

## 2. tétel

Entropia: Egy bizonyos átlagos információtartalom.

Def.: Ha az  $A_1, A_2, \dots, A_n$  jeleket  $P_1, P_2, \dots, P_n$  valószínűséggel adják, ahol  $\sum_{i=1}^n P_i = 1$ , akkor az entropia  $[H(P_1, P_2, \dots, P_n) =$  definiált szerint

$$-\sum_{i=1}^n P_i \log_2 P_i$$

súlyozott átlag ( $P_i$ -t átlaga)

Mindenképp valószínűséggel meg van korozva.

$$\text{ha } \forall P_i = \frac{1}{n}$$

$$H(\quad) = -\sum_{i=1}^n \frac{1}{n} \log_2 \frac{1}{n} = -\left( \frac{1}{n} \cdot \log_2 \frac{1}{n} + \frac{1}{n} \cdot \log_2 \frac{1}{n} + \dots \right) =$$

$n$  db tagja van

$$= n \cdot \frac{1}{n} \cdot \log_2 n = \log_2 n$$


a rendszer

az entropia általános információtartalmát írja le, vagyis a rendszer bizonytalanságát jelzi

Bizonytalanság: 1 jel kibocsátásakor annyi információt nyújtunk, hogy a bizonytalanság megszűnjék

Az az esemény kevésbé bizonytalan, amelyik entropiája nagyobb.

Entropia tulajdonságai:

1, az entropia folytonos a  $(0, 1]$  intervallumban. 

$$0 < P_i \leq 1$$

Biz.:  $-\sum P_i \log_2 P_i$  folytonos  $\rightarrow$  mivel folytonos a  $(0, 1]$  és

a folytonos függvény összege folytonos, így IGAZ.

2, az entropia nulla, ha  $\forall P_i$ -t felvesszünk egy másikral, ugyanazt kapjuk

Biz.:  $p_i \log_2 p_i$ -t felvesszük  $\varepsilon$ -ban.

Tagjai felváltóak, mert az összeadás kommutatív.

3, az entropia max-át felveszi  $p_i = \frac{1}{n}$  esetén, ha minden  $i$ -re igaz, hogy  $P_i = \frac{1}{n}$

(Ez az a legnagyobb az  $E$ , ha a  $P_i$ -t egyenlő valószínűséggel fordultok elő.)

Kódolás fogalma, határolás:

Ugyan, hogy a natoma szelítési tudja az információt, hogy  $n$  kell.

Kódolás. Def.: Ha az adott jeleket ( $A_1, A_2, \dots, A_n$  jeleket)

$P_1, P_2, \dots, P_n$  valószínűséggel adja és a natoma  $x_1,$

$x_2, \dots, x_m$  alapjelével képes dolgozni, szelítani, akkor a

natoma szelítése érdekében az  $x_{s1}, x_{s2}, \dots, x_{sm}$  jelsozatra

való kódolásról van szó.

ne sokkal kisebb, mint  $n$ .

Digitális esetben  $n$  pontosan 2.  $(0, 1)$



Def.: Az  $A_i$  jelű és az  $X_{ij}$  jelű való ( $X_i$  alapjellel segítségével) kölcsönösen egyértelmű leképezése úgy, hogy 'de kódolható' legyen.

A kódolás EGYÉRTELMŰ, ha  $X_{i|j} \neq X_{j|i}$  -ből következik

$$A_i \neq A_j$$

A kódolás KÖLCSÖNÖSEN EGYÉRTELMŰ, ha  $A_i \neq A_j$  -ből következik minden  $i$ -re,  $j$ -re, hogy  $X_{i|j} \neq X_{j|i}$ .

Többféle kódolási eljárás létezik úgy, hogy az egyértelműen de kódolható.

A kódolás hatékonysága: Átlagos kódhossz ha

$A_1, A_2, \dots, A_n$  jeleket adják  $P_1, P_2, \dots, P_n$  valószínűséggel,

melyeknek háza  $h_1, h_2, \dots, h_n$ , akkor az átlagos

$$\text{kódhossz } \bar{L} = \sum_{i=1}^n p_i h_i$$

szilyozott átlagos kódhossz

pl.:  $K_1: \bar{L} = \underbrace{\frac{1}{2}}_{\text{háza}} \cdot 2 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 2 + \frac{1}{8} \cdot 2 = 2$

#### 4. tétel

##### I. Separáló bináris kódolás:

Él: Olyan bináris kódrendszert konstruálunk, ahol biztosítva van az, hogy separálójelű nélkül is egyértelműen de kódolható.

$$F = \{A_1, \dots, A_n\}$$

algoritmus: a, Összekeljük a halmazt két-két részre ( $F_0$  és  $F_1$  halmazokra) úgy, hogy  $F_0 \cup F_1 = F$ . Rendeljük a  $F_0$  halmaz elemeihez 0-t, a  $F_1$  halmaz elemeihez 1 kódot, és újratesszük a kódot az eddigi kód után.

b, A  $F_0$  halmazt összekeljük két részre, amíg lehet, újratesszük az eddigi kódot után 0,1 kódot, és ugyanígy tesszük meg a  $F_1$  halmazzal, egészen addig, amíg elkészül.

