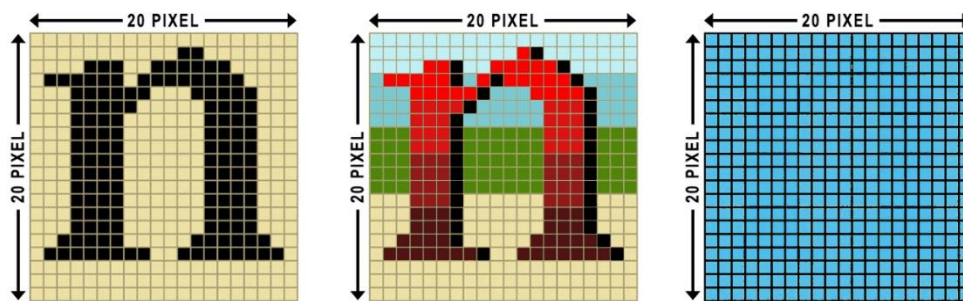




B_DATEN_KOMPRIMIEREN_ANNEX

Aufgaben

1. **Huffman-Algorithmus:** Wir beschäftigen uns hier mit sogenannten Baumstrukturen. Kennen sie noch andere Gebiete in der IT, wo Baumstrukturen zur Anwendung kommen? Beim Huffman handelt es sich sogar um eine spezielle Baumstruktur, nämlich einem sogenannten binären Baum. Was unterscheidet einen binären Baum von einem nicht binären Baum?
2. **Huffman-Algorithmus:** In dieser Aufgabe arbeiten sie zu zweit: Jeder denkt sich ein Wort mit ca. 15 Buchstaben aus und erstellt dazu den Huffman-Code inkl. Codetabelle und das entsprechend komprimierte Wort in Binärdarstellung. Tauschen sie ihre Codes und Codetabellen gegenseitig aus und vergewissern sie sich, dass ihr Partner ihr gewähltes Wort richtig dekomprimieren kann. Sie haben die Aufgabe dann vollständig gelöst, wenn sie einen korrekten binären Baum vorweisen können, die Codes herausgelesen und tabellarisch notiert haben und das komprimierte Wort in Huffman-Binärcode nicht fehlt.
3. **RLC:** Wie könnte die Komprimierung ausschauen, wenn es sich anstatt um ein Schwarz/Weissbild, um ein Farbbild handelt?



Benachbarte Pixel mit identischer Farbe werden ja bei RLC bekanntlich nicht einzeln genannt, sondern zu einer Anzahl zusammengefasst wie z.B. 11xGrün, 6xBlau, 3xWeiss etc. oder in binärer Schreibweise 1011Grün, 0110Blau, 0011Weiss. Welche Bitbreite (1011Grün ergäbe 4 Bit) wäre bei einem quadratischen Bild mit 20 Pixel Kantenlänge sinnvoll? Was wäre, wenn dieses Bild nur aus einer Farbe besteht?

4. **RLC:** Sie erhalten diesen RL-Code:
010100011110010010010010010010010010010110010110010010010010010
010010010001
Folgendes ist ihnen dazu bekannt: Es handelt sich um eine quadratische Schwarz-Weiss-Rastergrafik mit einer Kantenlänge von 8 Pixel. Es wird links oben mit der Farbe Weiss begonnen. Eine Farbe kann sich nicht mehr als siebenmal wiederholen. Zeichnen sie die Grafik auf. Was stellt sie dar?



5. **LZW-Verfahren**

- a. Erstellen sie die LZW-Codierung für das Wort «ANANAS» und überprüfen sie mit der Dekodierung ihr Resultat.
- b. Versuchen sie den erhaltenen LZW-Code «ERDBE<256>KL<260>» zu dekomprimieren.

6. **BWT (Burrows-Wheeler-Transformation):**

- a. Erstellen sie die BWT-Transformation für das Wort ANANAS und überprüfen sie mit der Rücktransformation ihr Resultat.
- b. Sie erhalten den Code IICRTGH6 in der Burrows-Wheeler-Transformation. Welches Wort verbirgt sich dahinter?

6. **ZIP-Komprimierung:** Wir wollen die Effizienz bei der ZIP-Komprimierung untersuchen. Dazu sollen sie ASCII-Textdateien erstellen.

- a. Die erste enthält 10, die zweite 100, die dritte 1000, die vierte 10'000 und die fünfte 100'000 ASCII-Zeichen.
- b. Achten sie darauf, dass die Zeichen möglichst zufällig gewählt werden. Auf dem Internet findet man entsprechende Textgeneratoren.
- c. Kopieren sie jede dieser fünf Textdateien in eine eigene ZIP-Datei. In der Folge erhalten sie fünf ZIP-Dateien.
- d. Werten sie nun in einer EXCEL-Tabelle die erforderlichen Speichergrößen aus: ASCII-Datei-Grösse zu ZIP-Datei-Grösse. Versuchen sie nun, ihr Resultat zu interpretieren bzw. zu begründen. Tipp: Sie können in EXCEL Zahlenreihen auch grafisch anzeigen.
- e. Nun legen wir noch einen drauf: Erstellen sie eine ASCII-Textdatei mit 100'000 Zeichen. Diesmal aber nicht zufällig (random) befüllt, sondern ausschliesslich mit dem Buchstaben A, danach zippen sie. Vergleichen sie nun die beiden ZIP-Dateien. Wie erklären sie sich den Unterschied der Speichergrößen?
- f. Zu guter Letzt wollen wir untersuchen, was die ZIP-Komprimierung bringt, wenn die Originaldatei, wie beim JPG-Bildformat, bereits komprimiert (DCT) vorliegt. Dazu erhalten sie die zwei folgenden Bilder:
<https://www.juergarnold.ch/Kompression/ZIPTestHi.jpg>
<https://www.juergarnold.ch/Kompression/ZIPTestLo.jpg>
Gehen sie nun gleich vor, wie beim vorangegangenen Untersuch der Textdateien. Begründen sie ihr Resultat.

7. Kenne sie noch weitere Verfahren, wo verlustlos komprimiert wird?
Welche Daten sollen überhaupt verlustlos komprimiert werden? Was würde passieren, wenn man ein Brief oder ein Java-Sourcecode verlustbehaftet komprimieren würde?