CURS 0x02

Cantitatea de informație

$$I(x_i) = \log_2\left(\frac{1}{p_i}\right)$$

unde pi este probabilitatea evenimentului xi

OBS: cu cât pi este mai mic cu atât cantitatea de informație este mai mare

Teoria probabilitatii:

p(evenimentA) = nf / np
p(evenimentA + evenimentB) = p(evenimentA) * p(evenimentB)

Entropia

= valoarea medie de informației primită despre o variabilă X

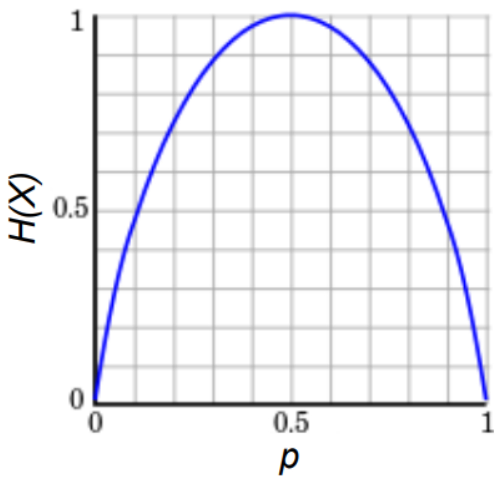
$$H(X) = E(I(X)) = \sum_{i=1}^{N} p_i \log_2 \frac{1}{p_i} = \sum_{i=1}^{N} -p_i \log_2 p_i$$

H(X) se numește entropia lui X
I(X) este informația despre X
E este “expected value”, operația care calculează valoarea medie

entropia ne spune limita inferioară de biți necesari pentru a coda o variabila fără să pierdem informație

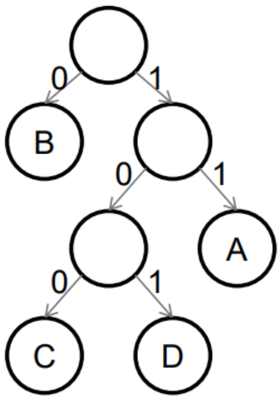
entropia este limita de compresie posibilă

exemplu considerăm o monedă pe care o putem arunca:



Codarea eficienta și unică

- **cum putem crea o codare eficientă și unică?**
  - un arbore binar
    - frunzele sunt codurile
    - stânga/dreapta e decis de 0/1
    - codarea este:
      - B = 0
      - A = 11
      - C = 100
      - D = 101
    - asta garantează codare eficientă
    și decodare unică



  - cum generăm codarea eficientă?
    - algoritmul Huffman
    - input: probabilitatea fiecărui eveniment {1/3, ½, 1/12, 1/12}
    - output: codurile care se citesc de pe un arbore binar (mai sus)
    - cheia: unele evenimente/simboluri apar mai des decât altele, deci acestea primesc o codare mai scurtă
    - dacă toate evenimente sunt equiprobabile, atunci nu putem face nimic

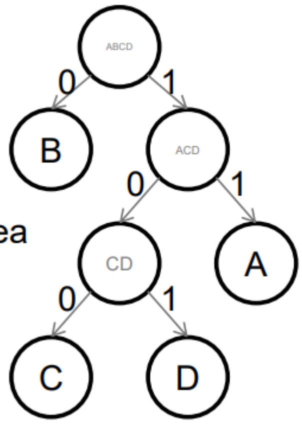
- exercițiu: cum calculăm eficiența acestei codări? dimensiunea in medie a unui mesaj este? probabilitățile sunt {1/3, ½, 1/12, 1/12}
  - 2x1/3 + 1x1/2 + 3x1/12 + 3x1/12 = 1.667 biți
  - comparat cu 1.626 biți care e optim

coeficientul este dat de nr de biți din codificare:

A -> 2 B -> 1 C -> 3 D -> 4

Algoritmul Huffman

- **algoritmul Huffman**
  - input: probabilitatea fiecărui eveniment {1/3, ½, 1/12, 1/12}



- **algoritmul:**
  - luați evenimentele cele mai improbabile
    - C și D, și le punem în arbore
    - creăm un nou eveniment CD, probabilitatea de apariție a acestuia este 1/6 (suma C și D)
    - noile evenimente sunt {A, B, CD} cu probabilități {1/3, ½, 1/6}
  - din nou evenimentele cele mai improbabile
    - A și CD, A merge pe cealaltă frunză
    - noile evenimente sunt {B, ACD} cu probabilități {½, ½}
  - din nou evenimentele cele mai improbabile
    - acum sunt doar două, {B, ACD}, B merge pe cealaltă frunză

Distanța Hamming:

Distanța Hamming între două șiruri binare = câți biți sunt diferiți (biți de pe aceleași poziții în preprezentarea binară)

ex: initial : 0110 0101
corupt : 1100 0001 => Distanța Hamming = 3