Egyértelműen de kódolható!

pl.:

$A_1$	0 0
$A_2$	0 1
$A_3$	1 0 0
$A_4$	1 1 0
$A_5$	1 1 1

##### II. Huffman kódolás:

Optimális kód hosszúságjára szolgál. Olyan eljárást biztosít, mely kétszöveges elosztásból kiindulva egy maximális hatékonyságú prefix kódhoz vezet.

(Prefix: olyan kód, ahol egyik kód sem eleje a másinak.)



$$X := \{A_1, \dots, A_n\}$$

$$p_1 \geq p_2 \geq p_3 \dots \geq p_n$$

algoritmus: a) Rendszű a kódolandó jelet a szerint,  
 hogy a kibocsátás valószínűsége wödező  
 sorrendben legyen.

b) A két legkisebb valószínűsűgűt sorrendben  
 kódoljuk 0,1 kódokkal, és így az eddigi  
 kód elé. Valószínűsűgeket vagyis  
 össze (a két legkisebbet, amíg két jelet  
 nem kapunk). Folytatjuk az eljárású, amíg  
 két jelet nem kapunk.

## 5. tétel

### I. Shannon-Fano file kódolás:

Egyenlő, mint a Huffman kódolás.  
 Ha biztosítható, hogy egyenlő valószínűsűgűre kerül a  
 helyezésűt, akkor optimális. Ez jelenti a gyengepontját,  
 hogy egyenlő valószínűsűgűre kell osztani.

algoritmus: a) A jelet valószínűsűgűt wödező sorrendűbe  
 rendezzű.

b) A jelet halmaszűt két lehetőleg egyenlő  
 valószínűi részhalmassra osztjuk. A  
 egyes részhalmassok tartozó minden  
 jelhez 0-t, a másiban egyet rendelünk.  
 Ezt így az eddigi kód után. Ezt a  
 lépűt ismételjű addig, amíg minden  
 részhalmass már két egyelemű.

$p_i$	$A_i$	
$\frac{1}{2}$	$A_1$	0
$\frac{1}{4}$	$A_2$	1 0
$\frac{1}{8}$	$A_3$	1 1 0
$\frac{1}{8}$	$A_4$	1 1 1



## II. Shannon-féle bináris kódolás:

Annál jobb a Shannon-Fano-tól, hogy az egyenlő valószínűségekre való osztást függetlenül megadja, és biztosítja az optimumot.

algoritmus: a,  $A_1, \dots, A_n$  jelzetet injektív módon valószínűségei sorrendjében ( $p_1 \geq p_2 \geq \dots \geq p_n$ )

b, számítsuk ki a f. sorozat elemeit.

$$\begin{aligned} a_1 &:= \emptyset & \rightarrow & a_1 := \emptyset \\ a_2 &:= a_1 + p_1 & \rightarrow & a_2 := p_1 \\ a_3 &:= a_2 + p_2 & \rightarrow & a_3 := p_1 + p_2 \\ a_n &:= a_{n-1} + p_{n-1} & \rightarrow & a_n := p_1 + p_2 + \dots + p_{n-1} \\ a_{n+1} &:= a_n + p_n & \rightarrow & a_{n+1} := \sum_{i=1}^n p_i = 1 \end{aligned}$$

c., határozzuk meg azokat a legkisebb

$h_i$  egész számokat, amelyekre még

teljesülnek a  $2^{h_i} \cdot p_i \geq 1$  egyenlőtlenségek.

szé.

(A sorozat mon. növ. esetén lesz eöztől legkisebb)

d, Írjuk át az  $\langle a_n \rangle$  elemeit bináris alakba és az  $a_i$  elem törtézetét elemeként tekintjük  $h_i$  hosszúra. (ha rövidebb, egészítsük ki, ha hosszabb vágjuk le)

## 3. tétel

Elődolás fogalma:

A separálós kód egyértelműen elődolozható.

Separálós kód: az az elődolás, amellyel a közlésű és az elemek közötti határolójelek alkalmazása nélkül egyértelműen elődolozható.

Egyértelmű elődolás tétele: A egyértelmű elődolás szükséges feltétele, hogy a kódrendszer prefix legyen.

