Compression

- Compression
- Lempel-Ziv
- Re-Pair and Grammars

- Encoding and decoding.
- Lossless and lossy
- Compressed computation.

- Statistical compression.
  - Huffman, arithmetic encoding, Burrows-Wheeler, PPM, ...
- Dictionary compression.
  - Lempel-Ziv 77, Lempel-Ziv 78, Lempel-Ziv-Welch, ...
- Grammar based schemes.
  - Re-Pair, sequitur, greedy, bisection, ...
- Kolmogorov compression.
  - Ultimate compression scheme.

- Transformation techniques.
  - Differencing, Burrows-Wheeler, run-length encoding, Fourier transform, ...
Compression
• Compression
• Lempel-Ziv
• Re-Pair and Grammars

Lempel-Ziv 77
• Encoding.
  • Parse from left-to-right into phrases.
  • Select longest substring starting before current position + 1 character.
  • Encode phrases by (previous occ, length, single character) or single character.

Lempel-Ziv 77
• Decoding. Read and decode left-to-right.
• Time. $O(n)$
Lempel-Ziv 78

- Encoding.
  - Parse from left-to-right into phrases.
  - Select longest phrase seen before + a single character.
  - Encode phrases (previous phrase, character) or

![Lempel-Ziv 78 Diagram](image)

- Decoding. Read and decode left-to-right.
  - Time. O(n)

![Lempel-Ziv 78 Diagram](image)

Compression

- Compression
  - Lempel-Ziv
  - Re-Pair and Grammars
Re-Pair Compression

- Recursive-pairing compression [Larsson and Moffat 2000].
- Start with string S.
- Replace a most frequent pair ab by new character X_i. Output rule X_i ➞ ab.
- Repeat until we have a single pair.
- Decoding. Unfold rules top-down.

```
x_9
x_8x_6 x_5 ➞ x_8x_6
x_3x_6 x_5 ➞ x_3x_7
x_3x_6 x_5 ➞ x_4x_5
x_3x_4x_3x_2 x_6 ➞ x_3x_5
x_3x_4acX_2 x_5 ➞ ac
x_3X_1acX_2 x_4 ➞ X_1X_1
X_2x_3x_1acX_2 x_3 ➞ X_2x_2
X_1cX_1cX_1acX_1c x_2 ➞ X_1c
abcabcabacababc x_1 ➞ ab
```

Grammar Compression

- Grammar compression. Encode string S as an grammar G that generates S.
- Parse tree. Unfolded set of rules.

```
x_{12} ➞ X_1X_9 x_6 ➞ X_6x_9
x_{11} ➞ X_1X_{10} x_5 ➞ X_5x_3
x_{10} ➞ X_4x_8 x_2 ➞ X_2X_2
x_9 ➞ X_9x_5 x_3 ➞ c
x_8 ➞ X_1x_3 x_2 ➞ b
x_1 ➞ a
```

```
```
```