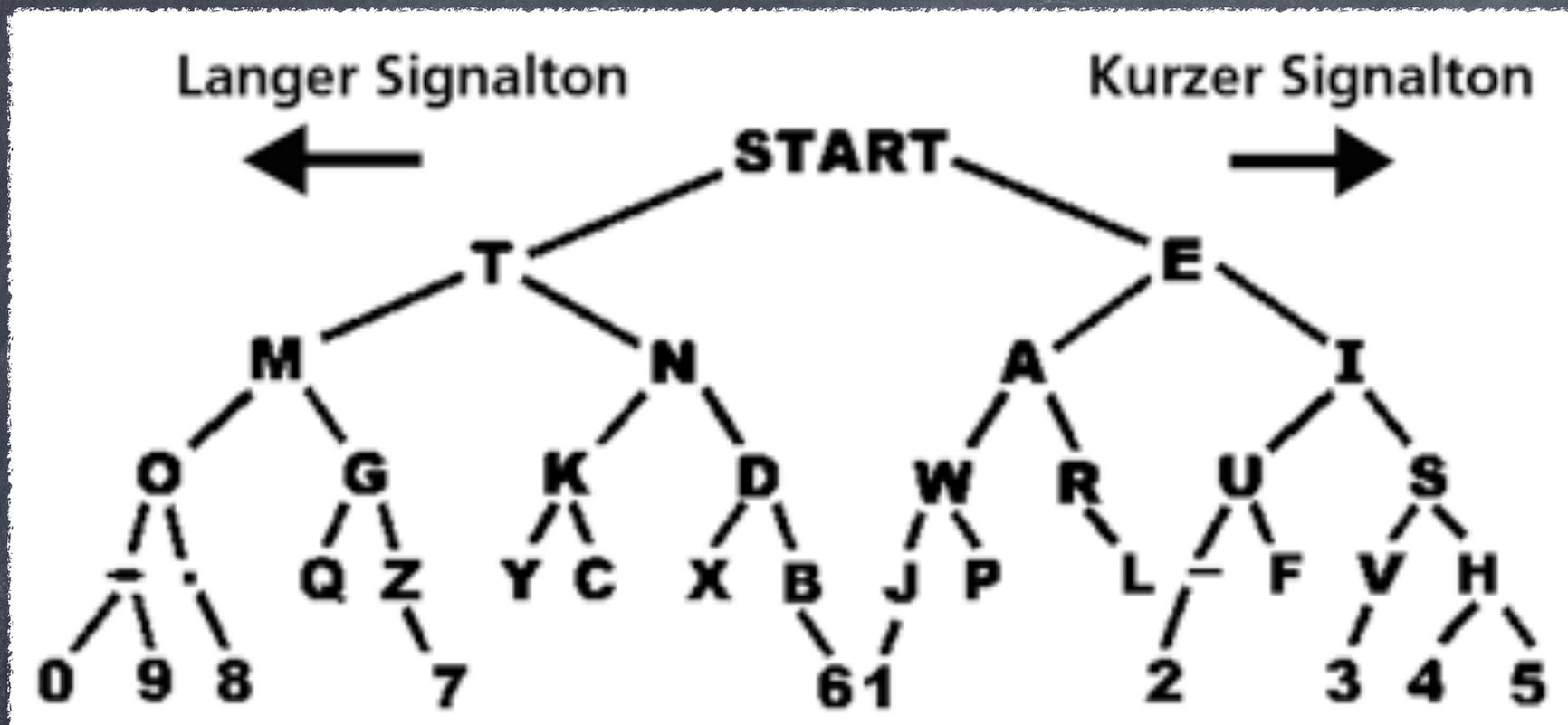




Komprimieren (#1)

Verlustlos oder verlustbehaftet?



Verlustlos: Morsecode

Was bedeutet: 00-0000- ?

