

A tömörítő eljárások lényege: olyan változó szóhosszúságú kódot választunk, hogy a sokszor előforduló szimbólumokhoz rövidebb, míg a ritkábbakhoz hosszabb kódot rendelünk.

## Shannon-Fano eljárás

Egyszerű adattömörítési eljárás, amelynek lényege az, hogy:

1. A továbbítandó szimbólumokat valószínűség szerint rendezzük.
2. A szimbólumhalmaz két lehetőség azonos valószínűségű részhalmazra bontjuk. Az egyikhez 0, a másikhoz 1 szimbólummal kezdődő szót rendelünk.
3. Az eljárást addig folytatjuk, míg el nem fogynak szimbólumok.

### Példa

Szimbólum	Valószínűség	Kód	Szóhossz
X1	0.25	00	2
X2	0.25	01	2
X3	0.125	100	3
X4	0.125	101	3
X5	0.0625	1100	4
X6	0.0625	1101	4
X7	0.0625	1110	4
X8	0.0625	1111	4

8 szimbólumot használunk a kódolásnál, amelynek egyes szimbólumai a táblázatban megadott valószínűséggel szerepelnek a küldendő üzenetben. Ezeket előfordulási gyakoriság alapján csökkenő sorrendbe rendeztük, azaz a leggyakoribbak vannak elől. A harmadik oszlopban hozzárendeltük változó hosszúságú kódokat, a gyakoriakhoz hosszabbakat.

A 8 szimbólum teljes eseményrendszert alkot, hiszen egymást kizáró eseményekből áll, valamint a szimbólumok előfordulási valószínűségeik összege 1.

Tömörítsük az X2 X3 X8 X7 X1 üzenetet:

Az eredmény:

011001111111000

A tömörítés nélkül az üzenet hossza  $4 * 5$  azaz 20 bit lett volna. A tömörítés után 15 bitre csökkent.

Számítsuk ki a kód entrópiáját.

$$H = - \left( \frac{2}{4} \log \frac{1}{4} + \frac{2}{8} \log \frac{1}{8} + \frac{4}{16} \log \frac{1}{16} \right) = 2.75 \text{ bit}$$

## Huffman kódolás

A Huffman kódolás egy veszteségmentes adatkompressziós algoritmus, amely a gyakrabban

előforduló szimbólumokhoz rövidebb, míg a ritkábban előfordulókhöz hosszabb kódokat rendel. Ez az előfordulási gyakoriság alapján történik az adatban. Az eljárás optimális, mert a legkisebb átlagos kódhosszt eredményezi.

## A Huffman-kódolás lépései

1. Számoljuk ki az egyes karakterek előfordulási gyakoriságát az adatban.
2. Építsünk bináris fát a gyakorisági adatok alapján.
3. Rendeljünk bináris kódokat minden karakterhez a fa szerkezetét követve, biztosítva, hogy a gyakoribb karakterek rövidebb kódokat kapjanak.

## Huffman-kódolási példa

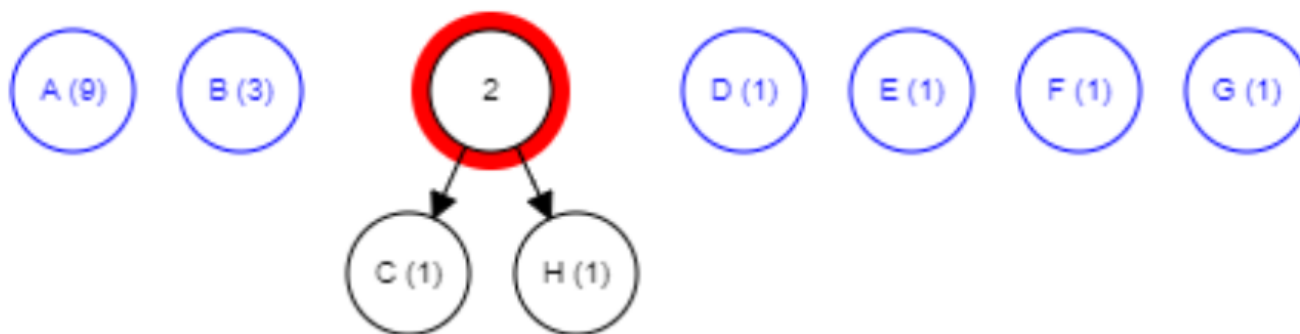
Kódoljuk a következő szöveget: BACADAEAFABBBAAAGAH

Karakter	Gyakoriság
A	7
B	3
C	1
D	1
E	1
F	1
G	2
H	1

Készítsük el a bináris fa leveleit úgy, hogy zárójelben feltüntetjük a gyakoriságokat:



Válasszuk ki a két legkisebb gyakoriságú levelet és vonjuk össze az alábbiak szerint:



From:  
<https://edu.iit.uni-miskolc.hu/> - Institute of Information Science - University of Miskolc

Permanent link:  
[https://edu.iit.uni-miskolc.hu/tanszek:oktatas:infrendalapjai\\_architekturak:informacio\\_feldolgozas:toemoerites?rev=1731417385](https://edu.iit.uni-miskolc.hu/tanszek:oktatas:infrendalapjai_architekturak:informacio_feldolgozas:toemoerites?rev=1731417385)

Last update: **2024/11/12 13:16**

