***Trabajo Práctico N° 1***

***TEORIA DE LA INFORMACIÓN Y CONDIFICACIÓN.***

1) ***Dos tiros libres para un jugador de básquet***

Conjunto de resultados 1=Entra 0= No entra y son 2 tiros

**E = {00,01,10,11}**

Los 4 mensajes son equiporbables y la probabilidad de cada uno es de ¼ y su cantidad de informacion es de

**I = log (1/ ¼) = 2bits**

**2) Sorteo para la copa mundial de futbol**

1. La probabilidad de ocurrencia de un símbolo. Existen 32 participantes por lo que la probabilidad de cada símbolo de ser seleccionado será de

*p =*

1. Cantidad de información obtenida al presentarse un símbolo

I = log2 (1/  ) = **5 bits**

1. Cantidad de información de un palabra formada por 4 simbolos.

P = (s1 s2 s3 s4) =()4  I (s1 s2 s3 s4) = log2 (1/)4 =**20 bits**

**3) Fichas de ajedrez**

**A)** Cantidad de información obtenida.

* 1. **P(Peon)= I(peón)= log2 = 1 BIT**
  2. **P(Alfil) = I(alfil)= log2 =2 BITS**
  3. **P(FichaCsillero) ) = I(FichaCsillero)= log2 = 1,19 BITS**

**B)** Aporta mayor información decir que una ficha puede moverse más de un casillero por vez.

**4) Alfabeto binario S = {0, 1}**

P(0101) = 2/3 \* 1/3 \* 2/3 \* 1/3 = 4/81

I(1) = log2 (1/ ) = 1.6 bits

I(0) = log2 (1/ ) = 0.6 bits entonces

I(0101) = I(0) + I(1) + I(0) + I(1) = 4,4 => 5bits

**5) Alfabeto S = {b, a, c, k, u, p}**

* I(b) = log2 (1/ ) = 2,32 bits
* I(a) = log2 (1/ ) = 2 bits
* I(c) = log2 (1/ ) = 3,32 bits
* I(k) = log2 (1/ ) = 2,73 bits
* I(u) = log2 (1/ ) = 2 bits
* I(p) = log2 (1/ ) = 4,32 bits

H(S) = 1/5 . 2,32 + ¼ . 2 + 1/10 . 3,32 + 3/20 . 2,73 + ¼ . 2 + 1/20 . 4,32 = **2,32bits**

**6) Alfabeto S = {S1 , S2 , S3}**

* I(S1) = log2 (1/ ) = 2 bits
* I(S2) = log2 (1/ ) = 2 bits
* I(S3) = log2 (1/ ) = 1 bits

i. Entropia

H(S) = ¼ . 2 bits + ¼ . 2 bits+ 1/2 . 1 bit = 1,5 bits

ii. Extensiones de Segundo orden

* 𝑝(𝑠1𝑠1 ) = 1/4 ∙ 1/4 = 1/16 → 𝐼(𝑠1𝑠1 ) = log2 (1/ ) = 4 𝑏𝑖𝑡s
* 𝑝(𝑠1𝑠2 ) = 1/4 ∙ 1/4 = 1/16 → 𝐼(𝑠1𝑠2 ) = log2 (1/ ) = 4 𝑏𝑖𝑡s
* 𝑝(𝑠1𝑠3 ) = 1/4 ∙ 1/2 = 1/8 → 𝐼(𝑠1𝑠3 ) = log2 (1/ ) = 3 𝑏𝑖𝑡s
* 𝑝(𝑠2𝑠1 ) = 1/4 ∙ 1/4 = 1/16 → 𝐼(𝑠2𝑠1 ) = log2 (1/ ) = 4 𝑏𝑖𝑡s
* 𝑝(𝑠2𝑠2 ) = 1/4 ∙ 1/4 = 1/16 → 𝐼(𝑠2𝑠2 ) = log2 (1/ ) = 4 𝑏𝑖𝑡s
* 𝑝(𝑠2𝑠3 ) = 1/4 ∙ 1/2 = 1/8 → 𝐼(𝑠2𝑠3 ) = log2 (1/ ) = 3 𝑏𝑖𝑡s
* 𝑝(𝑠3𝑠1 ) = 1/2 ∙ 1/4 = 1/8 → 𝐼(𝑠3𝑠1 ) = log2 (1/ ) = 3 𝑏𝑖𝑡s
* 𝑝(𝑠3𝑠2 ) = 1/2 ∙ 1/4 = 1/8 → 𝐼(𝑠3𝑠2 ) = log2 (1/ ) = 3 𝑏𝑖𝑡s
* 𝑝(𝑠3𝑠3 ) = 1/2 ∙ 1/2 = 1/4 → 𝐼(𝑠3𝑠3 ) = log2 (1/ ) = 2 𝑏𝑖𝑡s