Prefix kód: az olyan kódok, ahol egyik kód sem eleje egy másik kódnak, prefix kódok, vagy irreducibilis kódok nevezik.



# 14. tétel

14. tétel

## I. Dos:

### DOS (Disk Operating System)

- egyfeladatos
- egyfelhasználós
- tökéletes és irányítható módon is futtatható
- egyszerű felületű

Hosszú ideig uralta opr-t a PC-s piacot, és bár nagyon sokat fejlődött, mégis sok hibája van.

- egy időben csak egy pr. futtatható
- segédprogramok nélkül csak 640 KB memóriát képes kezelni
- a file- és alkönyvtárak azonosítója max. 8+3 karakter
- felület barátságos

## II.

### a, Lemezkezelő parancsok:

- MD, elérési út \ név : Könyvtár létrehozása  
 CD, " - " - : Könyvtár & elérési útvonal létrehozása  
 RD, " - " - : Könyvtár törlése  
 FORMAT, meghajtó / s : rendszerlemez formátálása  
 FORMAT, " - " - / F : méret : adott lekapcsolású lemez  
 FORMAT, " - " - / U : immálinthetetlenül töröl mindent  
 DIR, elérési út / file név / ... : alkönyvtárak listázása  
 / P : oldalankénti listázás  
 / W : öt onlogos listázás  
 / S : mindent elír  
 / A : a megadott attribútumokat listázza  
 D : könyvtár, H : rejtett, R : csak olvasható,  
 S : rendszerfile, A : archív  
 / O : omi alapján rendezve ír ki  
 N : név (abc sorrend), D : dátum

VOL, megh. : a lemez nevét és sorrendjét adja meg

LABEL, megh. útv. : kötet létrehozása ill. megváltoztatása

CHKDSK, megh/út/file név / : info-t ír ki a lemezről  
 / F : ha lehet, javítja a hibákat a FAT-ban  
 / V : részletes üzenet kiír

DISKCOPY, 1.megh. 2.megh. : átmásol 1-ről a 2-re.

APPEND : kiegészít utat állít be adattallományhoz

JOIN : 1 meghajtót hozzákapcsol az úttal megadott könyvtárhoz

SUBST : az úttal megadott alkönyvtárhoz a megadott meghajtó betűjelét rendel

b, FILE elérési út : COPY : másolás (A; B; U)  
 DEL : törlés (P)

REN : állomány átnevezése

ATTRIB : attribútum megadás

SHARE : engedélyezi az állományok megosztott használatát

TYPE : áll. tartalmának kiírása

PRINT : áll. nyomtatása

### c, Parancsállományokban használható:

CALL : batch indítása batch file-ből

ECHO ON, ECHO OFF, ECHO szöveg : növeg kiírása

GOTO : ugrás

IF : feltételes végrehajtás

FOR változó IN DO parancs : cikluskezelés

SHIFT : a formális paramétereket eggyel balra lépteti

REM növeg : megjegyzés

PAUSE : szünet.



## d, Egyéb parancsok:

- CLS: Építménytörölés  
DATE: dátum beállítás + beállítás  
TIME: idő.  
VER: DOS verziószámának beállítás  
MORE: a kimenetet építmények bővítése  
FIND: a megadott keresési sorozatot keresi  
(/ : amit nem tartalmaz, /C : megnéző a sorozat)  
PROMPT: prompt beállítás  
(\$p, \$g, \$d, \$t, \$v)

az opr. kézi leltőre a számítógép használatát.  
Egy olyan programrendszer, amely a PC bekapcsolásakor  
automatikusan elindul és listázza

- a felh.-i programok futtatásának lehetőségét
- a PC állapotának ellenőrzését
- a felh. és a PC közötti kapcsolatok fenntartását.

## 12. tábl

Program: Azokat a PC által közvetve vagy közvetlenül végrehajtható utasításcsoportokat, amelyek egy feladat megoldása érdekében vezélik a PC-t, programoknak nevezzük.

Softver: A softver a számítógépes programok, a pr.-ok által használt adatok és a programokhoz mellékelte dokumentációk összessége.

A PC-nél a legegyszerűbb feladat végrehajtásához a memóriába betöltött programra van szükség.

- Hogy a PC bekapcsolás után azonnal elinduljon, egy egyszerű programot építünk bele, melynek BIOS a neve.
- Bekapcsolás után a BIOS azonnal elindul, elvégzi az öntesztet, és betölti az opr.-t.
- az opr. számos egyéb feladata mellett lehetővé teszi, hogy a felh. utasításokat alapján a PC-nél, továbbá azt, hogy betölthesse a memóriára működés felhatalmazási csoportokat.
- A felh. a betöltött felh.-i program segítségével elvégzi a konkrét feladatokat.

Felh. és hardver kapcsolata:

