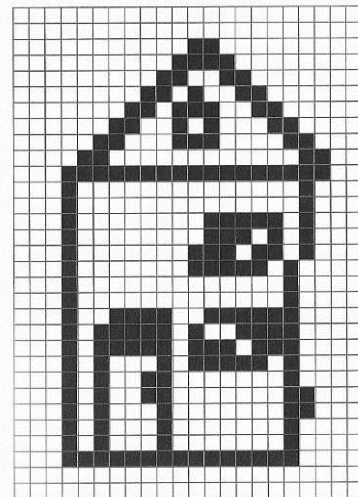
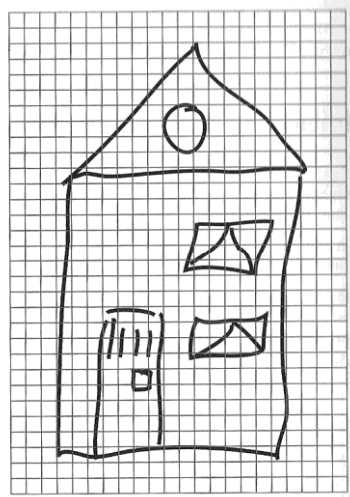
Diese Darstellung nennt man „Laufängen“ (engl. „run-length“). Bei der Laufängencodierung werden nicht einzelne Bildpunkte gespeichert oder übermittelt, sondern die Anzahl gleicher Bildpunkte hintereinander. Eine Laufängencodierung wird in vielen bekannten Bild- und Dokumentformaten (z.B. BMP, GIF, TIFF, PDF und Postscript) auf dem Computer standardmässig eingesetzt, um Speicherplatz zu sparen.

Die Kopiervorlage enthält mehrere Bilder, welche die Schüler/innen mit der eben gezeigten Methode dekodieren können.



Varianten

Auf einem Häuschenblatt lassen sich einfach eigene Bilder bzw. Faxe digitalisieren und dekodieren.



Quelle: Buch von Jens Gallenbacher: Abenteuer Informatik – IT zum Anfassen von Routenplaner bis Online-Banking, Spektrum Akademischer Verlag, 2007

Werden bei der linken Skizze des „Traumhauses“ alle Rasterpunkte (Häuschen) als schwarz definiert, falls eine Linie durchgeht, so entsteht das rechte Rasterbild, bei welchem einige Details verloren gegangen sind. Beim Fax Standard z.B. sind pro Millimeter 8 Rasterpunkte definiert.

Die vollständige Zahlenreihe der Lauflängencodierung ist:

55-2-19-4-17-2-2-2-15-2-1-1-2-2-13-2-1-3-2-2-11-2-2-1-1-1-3-2-9-2-3-3-4-2-7-2-12-2-5-17-5-1-14-1-6-1-14-1-6-1-8-5-1-1-6-1-8-3-1-1-1-6-1-7-3-1-2-1-1-6-1-7-5-1-1-7-1-13-1-7-1-13-1-7-1-2-4-2-6-7-1-1-5-1-1-2-4-7-1-1-5-1-3-1-3-7-1-1-1-3-1-1-5-1-1-7-1-1-1-2-2-7-1-7-1-1-1-2-2-7-2-6-1-1-1-3-1-7-2-6-1-1-1-3-1-7-1-7-1-1-1-3-1-7-1-7-15-48


Quelle: Buch von Jens Gallenbacher: Abenteuer Informatik – IT zum Anfassen von Routenplaner bis Online-Banking, Spektrum Akademischer Verlag, 2007

Anmerkung:

Diese Zahlen können nun übermittelt werden. Eine Zahl zwischen 0 und 63 benötigt 6 Bit. Insgesamt sind für dieses Faxdokument 159 Zahlen zu übermitteln, also 954 Bit. Obwohl diese Zahlen übersichtlich und eine Dekodierung einfach wäre, haben wir allerdings nichts gespart:

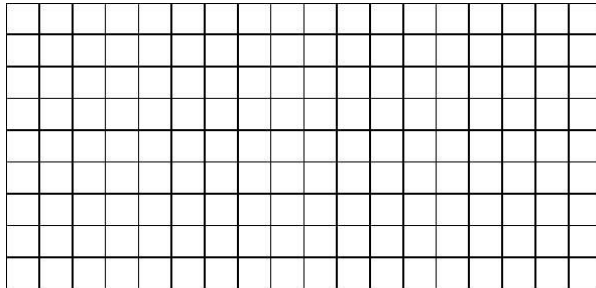
Wenn man die Bildpunkte mit je einem Bit übermittelt hätte (0 für „weiss“ und 1 für „schwarz“), hätten wir nur 31 Zeilen x 22 Spalten x 1 Bit = 682 Bit benötigt.

