1. Siano X, Y due v.a. sullo stesso alfabeto $A = \{a_1, a_2, a_3, a_4, a_5\}$ con d.d.p. p_X, p_Y rispettivamente.

Consideriamo due funzioni di codifica φ_1, φ_2 per gli elementi di A

simbolo(x)	$p_X(x)$	$p_Y(x)$	$\varphi_1(x)$	L	$\varphi_2(x)$	
a_1	1/2	1/2	0	(1)	0	(1)
a_2	1/4	1/8	10	(2)	100	(3)
a_3	1/8	1/8	110	(3)	101	(3)
a_4	1/16	1/8	1110	(4)	110	(3)
a_5	1/16	1/8	1111	(4)	111	(3)

(1) Calcolare $H(X), H(Y), D(p_X \| p_Y), D(p_Y \| p_X)$

(2) Mostrare che la lunghezza attesa delle parole di codice di φ_1 utilizzata per la v.a. X coincide con $H(X)$ e che quindi la codifica φ_1 è ottima per X. Mostrare che φ_2 è ottima per Y.

(3) Si supponga di utilizzare la codifica φ_2 per la v.a. X. Che lunghezza attesa delle parole di codice otteniamo? Di quanto è maggiore di $H(X)$?

Si evidenzi la relazione con $D(p_X \| p_Y)$.

(1) $\boxed{p_X = (1/2, 1/4, 1/8, 1/16, 1/16)}$, $p_Y = (1/2, 1/8, 1/8, 1/8, 1/8)$

$H(X) = 1/2 \cdot 1 + 1/4 \cdot 2 + 1/8\, 3 + 2/16 \cdot 4 = 1/2 + 1/2 + 3/8 + 1/2 = 15/8 = 2 - 1/8$. (bit)

$H(Y) = 1/2 \cdot 1 + 4/8 \cdot 3 = 1/2 + 12/8 = 1/2 + 3/2 = 2$ bit.

$\boxed{D(p_X \| \underset{\uparrow \varphi_2}{p_Y})} = \sum_{i=1}^{K} (p_X)_i \log \frac{(p_X)_i}{(p_Y)_i} = \cancel{1/2 \cdot \log 1} + 1/4 \log 2 + \cancel{1/8 \log 1} + 2/16 \log 1/2$

$= 1/4 + 1/8(-1) = \boxed{1/8}$.

$D(p_Y \| p_X) = \cancel{1/2 \log 1} + 1/8 \log 1/2 + \cancel{1/8 \log 1} + 2/8 \log 2 = -1/8 + 1/4 = 1/8$.

(2) $\overset{\swarrow \varphi_1}{\mathbb{E}[L^{(1)}]} = 1/2 \cdot 1 + 1/4 \cdot 2 + 1/8 \cdot 3 + 1/16 \cdot 4 + 1/16 \cdot 4 = 15/8 = 2 - 1/8 = H(X)$

$\Rightarrow$ quindi φ_1 è ottima per la v.a. X (tra i codici u.d. con blocchi di lunghezza $n = 1$)

$\mathbb{E}[L^{(2)}] = 1/2 \cdot 1 + 4 \cdot 1/8 \cdot 3 = 1/2 + 1/2\, 3 = 2$ bit $= H(Y) \Rightarrow \varphi_2$ è ottima per Y.

Se uso φ_2 per X, ottengo $\mathbb{E}[L] = 1/2 \cdot 1 + \underbrace{1/4 \cdot 3 + 1/8 \cdot 3 + 2/16 \cdot 3}_{(1/2 \cdot 3)} = 1/2 + 3/2 = 2 > \underset{2 - 1/8 = H(X)}{\underset{\parallel}{\mathbb{E}[L^{(1)}]}}$

$\Rightarrow \varphi_2$ <u>non</u> è ottima per X

Di quanto è maggiore $\mathbb{E}[L]$ di $H(X)$?

$\mathbb{E}[L] = 2 \qquad H(X) = 2 - 1/8 \qquad \overbrace{\mathbb{E}[L] - H(X)}^{\text{ridondanza}} = 1/8$

$\forall$ codifica per la v.a. X : $\mathbb{E}[L] \geq H(X)$ $\rightarrow$ d.d.p. $p = (p_1, \dots, p_k)$

$(|\mathcal{B}| = 2)$

(Caso binario : $D = 2$)

$$q_i = \frac{2^{-l_i}}{\alpha} \qquad \alpha = \sum_{i=1}^{k} 2^{-l_i}$$

$$D(p \| q) = \sum_{i=1}^{k} p_i \log \frac{p_i}{2^{-l_i}} \cdot \alpha = \underbrace{\mathbb{E}[L] - \overbrace{H(X)}^{H(p)}}_{\text{ridondanza}} + \log \alpha$$