- ◉ Problematisch bei Codes (wie dem Morsecode) mit unterschiedlicher Codewortlänge:
Wo beginnt das Wort, wo endet es!
- ◉ Beim Morsecode wird das Problem mit kurzen Pausen hinter jedem Codewort gelöst.
- ◉ Feststellung:
Wenn kein Codewort Anfangswort eines anderen Codewortes ist, dann wäre jede codierte Zeichenreihe eindeutig!
- ◉ Diese Bedingung erfüllt der **Huffman-Algorithmus**! Wie dieser funktioniert nun im Detail:

E R D B E E R E

Huffman N° 1

Das Wort ERDBEERE soll codiert werden...

E	R	D	B	E	E	R	E
1	2	3	4	5	6	7	8

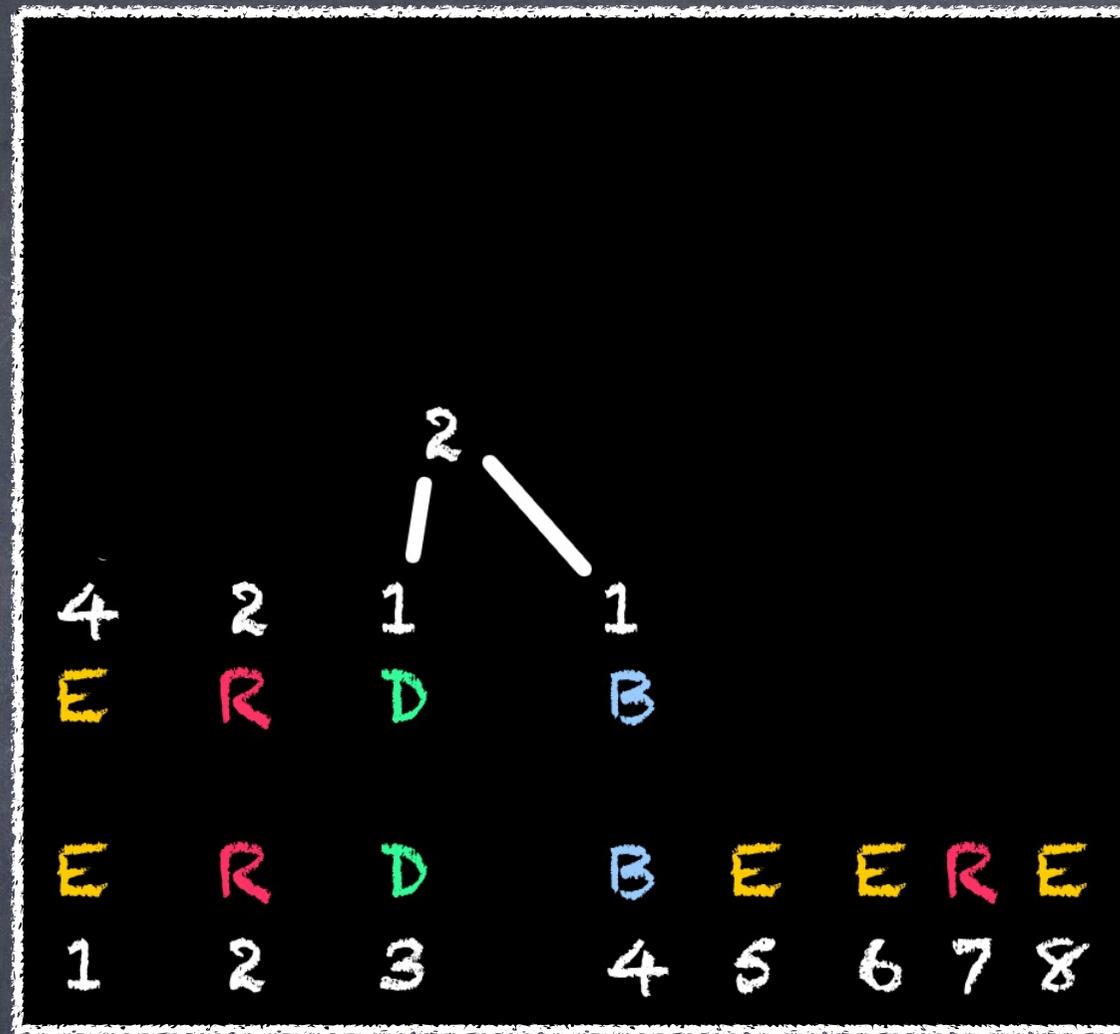
Huffman N° 2

Das Wort ERDBEERE zählt 8 Buchstaben...

