



圧縮

- 符号化などの方法を工夫して、通常よりも少ないデータ量(ビット数)で同等の情報を表現する手法
- 可逆圧縮
 - 圧縮したものから元の情報を完全に復元できる方法
 - 情報量は変わらないが、見かけのデータ量が減る
 - 例: PNG, GIF, ZIP
- 非可逆圧縮
 - 圧縮したものからもとの情報を復元できない方法
 - 情報量を若干減らすことで、見かけのデータ量を大きく減らす
 - 例: JPEG, MP3, MPEG(人間の感じ取りにくい部分のデータを間引くことによって圧縮するため、間引いた情報は復元できない)

圧縮の原理 1

ランレングス符号化

- ビット列を0または1の繰り返し回数で表現する
 - 元データ: 00011000000011111110000011111000
 - 繰り返し回数は左から0が3回, 1が2回, 0が7回, 1が7回, 0が5回, 1が5回, 0が3回となる
 - 0と1は交互に現れるので, 「0が」とか「1が」は書かなくてもわかる
 - 繰り返しの回数を3ビットずつ書き並べると, 011 010 111 111 101 101 011になる
 - 元データの32ビットが21ビットに圧縮できた

圧縮の原理 2

ハフマン符号化

- 出現確率の大きなものには短いビット列を割り当てる
 - 一週間の天気(晴れ, 雨, 雪, くもり)を伝達する場合, 4種類を区別するためには2ビットが必要
 - 1週間の天気: 晴れ, 晴れ, 晴れ, くもり, くもり, くもり, 雨
 - 全てを2ビットで符号化すると, 00 00 00 11 11 11 10の14ビット
 - 出現確率が 晴れ > くもり > 雨 > 雪 なら, 晴れ:1 くもり:01 雨:001 雪:000 と符号化すると, 1 1 1 01 01 01 001の12ビット

天気	2ビットで符号化	出現確率によって符号長を変更
晴れ	00	1
雨	01	001
雪	10	000
くもり	11	01