

SWE2 Labor

Lauf­längen­kodierung (Run Length Encoding)

Aufgabe 1 (Run Length En-/Decoder)

Ein Grossteil der Daten mit denen wir taeglich arbeiten, enthalten viel redundante Information. Ein Beispiel sind Bilder mit grossen einfarbigen Flaechen (z.B. Logos) oder Tabellen und Log-Dateien mit immer wiederkehrenden Eintraegen.

Diese Daten sind bereits mit verlustfreien Methoden relativ gut komprimierbar. Die erreichbare Kompressionsrate fuer einen bestimmten Algorithmus haengt dabei stark von den Daten ab.

Ein simples Schema zur Kompression ist die Lauflaengenkodierung. Hierbei werden haeufig hintereinander auftretende Zeichen durch einen Zaehler und das Zeichen selbst repraesentiert.

Die Zeichenfolge `AAAAAABBBBBBCDEFGG` wird dann zu `A6B6C1D1E1F1G2`.

Schreiben Sie ein Kommandozeilen-Programm, welchem als Parameter der gewuenschte Modus (De-/Kodierung), die Eingabe-Datei und die Ausgabe-Datei uebergeben werden. Der Aufruf erfolgt also so:

```
./rle (encode|decode) eingabe_datei ausgabe_datei
```

Anbei finden Sie zwei Test-Dateien:

1) `aabbcc.txt` Eine einfache Textdatei, welche das Alphabet enhaelt, bei dem einige Buchstaben dupliziert wurden.

2) `fhs_logo_bw.png` Unkomprimierte Schwarz-Weiss Version des FH-Salzburg Logos. Enthaelt viele weisse Bereiche, getrennt von kleinen schwarzen Bereichen.

Tipp: verwenden Sie einen Hex-Editor um den Inhalt der Datei darzustellen (z.B. dhex, ghex oder direkt als VSCode Plugin)

Schreiben Sie intern dazu zwei Funktionen:

```
void encode(FILE* fin, FILE* fout);
```

Diese Funktion nimmt den input stream `fin` entgegen, und wandelt eine Folge von 1..255 Byte in eine Folge aus Wertepaaren (`value`, `count`) und schreibt diese in den stream `fout`.

```
void decode(FILE* fin, FILE* fout);
```

Diese Funktion erwartet die mit der Funktion `encode()` erzeugten Wertepaare im stream `fin` und dekodiert diese Wertepaare (`value`, `count`) in eine Folge von gleichen Zeichen '`value`' mit der Laenge '`count`'. Das ergebnis wird in den stream `fout` geschrieben.

Und fangen Sie fehlerhafte Argumente und deren Kombinationen ab.

Das Programm soll die Dateien Byte-weise verarbeiten. Achten Sie ebenfalls darauf, dass Ihr Programm auch Zeichenfolgen laenger als 255 Bytes verarbeiten kann.