Extensión de 3rd orden: 𝑆 3 = {𝑠1𝑠1𝑠1, 𝑠1𝑠1𝑠2, 𝑠1𝑠1𝑠3, 𝑠1𝑠2𝑠1, 𝑠1𝑠2𝑠2, 𝑠1𝑠2𝑠3, 𝑠1𝑠3𝑠1, 𝑠1𝑠3𝑠2, 𝑠1𝑠3𝑠3, 𝑠2𝑠1𝑠1, 𝑠2𝑠1𝑠2, 𝑠2𝑠1𝑠3, 𝑠2𝑠2𝑠1, 𝑠2𝑠2𝑠2, 𝑠2𝑠2𝑠3, 𝑠2𝑠3𝑠1, 𝑠2𝑠3𝑠2, 𝑠2𝑠3𝑠3, 𝑠3𝑠1𝑠1, 𝑠3𝑠1𝑠2, 𝑠3𝑠1𝑠3, 𝑠3𝑠2𝑠1, 𝑠3𝑠2𝑠2, 𝑠3𝑠2𝑠3, 𝑠3𝑠3𝑠1, 𝑠3𝑠3𝑠2, 𝑠3𝑠3𝑠3,}

H(S3) = 3 . H(S) = 3. 3/2 = 4,5 bits

**7) Tasa de información R o velocidad de información.**

I(punto) = log2 (1/ ) = 0,59 bits

I(raya) = log2 (1/ ) = 1,58 bits

R(punto) = = 1,95 bits/seg

R(punto) = = 1,32 bits/seg

**8) Transmisión de trenes de ocho pulsos**

I(tension) = log2 (1/ ) = 2 bits

I(tren) = 7 \* 2 bits = 14 bits

𝑅 = 800 𝑡𝑟𝑒𝑛𝑒𝑠⁄𝑠𝑒𝑔 ∗ 14 𝑏𝑖𝑡𝑠 = 11200 𝑏𝑖𝑡𝑠⁄𝑠𝑒g

𝑉 = 800 𝑡𝑟𝑒𝑛𝑒𝑠⁄𝑠𝑒𝑔 ∗ 8 𝑝𝑢𝑙𝑠𝑜𝑠 = 6400 𝑝𝑢𝑙𝑠𝑜𝑠⁄𝑠𝑒g

**9) Tasa de informacion T de S = {b, a, c, k, u, p}**

𝐻(𝑆) ≅ **2,423 𝑏𝑖𝑡**

𝜏 = 0.3 𝑠𝑒𝑔 ∗ 1/5 + 0.2 𝑠𝑒𝑔 ∗ 1/4 + 0.8 𝑠𝑒𝑔 ∗ 1/10 + 0.7 𝑠𝑒𝑔 ∗ 3/20 + 0.4 𝑠𝑒𝑔 ∗ 1/4 + 0.5 𝑠𝑒𝑔 ∗ 1/20 = 21/50 𝑠𝑒𝑔 = **0.42 𝑠𝑒𝑔**

**T = = 5.77 bits/seg**

**10) Tasa de informacion de un bloque de 19 simbolos**

La probabilidad de cada símbolo será de 𝑝(𝑠) = 1/16, ya que son equiprobables

La entropía de la fuente será 𝐻(𝑆) = 16 ∙ ( 1/16 ∙ log2 16) = log2 16 = 4 𝑏𝑖𝑡𝑠

Teniendo en cuenta que el bloque es una extensión de la fuente de orden 19, su entropía sería: 𝐻(𝑆19) = 19 ∗ 4 𝑏𝑖𝑡𝑠 = 76 𝑏𝑖𝑡s

La duración de cada símbolo se de 1ms por lo que su duración promedio sería: 𝜏(𝑠í𝑚𝑏𝑜𝑙𝑜) = 1 ms

