SISTEMAS DE NUMERACION

Digital: todas las máquinas de computo usan dígitos para realizar sus procesos (operaciones aritméticas, escribir texto, escuchar música, ver videos, etc).

Tabla 1. Resumen de sistemas numéricos

	Decimal	Binario	Octal	Hexadecimal
Base	10	2	8	16
Dígitos (y caracteres)	0,1,…,8,9	0,1 (Bits: Binary Digit)	0,1,…,6,7	0,1,...,9,A,B,…,E,F
Polinomio	568.23	101.101	436.5	F1A3

Tabla 2. Digitos y caracteres de sistemas numericos

Decimal	Binario	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2		2	2
3		3	3
4		4	4
5		5	5
6		6	6
7		7	7
8			8
9			9
			A
			B
			C
			D
			E
			F

Polinomios:

$$568,23 = (5 \times 10^2) + (6 \times 10^1) + (8 \times 10^0) + (2 \times 10^{-1}) + (3 \times 10^{-2})$$

$101.101 = 1x2^2 + 0x2^1 + 1x2^0 + 1x2^{-1} + 0x2^{-2} + 1x2^{-3}$

$$436.5_8 = 4 \cdot 8^2 + 3 \cdot 8^1 + 6 \cdot 8^0 + 5 \cdot 8^{-1}$$

$$F1A3_{16} = 15 \cdot 16^3 + 1 \cdot 16^2 + 10 \cdot 16^1 + 3 \cdot 16^0$$

Conversiones entre sistemas numéricos

Conversión de decimal a otras bases: uno de los métodos es mediante divisiones sucesivas.

De decimal a binario:

$$
\begin{aligned}
108_{10} \div 2 &= 54 \text{ residuo } 0 \quad \text{(LSB)} \\
\div 2 &= 27 \text{ residuo } 0 \\
\div 2 &= 13 \text{ residuo } 1 \\
\div 2 &= 6 \text{ residuo } 1 \\
\div 2 &= 3 \text{ residuo } 0 \\
\div 2 &= 1 \text{ residuo } 1 \\
\div 2 &= 0 \text{ residuo } 1 \quad \text{(MSB)}
\end{aligned}
$$

$$108_{10} = 1101100_2$$

Conversión de números con parte fraccionaria:
la conversión de la parte fraccionaria se realiza multiplicándola sucesivamente por 2, la parte entera de los productos obtenidos solo deben ser ceros o unos y corresponden al binario.

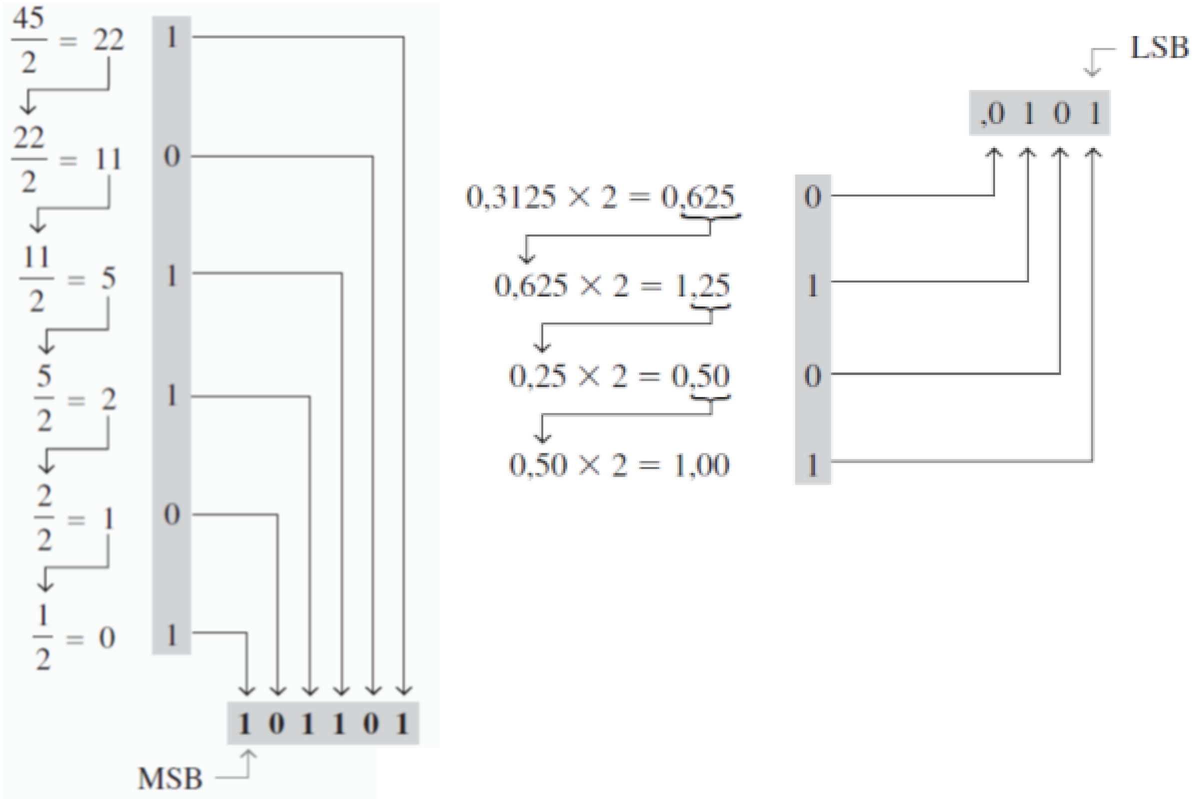
En esta conversión se pueden presentar tres casos:

a) la parte fraccionaria en algún momento es cero, en ese momento termina la conversión.

b) la parte fraccionaria en ningún momento es cero pero comienza a repetirse, en ese momento termina la conversión.

c) la parte fraccionaria no es cero ni tampoco se repite, se deben hacer las operaciones hasta considerar que el resultado obtenido es muy cercano a la parte fraccionaria del decimal.

Ejemplo: el decimal 45.3125 convertirlo a binario.



45.3125 = 101101.0101, este resultado corresponde al caso a).

Les recomiendo el video sobre el sistema binario: “Sistema binario (clase 12 parte 1)”.

Ejemplos

Tocci: páginas 54 y 55, ejemplo 2.1.

Floyd: página 61, ejemplo 2.6, página62.

Wakerly: página 57.

Ejercicios: dar solución a algunos de los siguientes problemas.

Tocci: preguntas de repaso (página 57) 1. Y 2., problema al final del capítulo: 2-2.

Floyd: problema relacionado (página 62), revisión de la sección 2.3 (página 63) , problemas al final del capítulo: 11 a 14.

Wakerly: problemas al final del capítulo: 2.6 a), c) y e).

De decimal a octal:

$108_{10} \div 8 = 13$ residuo 4 (dígito menos significativo)

$\div 8 = 1$ residuo 5

$\div 8 = 0$ residuo 1 (dígito más significativo)

$108_{10} = 154_8$

Ejemplos

Floyd: página 91.

Wakerly: página 57.

Ejercicios: dar solución a algunos de los siguientes problemas.

Floyd: revisión de la sección 2.9 (página 92) 2., problema al final del capítulo: 42.

Wakerly: problemas al final del capítulo: 2.6 b), d) y e).

De decimal a hexadecimal:

$$\begin{aligned} 108_{10} \div 16 &= 6 \text{ residuo } 12 \quad \text{(dígito menos significativo)} \\ \div 16 &= 0 \text{ residuo } 6 \quad \text{(dígito más significativo)} \\ 108_{10} &= 6C_{16} \end{aligned}$$

Les recomiendo el video sobre el sistema hexadecimal: “Como entender y trabajar con sistema hexadecimal (clase 12 parte 2)”.

Ejemplos.

Tocci: ejemplo 2-3.

Floyd: ejemplo 2.28.

Wakerly: página 57.

Ejercicios: dar solución a algunos de los siguientes problemas.

Tocci: preguntas de repaso (página 61) 2., problemas al final del capítulo: 2-5,10, 12 y 18.

Floyd: Problema relacionado (página 86), revisión de la sección 2.8 (página 89) 4., problema al final del capítulo: 38.

Wakerly: problemas al final del capítulo: 2.6 f), h) y j).

Conversión de cualquier base a decimal: se plantea el polinomio y se realizan las operaciones, el resultado es el decimal.

De binario a decimal:

$$\begin{aligned} 10111011001_2 = {} & 1 \cdot 1024 + 0 \cdot 512 + 1 \cdot 256 + 1 \cdot 128 + 1 \cdot 64 \\ & + 0 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 0 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 = 1497_{10} \end{aligned}$$

Ejemplos

Tocci: página 54.

Floyd: ejemplos 2.3 y 4.

Wakerly: página 54.

Ejercicios: dar solución a algunos de los siguientes problemas.

Tocci: preguntas de repaso (página 54), problemas al final del capítulo: 2-1 y 3.

Floyd: problemas relacionados (página 59), revisión de la sección 2.2 (página 60), problemas al final del capítulo: 5 a 7.

Wakerly: problemas al final del capítulo: 2.5 a), c) y e).