4	2	1	1					
E	R	D	B					
E	R	D	B	E	E	R	E	
1	2	3	4	5	6	7	8	

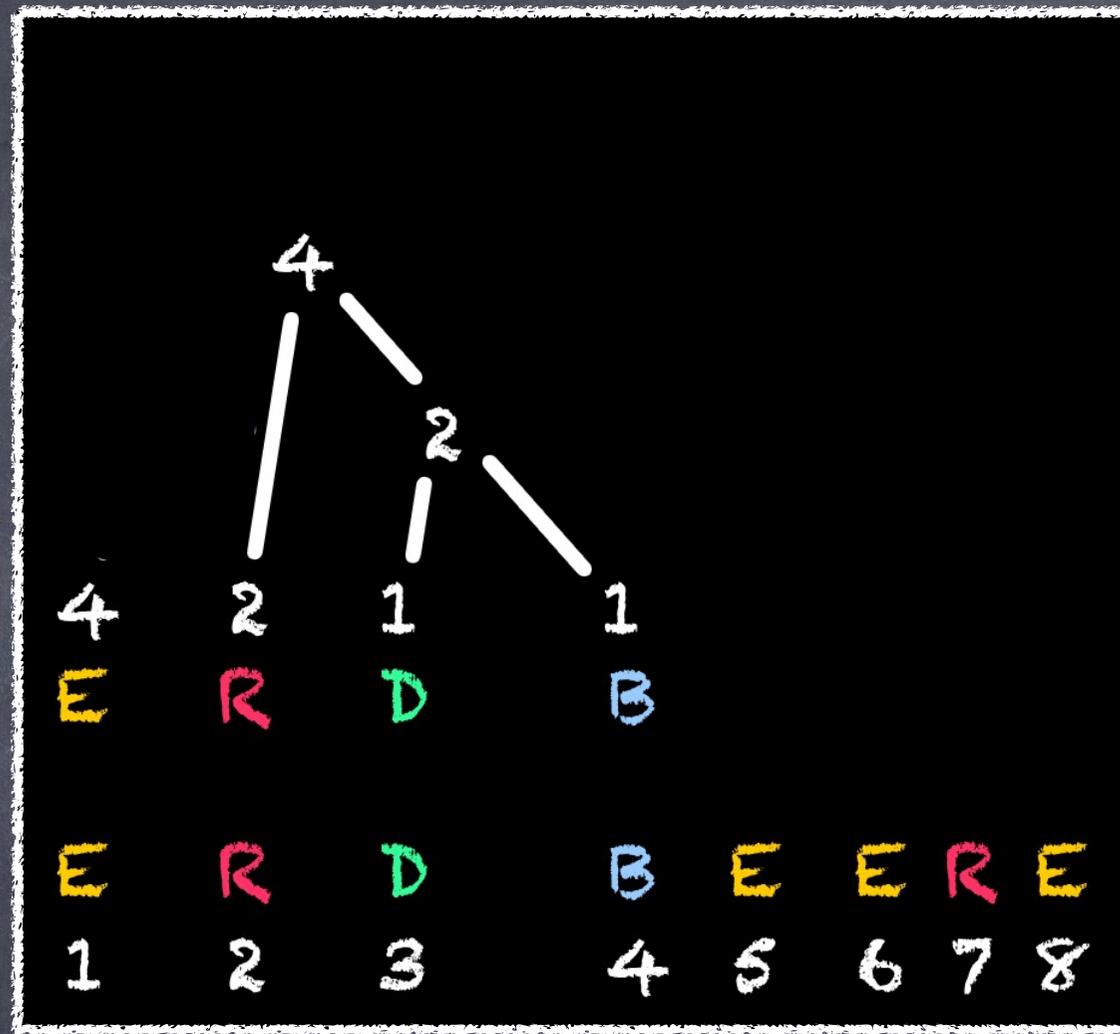
Huffman N° 3

Und das ist die Häufigkeit der Buchstaben...



