

Algorytmy kompresji

Algorytm Run Length Encoding

Algorytm RLE (Run Length Encoding) jest jednym z najprostszych algorytmów bezstratnej kompresji danych. Jego idea polega na zamianie ciągów powtarzających się symboli oznaczeniem, że wystąpił ciąg danego symbolu i określeniu jego długości.

Rozpatrzmy różnice pomiędzy tym kodowaniem na przykładzie pliku zawierającego następujące dane:

ABBBBBBBBBCCDEFB

W pierwszym sposobie musimy mieć jakiś symbol, który nie istnieje w kompresowanym pliku, do oznaczenia sekwencji powtarzających się symboli. W naszym przypadku może to być znak średnika. Zatem nasz przykładowy plik po skompresowaniu wyglądałby następująco: AB;9;C;2;DEFB Należy zauważyć, że znak średnika wykorzystujemy tylko do oznaczenia powtórzeń, dzięki temu nie tracimy miejsca w przypadku pojedynczych symboli.

W odniesieniu do kodu:

Program zapisuje listę słowników w której kluczem jest znak a jego wartość to liczba wystąpień w sekwencji.

Algorytm Lempel-Ziv-Welch

Metoda LZW jest względnie łatwa do zaprogramowania, daje bardzo dobre rezultaty. Wykorzystywana jest m.in. w programach ARC, PAK i UNIX-owym compress, w formacie zapisu grafiki GIF, w formatach PDF i PostScript (filtry kodujące fragmenty dokumentu) oraz w modemach (V.42bis). LZW było przez pewien czas algorytmem objętym patentem, co było przyczyną podjęcia prac nad nowym algorytmem kompresji obrazów, które zaowocowały powstaniem formatu PNG. Gdy w przypadku LZ78 występuje konieczność kodowania drugiego elementu pary $\langle i, c \rangle$, w LZW istnieje sposób na usunięcie tej konieczności. Aby było to możliwe, trzeba na początku kodowania umieścić w słowniku wszystkie litery alfabetu wejściowego. Koder gromadzi kolejne symbole z ciągu danych wejściowych, tworząc z nich wzorzec p tak długo, jak długo p jest elementem słownika. Jeśli dodanie kolejnej litery a do wzorca daje wzorzec $p * a$ ($*$ oznacza konkatencję), którego nie ma jeszcze w słowniku, to do odbiorcy przesyłany jest indeks wzorca p i do słownika zostaje dodany wzorzec $p * a$. Następnie zaczynamy odczytywanie kolejnego wzorca, począwszy od litery a .

W odniesieniu do kodu:

Program zapisuje listę w której znaki są kodowane jako typ char, natomiast liczby to miejsce wzorca w słowniku wejściowym.

Instrukcja uruchomienia

Aby uruchomić program należy z poziomu linii komend będąc w folderze z plikami .py wywołać komendę:

```
python CompressionExecutor.py <arg1> <arg2>
```

gdzie:

arg1 – skrót wywoływanego algorytmu (**lzw** dla Lempel-Ziv-Welch, **rle** dla Run Length)

arg2 – ciąg znaków który ma zostać skompresowany danym algorytmem (najlepiej taki w którym jest dużo powtórzeń)

Źródła:

<https://pl.wikipedia.org/wiki/RLE>

http://www.fileformat.info/mirror/egff/ch09_03.htm

<https://pl.wikipedia.org/wiki/LZW>

<https://www.cs.cf.ac.uk/Dave/Multimedia/node214.html>