Por lo tanto, la duración del bloque más el de sincronización será de: 𝜏(𝑏𝑙𝑜𝑞𝑢𝑒) = 19 ∗ 1 𝑚𝑠 + 6 ∗ 1 𝑚𝑠 = 25 ms

Así, la tasa de información T sería:

𝑇 = 76 𝑏𝑖𝑡𝑠 / 25 𝑚𝑠 = 3.04 𝑏𝑖𝑡𝑠/𝑚𝑠 = 3040 𝑏𝑖𝑡𝑠⁄seg

**11) Tiempo de transmisión de una imagen de 640x280 pixeles**

Cantidad de pixeles en la imagen:

𝑃𝑖𝑥𝑒𝑙𝑒𝑠 = 640 ∗ 480 𝑝𝑖𝑥𝑒𝑙𝑒𝑠 = 307200 𝑝𝑖𝑥𝑒𝑙𝑒𝑠

Cantidad de información de cada pixel:

I(pixel) = log2 (1/ ) = 8 bits

El total de la informacion de la imagen seria

I(imagen) = 307200 pixeles \* 8 bits= 2.457.600 bits

Si el canal de transmisión tiene una tasa de transmisión de 1Mbps, entonces:

1𝑀𝑏𝑝𝑠 = 1.000.000 𝑏its

R = I/T => T = I/R = 2.456.600bits / 1.000.000 bits/seg =2.4576 seg

Por lo tanto la imagen tardara en transmitirse 2,4576 seg.

**12) Sensor de temperatura de una fabrica**

La probabilidad de ocurrencia de cada valor del sensor sera de p(valor) = 1/8 y la información de cada valor será de I(valor) = log2 8 = 3bits.

Si el medidor toma medidas cada segundo, durante una hora habrá tomado 3600 mediciones, por lo que la información del sensor después de una hora será:

I(sensor) = 3600 \* 3 bits = 10800 bits

Si se posee un canal de transmisión de 10kbps:

10 kbps = 10.000 bps

T = 10800 bits / 10.000bps = 1.08 seg

Por lo tanto se tardaría en transmitir 1.08 seg

**13) Fuente de memoria nula S = {z, x} con p(z) = ¾; p(x) = ¼**

Un código compacto binario para dicha fuente estaría dado por C = {0, 1}

La longitud media del código seria: L = ¾ . 1 + ¼ . 1 = 1 bit

La entropía de la fuente seria: H(S) = ¾ log2 (4/3) + ¼ log2 4 = 0.81 bits

Entonces el rendimiento del código estaría dado por:

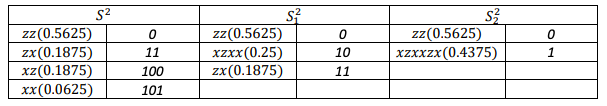
n = 0.81 bits / 1bits = 0.81

Extensión de 2do orden

S2 = {zx, zz, xx, xz} con P(S2) = {3/16; 9/16; 1/16; 3/16}

Codigo compacto binario

Orden según probabilidad de ocurrencia: 𝑧𝑧(0.5625), 𝑧𝑥(0.1875), 𝑥𝑧(0.1875), 𝑥𝑥(0.0625)



Por lo tanto, un código compacto binario para dicha fuente estaría dado por C = {11, 0, 101, 100}

La longitud media del código seria:

L = 3/16 . 2 + 9 /16 . 1 + 1 /16 . 3 + 3/16 . 3 = 27 /16 bit = 1,6875 bit

La entropía de la fuente seria:

H(S2) = 3/16 log2 (16/3) + 9/16 log2 (16/9) + 1/16 log2 (16) + 3/16 log2 (16/3) = 1,62 bits

Entonces, el rendimiento del código estaría dado por:

n = 1.62 bits / 1.6875 bits = 0.96

Extension de 3er orden

S3 = {𝑧𝑧𝑧, 𝑧𝑧𝑥, 𝑧𝑥𝑧, 𝑧𝑥𝑥, 𝑥𝑧𝑧, 𝑥𝑧𝑥, 𝑥𝑥𝑧, 𝑥𝑥𝑥} 𝑐𝑜𝑛 𝑃(𝑆3) = { 27/64 ; 9/64 ; 9/64 ; 3/64 ; 9/64 ; 3/64 ; 3/64 ; 1/64}

