① "TWENTY QUESTIONS" (esempio: sito AKINATOR.COM)

Gioco: Giocatore A sceglie un oggetto X da un insieme universo $\mathcal{X}$

Giocatore B cerca di indovinare X con sequenze di domande a risposta sì/no.

Supponiamo B usi un codice a lunghezza media minima rispetto a p_X

Se osserviamo una media di 20 domande, qual è il numero minimo di elementi dell'universo $\mathcal{X}$?

Codifica ottima

Codice di Shannon-Fano

$$E[L^*] \leq E[L] < H(X) + 1 \leq \log|\mathcal{X}| + 1$$

($\forall$ codifica: $E[L] \geq H(X)$)

$$20 \leq \log|\mathcal{X}| + 1$$

$$19 \leq \log|\mathcal{X}|$$

$$|\mathcal{X}| \geq 2^{19} \quad \sim \text{mezzo milione}$$

(2) Siano $X_1, X_2, X_3, \dots$ v.a. indipendenti identicamente distribuite su $\mathcal{A} = \{a, b, c, d, e, f\}$

con $p(a) = 1/2$, $p(b) = 1/4$, $p(c) = 1/8$, $p(d) = 1/16$, $p(e) = p(f) = 1/32$.

(a) Calcolare $H(X_1)$

→ (b) Qual è la sequenza più probabile di lunghezza n? Quanto vale la sua probabilità?

(c) In relazione all'insieme di tipicità

$n=10$

$$\mathcal{T}^{(n,\delta)} = \{ \vec{x} \in \mathcal{A}^n : | \overbrace{-\tfrac{1}{n} \log p(\vec{x})}^{\mathcal{I}(\vec{x})/n} - H(\vec{X}) | < \delta \},$$

$p(\overbrace{baacddabbf}) = \dots$

$p(cccabacbae) = \dots$

dire se la sequenza trovata al punto (b) appartiene a $\boxed{\mathcal{T}^{(n,0.1)}}$. $(\delta = 0.1)$

Cosa si può dire su $\mathcal{T}^{(n,1.0)}$?

(a) $H(X_1) = 1/2 \cdot 1 + 1/4 \cdot 2 + 1/8 \cdot 3 + 1/16 \cdot 4 + 2/32 \cdot 5 = 1/2 + 1/2 + 3/8 + 4/16 + 5/16 = 1 + 15/16 = 2 - 1/16 = 1.9375$

(b) $p(x^n) = \prod_{i=1}^n p(x_i) \xrightarrow{\max} x^n = \underbrace{aaaaaaaaaa}_{n}$ $\quad p(x^n) = (1/2)^n = 2^{-n}$

(c) $H(\vec{X}) \overset{\downarrow}{=} n H(X_1)$ $\qquad \vec{X} = (X_1, X_2, \dots, X_n)$

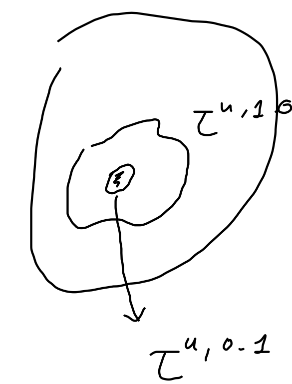
$$x^n \in \tau^{(n,\delta)} \iff 2^{-n[\underset{\uparrow\, H(X_1)}{H}+\delta]} \le p(x^n) \le 2^{-n[\underset{\uparrow\, H(X_1)}{H}-\delta]}$$

$\delta = \underline{0.1}$
$H = 1.9375$ bit

$$2^{-n[2.0375]} \overset{?}{\le} p(x^n) \overset{?}{\le} 2^{-n[1.8375]}$$

$$p(x^n) = 2^{-n}$$

$2^{-n\cdot[1]} \to$ NO $\qquad x^n = (\overbrace{aa\dots a}^{n}) \notin \tau^{(n,0.1)}$



$\tau^{(n,1.0)}$

$\delta = \underline{1.0}$

$$2^{-n[2.9375]} \overset{?}{\le} p(x^n) \overset{?}{\le} 2^{-n[0.9375]}$$

$p(x^n) = 2^{-n}$ sì perché $0.9375 \le 1 \le 2.9375$

$x^n = (aa\dots a) \in \tau^{(n,1.0)}$

(3) Considerare la v.a. X a valori in $\{x_1, x_2, \ldots, x_6\}$ e il codice binario