De octal a decimal:

$$1234_8 = 1 \cdot 512 + 2 \cdot 64 + 3 \cdot 8 + 4 \cdot 1 = 668_{10}$$

Ejemplos

Floyd: página 90.

Wakerly: página 56.

Ejercicios: dar solución a alguno de los siguientes problemas.

Floyd: revisión de la sección 2.9 (página 92) 1., problema al final del capítulo: 41.

Wakerly: problemas al final del capítulo: 2.5 b), d) e i).

De hexadecimal a decimal:

$$C0DE_{16} = 12 \cdot 4096 + 0 \cdot 256 + 13 \cdot 16 + 14 \cdot 1 = 49374_{10}$$

Ejemplos

Tocci: página 58.

Floyd: ejemplos 2.26 y 27.

Wakerly: página 56.

Ejercicios: dar solución a algunos de los siguientes problemas.

Tocci: preguntas de repaso (página 61) 1., problemas al final del capítulo: 2-4 y 11.

Floyd: problemas relacionados (página 85), revisión de la sección 2.8 (página 89) 3., problema al final del capítulo: 37.

Wakerly: problemas al final del capítulo 2.5 f), h) y j).

Conversión entre binario, octal y hexadecimal

Se usa la relación existente entre octal y binario y entre hexadecimal y binario

Octal	Binario
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Hexadecimal	Binario
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

Conversión de binario a octal

$$10111011001_2 = 10\ 111\ 011\ 001_2 = 2731_8$$

Ejemplos

Floyd: ejemplos 2.32 y 31

Wakerly: páginas 54, 55 y 58.

Ejercicios: dar solución a algunos de los siguientes problemas.

Floyd: problemas relacionados (página 92), revisión de la sección2.9 (página 92) 4. Y 3., problemas al final del capítulo: 44 y 43.

Wakerly: Problemas al final del capítulo: 2.1 g), i), b) y d), 2.2.

Conversión de binario a hexadecimal

$$10111011001_2 = 101\ 1101\ 1001_2 = 5D9_{16}$$

Ejemplos

Tocci: página 59.

Floyd: ejemplos 2.24 y 25.

Wakerly: páginas 54, 55 y 58.

Ejercicios: dar solución a algunos de los siguientes problemas.

Tocci: preguntas de repaso (página 61) 3. y 5., problemas al final del capítulo: 2.7, 6 y 16.

Floyd: problemas relacionados (página 84), revisión de la sección 2.8 (página 89) 1. Y 2., problemas al final del capítulo: 36 y 35.

Wakerly: problemas al final del capítulo: 2.1 a), c), e), f), h) y j), 2.3.

Conversión de octal a hexadecimal y hexadecimal a octal

$$1234_8 = 001\ 010\ 011\ 100_2 = 0010\ 1001\ 1100_2 = 29C_{16}$$

$$C0DE_{16} = 1100\ 0000\ 1101\ 1110_2 = 1\ 100\ 000\ 011\ 011\ 110_2 = 140336_8$$

Ejemplos

Wakerly: Página 58.

Ejercicios: dar solución a algunos de los siguientes problemas.

Wakerly: problemas al final del capítulo: 2.2, 3 , 16, de este último el f) es $\sqrt{14} = 5$ y 17.

Ejemplo: solución del problema 2.16 f).

La operación 41/3 = 13 es correcta en por lo menos un sistema numérico. Calcular las posibles bases del número en dicha operación.

41/3 = 13, 41 = 3(13) = 39, en decimal no es posible porque no se cumple la igualdad. Tentativamente se puede pensar que la base es 5 por los dígitos presentes en la operación (hasta el 4), pero se plantea la solución cambiando el producto por la suma asi: 41 = 13 +13 + 13, luego se escribe el polinomio de cada miembro de la ecuación proponiendo por ejemplo para la base desconocida la variable b: $4b^1 + 1b^0 = 1b^1 + 3b^0 + 1b^1 + 3b^0 + 1b^1 + 3b^0$, se realizan primero las operaciones con las potencias y se obtiene: $4b + 1 = b + 3 + b + 3 + b + 3$, luego $4b + 1 = 3b + 9$; $4b – 3b = 9 - 1$; $b = 8$.

Para garantizar que la base es solamente 8, se debe hacer la prueba, y asumiendo que posiblemente hubo error en el procedimiento, se ejecuta la prueba para las bases 7, 8 y 9.

Para base 7: se debe obtener que al sumar 3 veces 13 el resultado sea 41; 13 +13 = 26, no hay problema porque ambos son dígitos en dicha base, 26 + 13 = 39, no es posible porque 9 no es dígito en dicha base, por lo tanto, al 9 se le resta la base y el uno que sobra se pasa a la siguiente columna, 9 -7 = 2, entonces 26 +13 +1 = 42, esto implica que 13 +13 +13 = 42 no cumple con el resultado de 41.