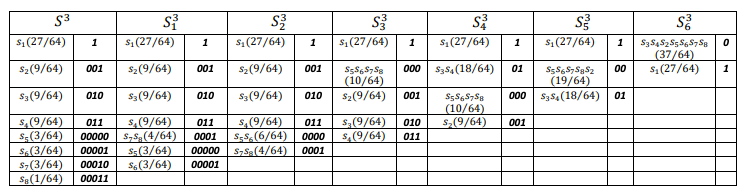
Codigo compacto binario

Orden según probabilidad de ocurrencia

𝑠1 = 𝑧𝑧𝑧 ( 27/64) ; 𝑠2 = 𝑧𝑧𝑥 ( 9/64) ; 𝑠3 = 𝑧𝑥𝑧 ( 9/64) ;

𝑠4 = 𝑥𝑧𝑧 ( 9/64) ; 𝑠5 = 𝑧𝑥𝑥 ( 3/64) ; 𝑠6 = 𝑥𝑧𝑥 ( 3/64) ;

𝑠7 = 𝑥𝑥𝑧 ( 3/64) ; 𝑠8 = 𝑥𝑥𝑥 ( 1/64);



Por lo tanto, un codigo compracto binario para dicha fuente estaría dado por:

𝐶 = {1,001,010,00000,011,00001,00010,00011}

La longitud media del código seria:

𝐿 = 27/64 ∙ 1 + 9/64 ∙ 3 + 9/64 ∙ 3 + 3/64 ∙ 5 + 9/64 ∙ 3 + 3/64 ∙ 5 + 3/64 ∙ 5 + 1/64 ∙ 5 =

= 79/32 𝑏𝑖𝑡 = 2.46875 𝑏𝑖𝑡

La entropía de la fuente seria:

H(S3)= 27/64 log2 ( 64/27) + 9/64 log2 ( 64/9 ) + 9/64 log2 ( 64/9 ) + 3/64 log2 ( 64/3 ) + 9/64 log2 ( 64/9 ) + 3/64 log2 ( 64/3 ) + 3/64 log2 ( 64/3 ) + 1/64 log2 (64) ≅ 2.43 𝑏𝑖𝑡s

Entonces, el rendimiento del código estaría dado por:

n = 2.4338 bits / 2.46785 bits = 0,986

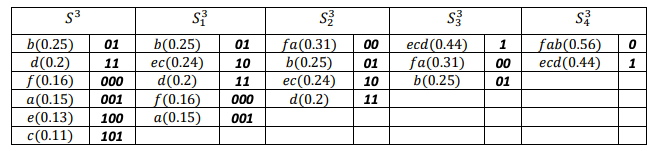
**14) Fuente de memoria nula S = {a, b, c, d, e, f}**

* P (𝑎) = 0.15
* P (𝑏) = 0.25
* P (𝑐) = 0.11
* P (𝑑) = 0.2
* P (𝑒) = 0.13
* P (𝑓) = 0.01 + P (a) = 0.16

Codigo compacto binario

Orden según probabilidad de ocurrencia:

b(0.25) > d(0.2) > f(0.16) > a(0.15) >e(0.13) > c(0,11)

Por lo tanto, un codigo compacto binario para dicha fuente estaría dado por:

C = {001,01,101,11,100,000}

Calcular la longitud media del codigo obtenido.

La longitud media del codigo seria: S = {a, b, c, d, e, f}

L = 0.15 ∙ 3 + 0.25 ∙ 2 + 0.11 ∙ 3 + 0.2 ∙ 2 + 0.13 ∙ 3 + 0.16 ∙ 3 = 2.55 𝑏𝑖𝑡

Calcular la entropía de la fuente.

La entropía de la fuente seria:

𝐻(𝑆) = 0.15 ∙ log2 ( 1/0.15) + 0.25 ∙ log2 ( 1/0.25) + 0.11 ∙ log2 ( 1/0.11) + 0.2 ∙ log2 ( 1/0.2 ) + 0.13 ∙ log2 ( 1/0.13) + 0.16 ∙ log2 ( 1/0.16) ≅ 2.531 bits

Calcular el rendimiento del codigo.

El rendimiento del codigo estaría dado por:

N = 25.531 bits / 2.55 bits = 0.993