Diese Erkenntnis führt zur Entwicklung von Codebäumen mit variabler Codelänge (vgl. Morsealphabet) und zum Huffman-Algorithmus.

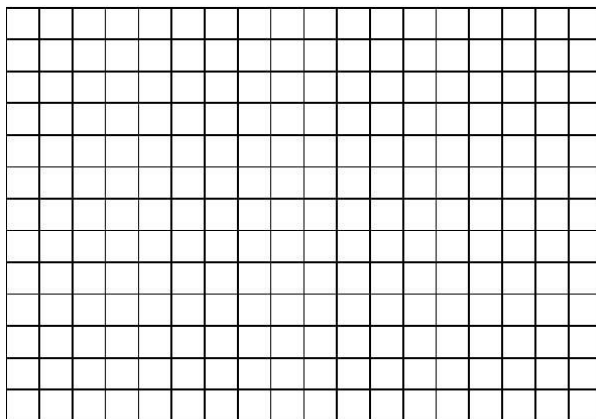
	<p>Folgende Lerneinheiten sind im Anschluss an diese Aktivität zu empfehlen: Abenteuer Informatik (Jens Gallenbacher) Kap. 5 „Von Kamelen und dem Nadelöhr“, S. 120 – 138</p>
 <p>Weitere Ideen</p>	<p>Informatik erLeben: Diverse Arbeitsblätter zur Codierung http://informatik-erleben.uni-klu.ac.at/einheiten/c/</p> <p>Huffman-Algorithmus und Datenkomprimierung http://www.swisseduc.ch/informatik/daten/huffmann_kompression/index.html</p> <p>Computer Science Unplugged: Colour by Number http://csunplugged.org/image-representation</p> <p>Computer Science Unplugged: Text Compression http://csunplugged.com/text-compression</p> <p>Abenteuer Informatik: Kompression http://www.abenteuer-informatik.de/PDF/kompression_a644.pdf</p> <p>In der folgenden Aktivität werden die Bilder computergerecht nicht im Zehner-, sondern im Binärsystem und als Hexadezimal-Code digitalisiert. http://cse4k12.org/binary/bitmaps.html</p> <p>Eine Möglichkeit ist auch, ein Bild mit einem Text format zu beschreiben. Diese Technik hat den Namen SVG (Scalable Vector Graphics). In der folgenden Aktivität werden Grafiken mit XML-Dateien erstellt (Abschnitt 6 „Ran ans Bild“) http://www.abenteuer-informatik.de/PDF/codierung2_a802.pdf</p>

Faxen machen

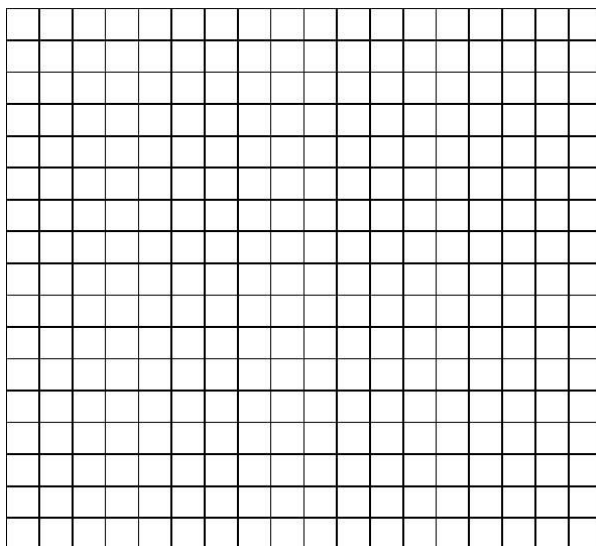
Das erste Bild ist das einfachste, das letzte das aufwändigste. Da es leicht ist, Fehler zu machen, ist die Verwendung von Bleistift ratsam. Ein Radiergummi ist hilfreich.



4, 11 4, 9,
2, 1 4, 9,
2, 1 4, 11
4, 9 4, 9
5, 7 0, 17
1, 15



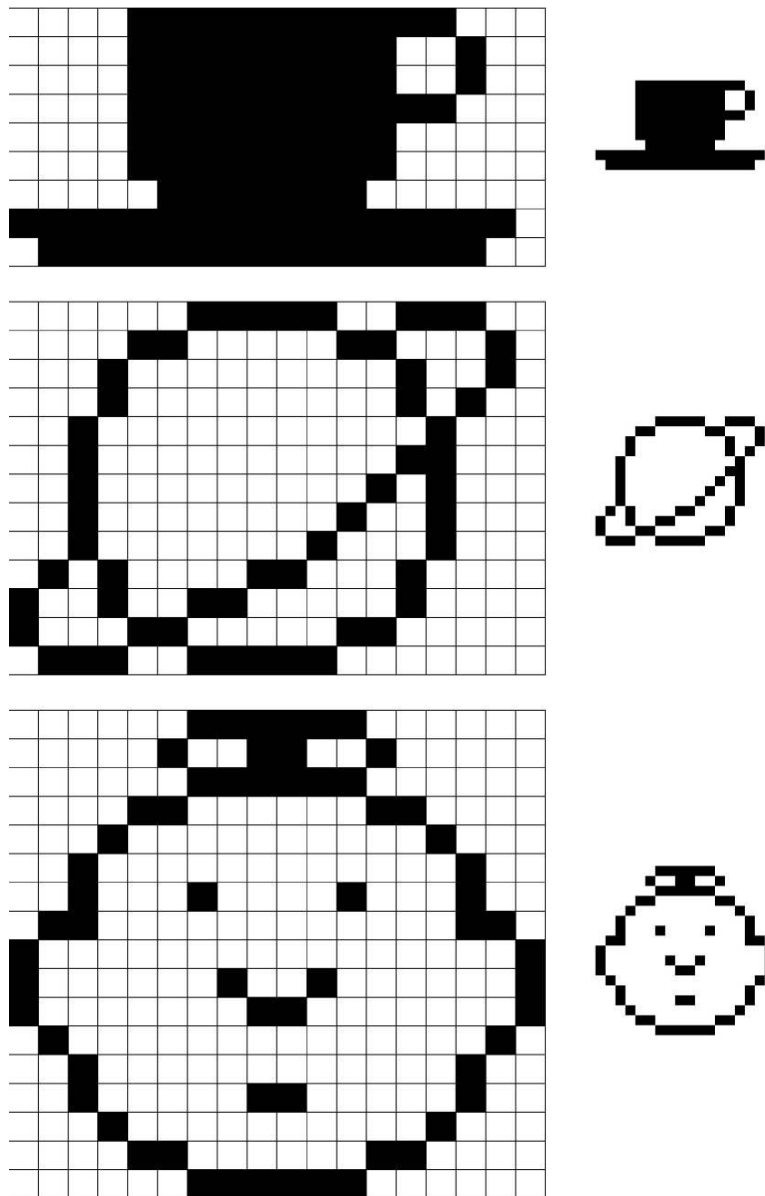
6, 5, 2, 3 4, 2, 5,
2, 3, 1 3, 1, 9, 1, 2,
1 3, 1, 9, 1, 1, 1 2,
1, 11, 1 2, 1, 10, 2
2, 1, 9, 1, 1, 1 2, 1,
8, 1, 2, 1 2, 1, 7, 1,
3, 1 1, 1, 1, 1, 4, 2,
3, 1 0, 1, 2, 1, 2, 2,
5, 1 0, 1, 3, 2, 5, 2
1, 3, 2, 5



6, 2, 2, 2 5, 1, 2,
2, 2, 1 6, 6 4, 2, 6,
2 3, 1, 10, 1 2, 1,
12, 1 2, 1, 3, 1, 4,
1, 3, 1 1, 2, 12, 2
0, 1, 16, 1 0, 1, 6,
1, 2, 1, 6, 1 0, 1, 7,
2, 7, 1 1, 1, 14, 1 2,
1, 12, 1 2, 1, 5, 2,
5, 1 3, 1, 10, 1 4,
2, 6, 2 6, 6

Quelle: http://csunplugged.org/sites/default/files/activity_pdfs_full/unplugged-02-image_representation.pdf

Auflösung zum Arbeitsblatt Faxen machen



Quelle: http://csunplugged.org/sites/default/files/activity_pdfs_full/unplugged-02-image_representation.pdf