Se $\alpha = 1$, la ridondanza è $= D(p \| q)$

Se $\alpha \leq 1$, la ridondanza è $\geq D(p \| q)$

(2) Shannon-Fano : non è (in generale) ottima per n finito
→ Fano : non è (" ") ottima per n finito
Huffman : è ottima per n finito

$\mathcal{A} = \{a, b, c, d, e, f\}$

$p = (\boxed{0.13}, \boxed{0.1}, \boxed{0.03}, \boxed{0.15, 0.18}, \boxed{0.41})$

a	13/100
b	10/100
c	3/100
d	15/100
e	18/100
f	41/100

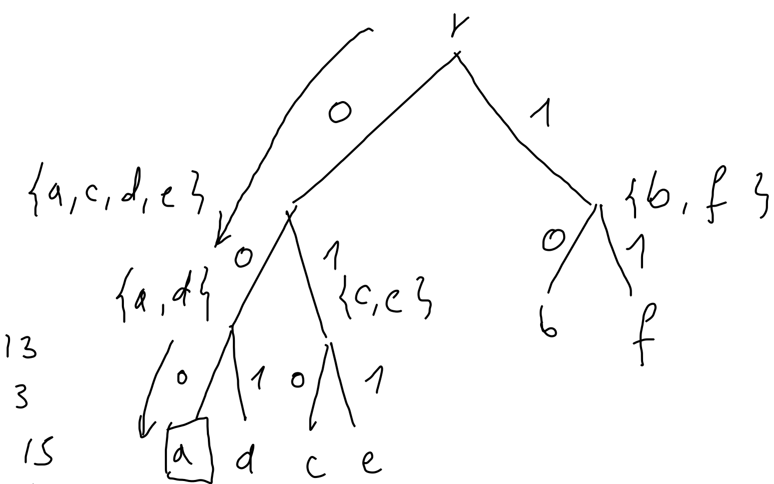
$\{f\}, \{a,b,c,d,e\} \to |41 - 59| = 18$

$\{c,f\}, \{a,b,d,e\} \to |44 - 56| = 12$

$\{a,c,d,e\}, \{b,f\} \to |49 - 51| = 2$

Fano :

	$\varphi(x)$	ℓ
a	000	3
b	10	2
c	010	3
d	001	3
e	011	3
f	11	2



$E[L^{Fano}] = 2 \cdot (0.51) + 3 \cdot (0.49) = 2.49$

$7 \leftarrow$

13
15
3
18

$13 \leftarrow$

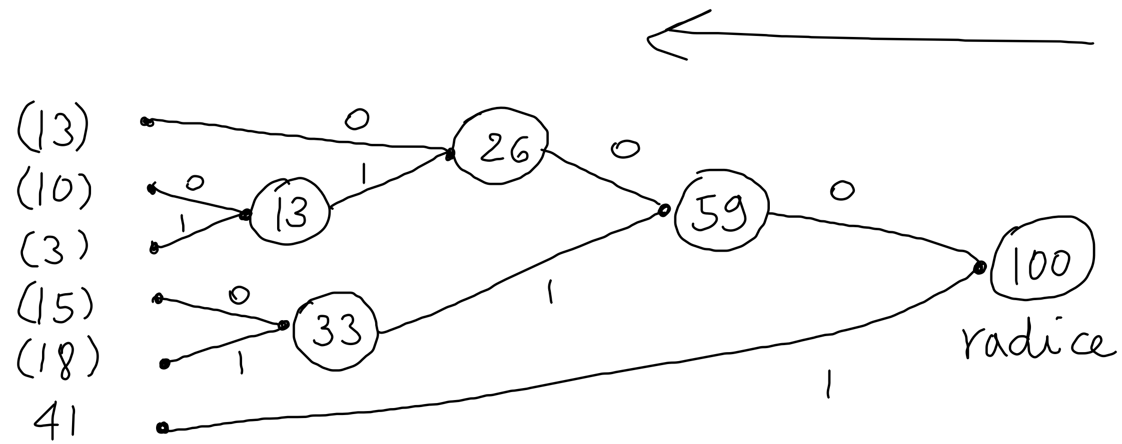
13
3
15
18

a	13
c	3
d	15
e	18

3

Huffman

a	13/100
b	10/100
c	3/100
d	15/100
e	18/100
f	41/100



φ	L
000	3
0010	4
0011	4
010	3
011	3
1	1

$$\mathbb{E}[L^{Huffman}] = 0.13 \cdot 3 + 0.10 \cdot 4 + \dots = 3 \cdot (0.13 + 0.15 + 0.18) + 4 \cdot (0.10 + 0.03) + 1 \cdot (0.41)$$

$$= 2.31 < \mathbb{E}[L^{Fano}] = 2.49 .$$

$\Rightarrow$ la codifica di Fano non è ottima per questa d.d.p.