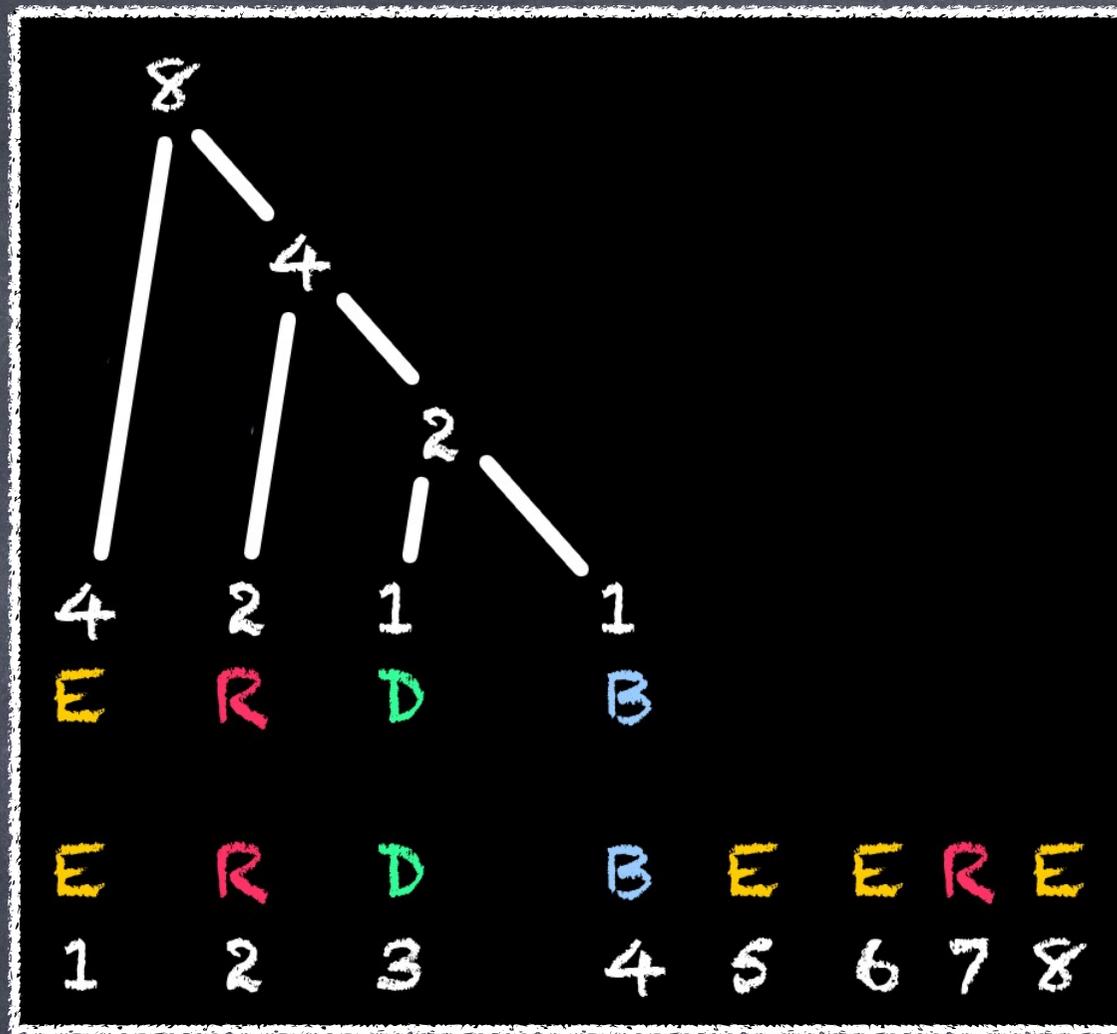
Huffman N° 4

Wir erstellen einen binären Baum!
 Die am wenigsten häufigen Buchstaben werden zusammengefasst...



