

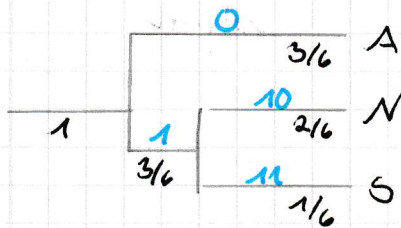
Wort: ANANAS

RLE: 1A1N1A1N1A1S

→ schlimmer als vorher

Huffman: A | N | S

→ N+S

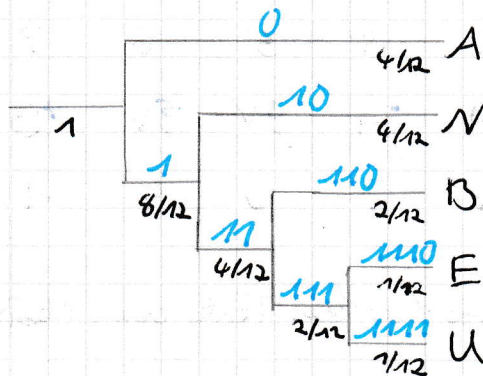


Symbol	Code
A	0
N	10
S	11

Wort: BANANENANBAU

B | A | N | E | U

→ E+U → EU+B → EUB+N



A → 0
N → 10
B → 110
E → 1110
U → 1111

Burrows - Wheel:

1) Aufstellen der Permutationen

A	N	A	N	A	S	\$
N	A	N	A	S	\$	A
A	N	A	S	\$	A	N
N	A	S	\$	A	N	A
A	S	\$	A	N	A	N
S	\$	A	N	A	N	A
\$	A	N	A	N	A	S

2) Zeilen alphabetisch sortieren

\$	A	N	A	N	A	S
A	N	A	N	A	S	\$
A	N	A	S	\$	A	N
A	S	\$	A	N	A	N
N	A	N	A	S	\$	A
N	A	S	\$	A	N	A
S	\$	A	N	A	N	A

3) letzte Spalte
als Output
nehmen

Codierter Text	S	\$	N	N	A	A	A
Position	1	2	3	4	5	6	7
x sortierter Text	1\$	2A	4A	6A	3N	5N	7S
sort. Position	2	5	6	7	3	4	1

Rücktransformation nur mit sortiertem Text und Position:
→ ANANAS

x RLE: 3A2N1S → effektiver, je größer Eingabe

Python:

letters = list(set("ANANAS")) → [A, N, S]

liste.A.sort(key = myFunction)

z.B. liste.A.sort(key = lambda h: h, height)