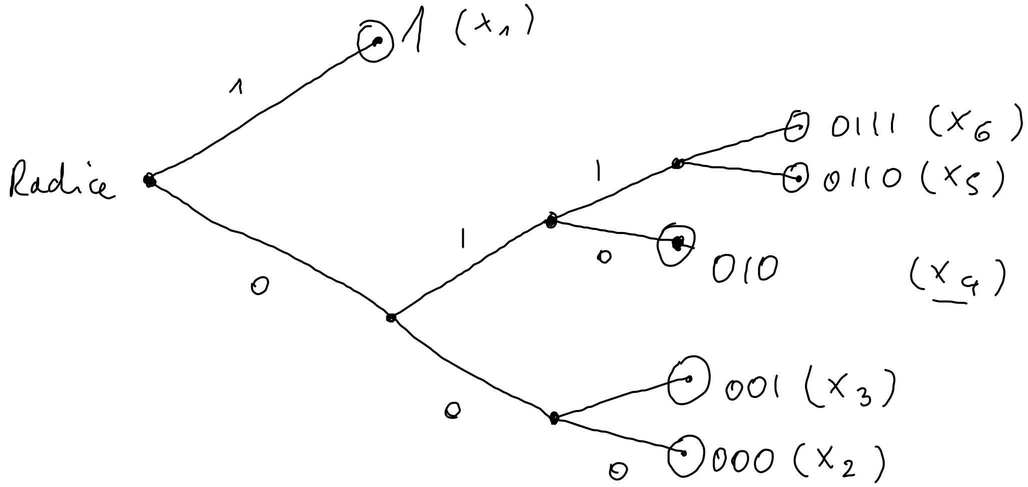
X	$p(X)$	$Y = \varphi(X)$
x_1	0.5	1
x_2	0.15	000
x_3	0.1	001
x_4	0.1	0100 $\to$ 010
x_5	0.1	0110
x_6	0.05	0111

B-LV

(a) Mostrare che il codice è a prefisso disegnandone una rappresentazione ad albero

(b) Derivare l'entropia della sorgente e la lunghezza media del codice

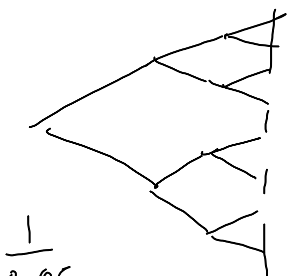
(c) Determinare se il codice è ottimale. Se non lo è, fornire un codice con lunghezza media inferiore.



Albero completo

↳ Ogni nodo ha 0 figli oppure D figli
(nessun nodo ha # figli > 0 ma < D)

Albero zeppo ~~0100~~
→ 010



(a) è un codice a prefisso

(b) $H(X) = 0.5 \log \frac{1}{0.5} + 0.15 \log \frac{1}{0.15} + 3 \times 0.1 \log \frac{1}{0.1} + 0.05 \cdot \log \frac{1}{0.05}$

$\simeq \underline{2{,}123}$ bit

$0.15 + 0.1$

$E[L] = E_X[|\varphi(X)|] = 0.5 \cdot 1 + 0.25 \cdot 3 + 0.25 \cdot 4 = 0.5 + 0.75 + 1.0 = \boxed{2.25}$

(c) Non è ottimale: esiste un altro codice binario con lunghezza media inferiore:

$E[L] = 0.5 \cdot 1 + 0.25 \cdot 3 + 0.1 \cdot 3 + 0.15 \cdot 4 = 0.5 + 1.05 + 0.6 = \underline{2.15}$

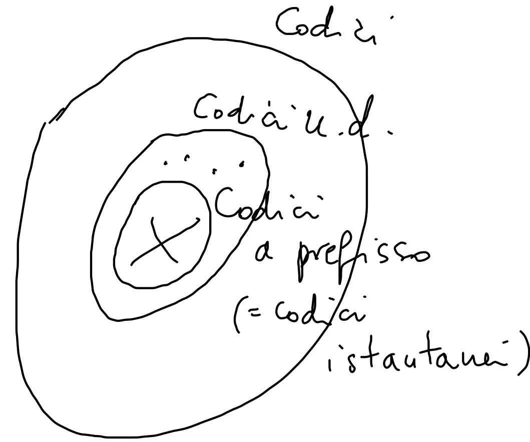
④ Dati i tre codici B-LV: $\mathcal{A}=\{x_0, x_1, \dots, x_5\}$ $\mathcal{B}=\{0,1\}$