$|A| = K$

e

$|B| = D$

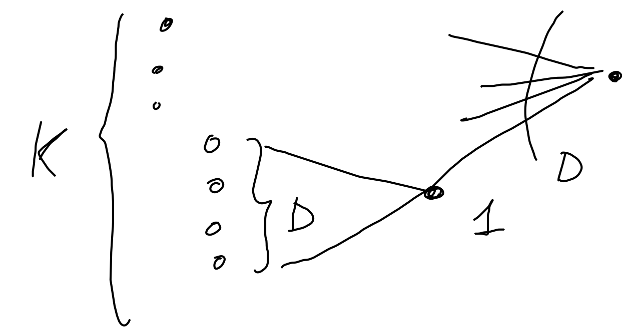
③ Codifica di Huffman per alfabeto di codifica D-ario con $D > 2$

Aggreghiamo ad ogni passo i D simboli meno probabili.

Dà un codice ottimo solo se l'albero di codice risultante è completo: ogni nodo ha o 0 figli o esattamente D figli; questo accade se e solo se $K \overset{(*)}{=} D + j \cdot (D-1)$ per qualche intero nonnegativo j.

$\iff K - j(D-1) = D$ per qualche j

$\iff K - D \equiv 0 \mod (D-1)$

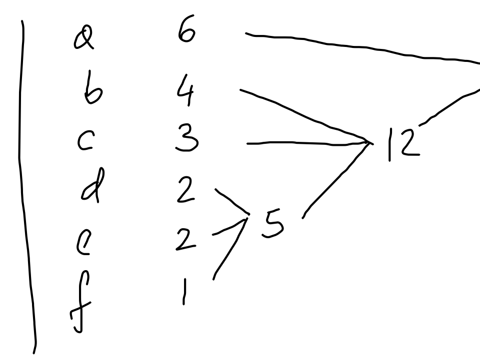


Se la condizione $(*)$ non è soddisfatta, aggiungiamo un numero minimo di simboli fittizi (a probabilità zero) in modo tale da soddisfarla.

Es. $A = \{a, b, c, d, e, f\}$ $(K = 6)$. Voglio $D = 3$.

$p = (6/18, 4/18, 3/18, 2/18, 2/18, 1/18)$

$K = 6$ $D = 3$ $\nexists j$ intero t.c. $6 = 3 + j(2) \to (*)$ non è soddisfatta

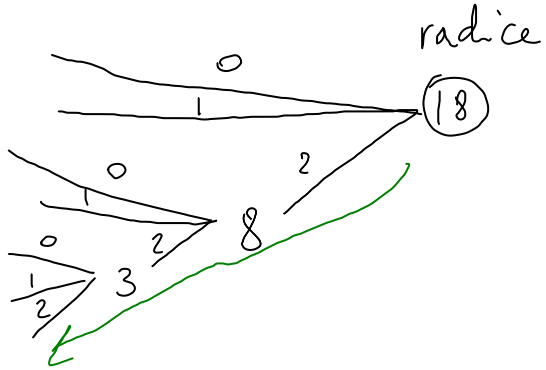


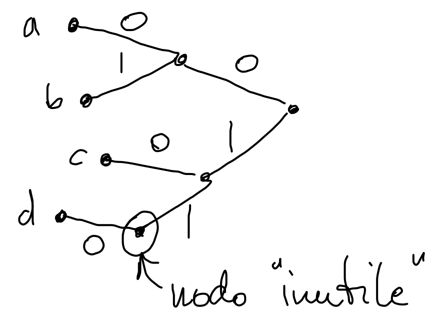
NON è ottimo per p

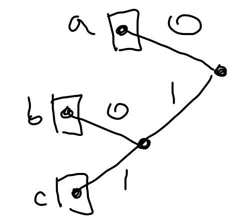
Aggiungo un simbolo fittizio g con prob. $0 = 0/18$.

$\Rightarrow \quad K = 7 \qquad 7 \overset{?}{=} 3 + j(2) \qquad$ OK : $j = 2$

$\varphi(x)$	x	
0	a	6
1	b	4
20	c	3
21	d	2
220	e	2
221	f	1
(222)	g	0





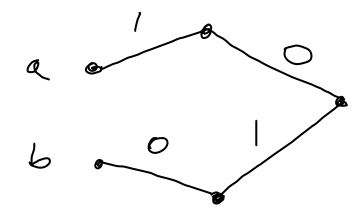


④ Quali di questi insiemi di parole di codice sono ottenibili tramite una codifica di Huffman?

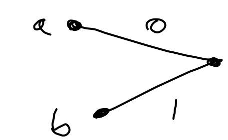
(a) $\{0, 10, 11\}$: sì

(b) $\{00, 01, 10, 110\}$: no : non è ottimo!
$\{00, 01, 10, 11\}$

(c) $\{01, 10\}$: no : non è ottimo



$\Rightarrow$



(5) Decomprimere la seguente sequenza in base 10 compressa col metodo Ziv-Lempel LZ77 : $\boxed{605651600640}$ $\Longrightarrow$ decodifica ?