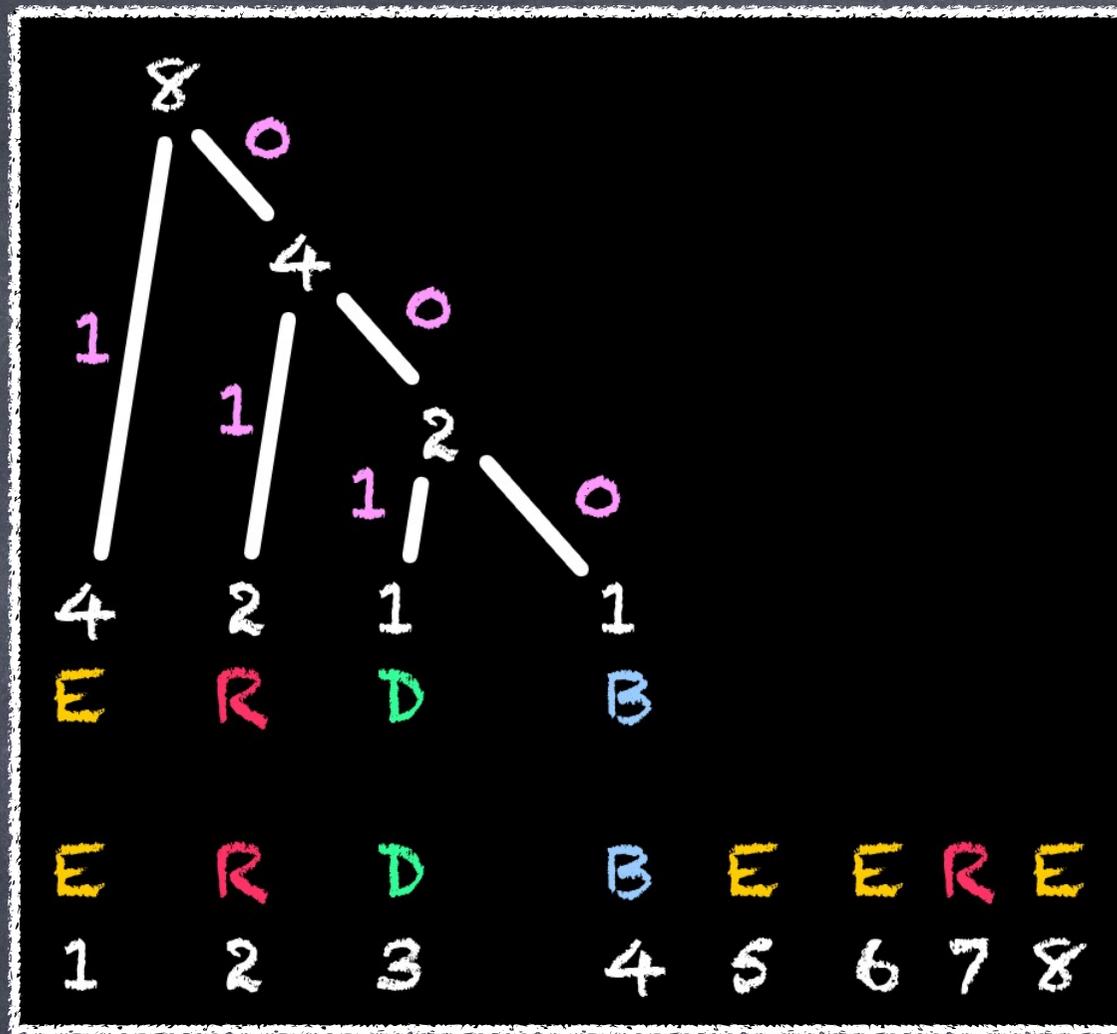
Huffman N° 5

Und so geht das entsprechend weiter...