	(1)	(2)	(3)
x_0	11 ×	01 ×	01 ×
x_1	10 ×	10 ×	10
x_2	01 ×	110 ×	001
x_3	001 ×	111 ×	100
x_4	0001 ×	0000 ×	000
x_5	0000 ×	0001 ×	111

10 è prefisso di 100

$\overset{x_1}{10}/\overset{x_2}{001}$

$x_3\ 100/01\ x_0$



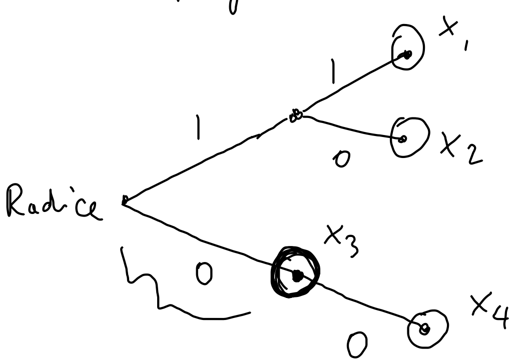
quale di questi codici è u.d.? Perché?

(1) è a prefisso $\Longrightarrow$ è u.d.

(2) è a prefisso $\Longrightarrow$ è u.d.

(3) non è a prefisso
(potrebbe però essere u.d.)
Non u.d.

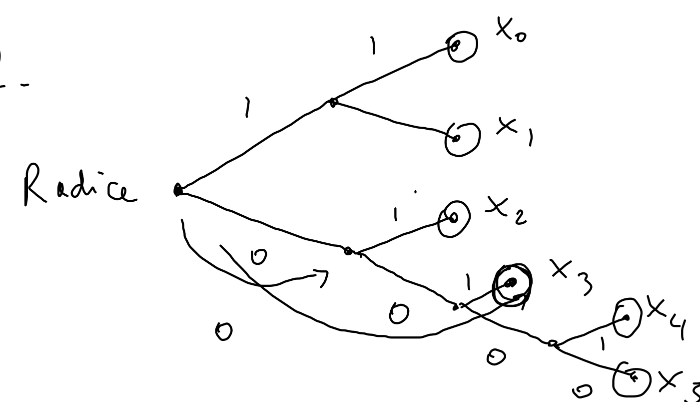
(codice a prefisso $\Longleftrightarrow$ istantaneo)



codice non a prefisso

$\boxed{0}$ $\boxed{0}$ $\boxed{0}\,0\ldots$

$\varphi(x_3)\,\varphi(x_3)\,\varphi(x_1)$
$\varphi(x_4)\,\varphi(x_1)$



001 | ✓ $\Longrightarrow x_3$

Algoritmo Sardinas-Patterson

$W = \{01, 10, 001, 100, 000, 111\}$

W	Residui liv. 1	Residui liv. 2
01	0	1
10		$\boxed{01} \in W$
001		00
100		
000		$\Downarrow$
111		non è u-d.

Residui $\emptyset$ $\Rightarrow$ è a prefisso $\Rightarrow$ è u-d.

10	
10	0

Residui di livello 0 = W

Residui di livello 1:

confrontando i residui di livello 0 con le parole di codice

Residui di livello n:

confrontando i residui di livello n − 1 con le parole di codice

Si ottiene una qualche parola di codice come residuo $\Longleftrightarrow$ il codice non è u-d.

Es. $W = \{10, 010, 1, 1110\}$

W	Residui di liv. 1	Residui di liv. 2
10	0	10 $\in W$
010	110	$\Downarrow$
1		codice non è u.d.
1110		

Es. $W = \{0, 001, 101, 11\}$

R4 = R3

W	R1	R2	R3	R4
0	01	1	01	1
001			1	01
101				
11				

$\Rightarrow$ il codice è u.d.