

# Huffmanovo kódování

**Huffmanovo kódování** je algoritmus, který využívá metodu kódování s proměnnou délkou slova. Hlavní myšlenka je prostá: znaky (nebo symboly), které se v datech vyskytují nejčastěji, jsou kódovány kratší sekvencí bitů, zatímco méně časté znaky mají kódy delší.

Tímto způsobem se celková velikost dat zmenší, aniž by došlo k jakékoli ztrátě informací při zpětném dekódování.

## Jak algoritmus funguje (Tvorba stromu)

Proces vytvoření Huffmanova kódu probíhá v několika krocích:

- Analýza četnosti:** Spočítáme, kolikrát se každý znak v textu vyskytuje.
- Vytvoření prioritní fronty:** Každý znak se stane samostatným uzlem (listem) stromu s váhou odpovídající jeho četnosti.
- Stavba binárního stromu:**
  - \* Vybereme dva uzly s nejmenší četností.
  - \* Spojíme je do nového uzlu, jehož váha je součtem obou.
  - \* Tento proces opakujeme, dokud nezůstane pouze jeden kořenový uzel.
- Přiřazení kódů:** Cestě doleva k potomkovi přiřadíme hodnotu **0** a cestě doprava hodnotu **1**.

## Příklad v praxi

Mějme slovo: BANANA

- Četnosti: A (3x), N (2x), B (1x).
- Znak 'A' je nejčastější, dostane tedy nejkratší kód (např. 0).
- Znak 'B' je nejméně častý, dostane nejdelší kód (např. 110).

Bez komprese (v ASCII) by každý znak zabral 8 bitů (celkem 48 bitů). Po Huffmanově kódování může celé slovo zabrat například jen 10-12 bitů.

# Prefixová vlastnost

Klíčovou vlastností Huffmanova kódování je, že **žádný kód není prefixem (začátkem) jiného kódu**. Díky tomu při čtení bitového proudu (např. 01101...) dekodér přesně ví, kde jeden znak končí a druhý začíná, aniž by potřeboval oddělovače. Jakmile narazí na sekvenci, která odpovídá nějakému znaku v Huffmanově stromu, okamžitě jej zapíše a začne číst další znak od nuly.

## Výhody a omezení

Vlastnost	Popis
<b>Beztrátovost</b>	Po dekompresi získáme identická data, bit po bitu.
<b>Optimalita</b>	Je to nejefektivnější metoda kódování jednotlivých symbolů.
<b>Statická vs. Dynamická</b>	Vyžaduje buď uložení tabulky kódů spolu se souborem (statická), nebo složitější výpočet za běhu (adaptivní).
<b>Závislost na četnosti</b>	Pokud mají všechny znaky stejnou četnost, komprese nebude fungovat (soubor se nezmenší).

## Kde se Huffmanovo kódování používá?

Huffmanův algoritmus se málokdy používá samostatně, většinou je součástí složitějších kompresních řetězců:

- **Deflate:** Algoritmus v ZIPu a PNG, který kombinuje Huffmanovo kódování s metodou LZ77.
- **Multimedia:** Poslední fáze komprese u formátů JPEG a MP3.
- **Síťová komunikace:** Součást protokolu HTTP/2 pro kompresi hlaviček.

*Související pojmy: Binární strom, Algoritmus, Komprese dat, ZIP, JPEG, Bitová hloubka, Entropie.*

From:  
<https://serviceit.cz/> - IT ENCYKLOPEDIE

Permanent link:  
[https://serviceit.cz/doku.php?id=huffmanovo\\_kodovani](https://serviceit.cz/doku.php?id=huffmanovo_kodovani)

Last update: 2025/12/31 19:46