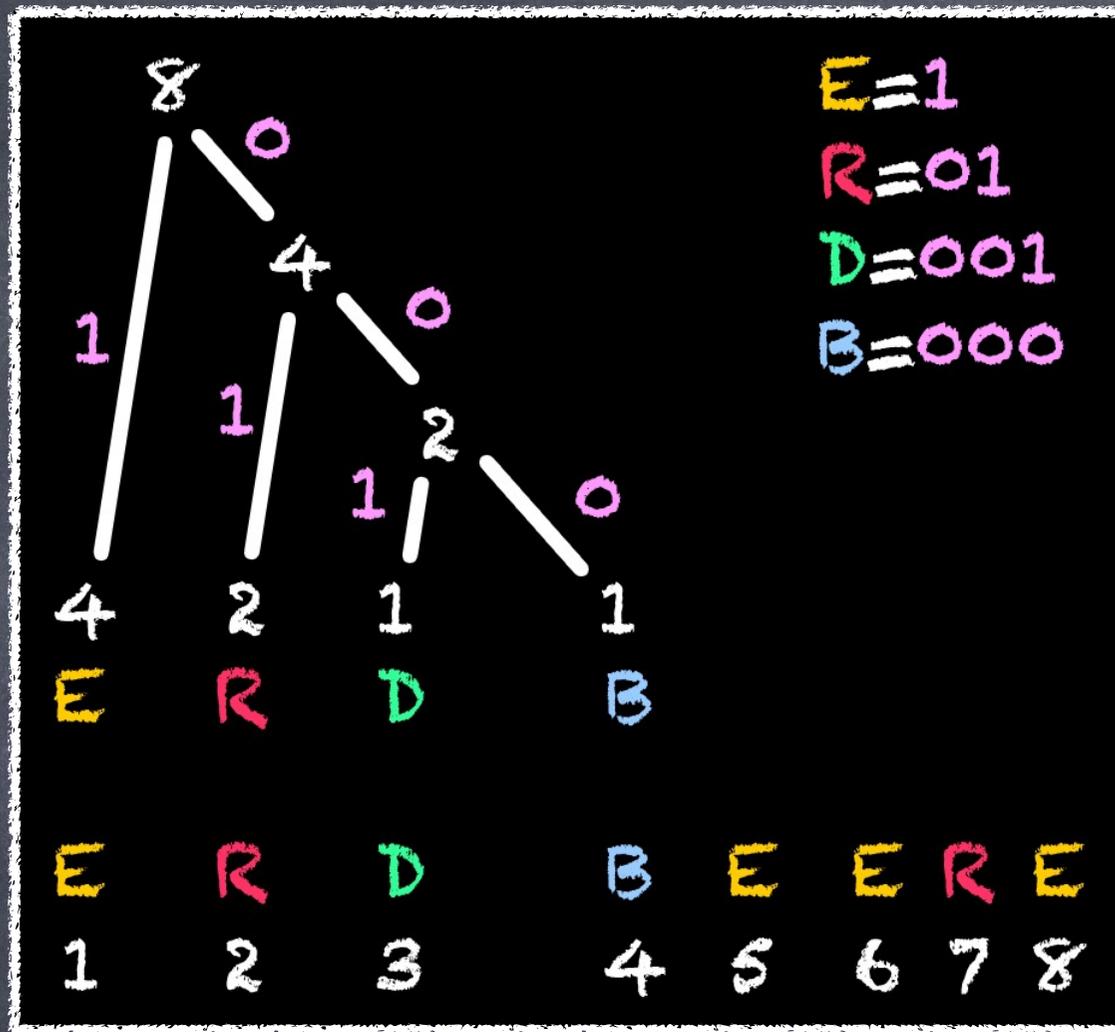
Huffman N° 6

...bis zuoberst! Die Zahl 8 muss der Anzahl Buchstaben in ERDBEERE entsprechen!



Huffman N° 7

Nun noch die Äste des Baums beschriften...



Huffman N° 8

...und die Codes auslesen!

E=1

R=01

D=001

B=000

1 01 001 000 1 1 01 1

E R D B E E R E

1 2 3 4 5 6 7 8

Huffman N° 9

Und das ist der Huffman-Code für ERDBEERE!
Wesentlich effizienter als mit 8 Stück 8-bit-ASCII-Zeichen! Oder?

Und nun sie!

Erstellen sie den Huffman-Code für die Textzeile:

GREIFENSEE SCHIFFFAHRT

Beachten sie, dass das Leerzeichen zwischen Greifensee und Schifffahrt auch ein zu codierendes Zeichen ist!
Geben sie die Codeeffizienz gegenüber einer ASCII-Codierung an!