**Compito#33: ripasso**

**A)** Data la sequenza S = {&, !, \*, !, \*, &, $, \*, !, \*, &, &}

1. calcolare l’entropia H(S) svolgendo tutti i passaggi

$ = 1/12 = 0,08

& = 4/12 = 0,33

\*= 4/12 = 0,33

!= 3/12 = 0,25

H(Si) = -2(0,33\*(log (2)0,33) - (0,25\*(log (2)0,25) - (0,33\*(log(2)0,33) -(0,08\*(log(2)0,08) = -2(0,33\*(log (10)0,33/0,301) -(0,25\*(log (10)0,25/0,301)

-(0,33\*(log (10)0,33/0,301) -(0,08\*(log (10)0,08/0,301)= 0,51+ 0,29+ 0,48 +0.51= 1.79 bit/sym

1. creare una codifica FLC con il numero minimo possibile di bit

FLC= H(Si)\*numero di simboli= 2\*4 = 8

minima = 1,79\*4= 7,16

x

1. calcolare la distanza di Hamming tra tutti i simboli

|  |  |
| --- | --- |
| $ | 00 |
| & | 01 |
| \* | 10 |
| ! | 11 |

$-&= 1

$-\*= 1

$-!= 2

$-\*= 2

$-!= 1

\*-!= 1

1. assumendo la codifica 2) e calcolata la codeword o stringa di bit che rappresenta l’intera sequenza S calcolare il bit di parità pari e dispari

S = {&, !, \*, !, \*, &, $, \*, !, \*, &, &}

011110111001001011100101

bit di parità pari = 0

bit di parità dispari = 1

1. calcolare la checksum di S considerando la codifica 2) svolgendo tutti i passaggi

& xor ! = 00 xor 11=11

! xor \* = 11 xor 10= 01

01 xor \* = 01 xor 10 = 11

11 xor $ = 11 xor 00 = 11

11 xor \* = 11 xor 10 = 01

01 xor ! = 01 xor 11 = 10

10 xor \* = 10 xor 10 = 00

00 xor & = 00 xor 01 = 01

01 xor $ = 01 xor 00 = 01

checksum=01

1. creare una codifica VLC di Huffman della sequenza S svolgendo tutti i passaggi

Immagine che contiene testo, lavagnabianca

Descrizione generata automaticamente

1. alcolare il rapporto di compressione tra il messaggio codificato con 2) e con 6)

rapporto di compressione = 24:24=1