Para la base 8: 13 +13 = 26; 26 + 13 = 39; 9 – 8 = 1, entonces 26 + 13 + 1 = 41, esto implica que 13 +13 +13 = 41, se cumple el resultado.

Para la base 9 realicen ustedes dicha operación.

Utilidad del sistema hexadecimal

Estudien uds. este tema.

CODIGOS BINARIOS

Es un conjunto, cadena o secuencia de bits que representan información.

De acuerdo con el trabajo que debe realizar el sistema de cómputo, el conjunto de bits se agrupa de una forma específica y por lo tanto se tienen diferentes códigos binarios:

Binario natural o directo.

Decimal Codificado en Binario (BCD).

Gray.

Otros.

Binario natural o directo

Es el mismo sistema de numeración binario, es un código ponderado (pesado) porque cada bit tiene un valor según la posición que ocupe, solo usa los bits que son necesarios para representar el decimal.

Ejemplos: 1 = 1; 2 = 10; 6 = 110; 10 = 1010; 18 = 10010.

Decimal Codificado en Binario (BCD)

Es una familia de códigos en los cuales cada dígito del número decimal se representa con un grupo de 4 bits. El más usado en la actualidad es el **BCD natural u 8421**, es un código ponderado porque cada bit del grupo tiene un valor según la posición que ocupa. Solo usa 10 de los 16 grupos (combinaciones) que se obtienen con 4 bits: 0000 a 1001. Los otros 6 grupos 1010 a 1111 son conocidos como combinaciones BCD no válidas.

Ejemplos: 1 = 0001; 2 = 0010; 6 = 0110; 10 = 0001 0000; 18 = 0001 1000.

Código Gray

Es conocido como código de cambio mínimo porque entre números sucesivos solo cambia un bit del grupo, no es ponderado. Solo usa los bits que son necesarios para representar el decimal.

Es usado en la industria metalmecánica, en el sistema de frenado de autos, en fotocopiadoras, para detectar la posición de ejes, etc.

Ejemplos: 1 = 1; 2 = 11; 6 = 101; 10 = 1111; 18 = 11011.

BYTE, NIBBLE Y PALABRA: estudiar estos temas.

Tabla 3. Códigos binarios

Decimal	Binario	BCD	GRAY
0	0	0000	0000
1	1	0001	0001
2	10	0010	0011
3	11	0011	0010
4	100	0100	0110
5	101	0101	0111
6	110	0110	0101
7	111	0111	0100
8	1000	1000	1100
9	1001	1001	1101
10	1010	0001 0000	1111
11	1011	0001 0001	1110
12	1100	0001 0010	1010
13	1101	0001 0011	1011
14	1110	0001 0100	1001
15	1111	0001 0101	1000

Conversión de binario a Gray

El decimal se convierte a binario natural y la conversión se inicia por la izquierda, el MSB del binario corresponde al bit de la izquierda del Gray, el siguiente bit del Gray se obtiene comparando el MSB con el siguiente bit a la derecha del binario, si los bits son iguales el bit Gray resultante es **cero**, si son diferentes el bit es **uno** y así con los otros bits. Comparar las columnas binario y Gray de la tabla anterior.

Conversión de Gray a binario

Estudien ustedes este tema.

Ejemplos de códigos binarios.

Tocci: páginas 62 y 63, ejemplos 2-6 a 12.

Floyd: ejemplos 2.33, 34 y 37.

Ejercicios: dar solución a algunos de los siguientes problemas.

Tocci: preguntas de repaso (páginas 62, 65 y 67), problemas al final del capítulo; 2-19 a 23.

Floyd: problemas relacionados (páginas 93, 94 y 97), revisión de la sección 2.10 (página 96) 1. a 3., problemas al final del capítulo: 45 a 49, 54 y 55.

Códigos alfanuméricos

Son códigos binarios que representan dígitos, letras, signos de puntuación, algunas instrucciones, etc. necesarios para la transferencia de información en las máquinas de computo.

Código ASCII

Es el código alfanumérico más usado. Usa cadenas de 7 bits y por lo tanto puede representar 128 elementos diferentes.

Los 7 bits se designan $b_7b_6b_5b_4b_3b_2b_1$, de los cuales los bits $b_7b_6b_5$ se conocen como grupo MSB con b_7 como el bit MSB, los bits $b_4b_3b_2b_1$ son el grupo LSB con b_1 como el bit LSB.

Este código codifica 94 caracteres gráficos que pueden imprimirse y 34 elementos no imprimibles que se usan como diversas funciones de control. Los caracteres imprimibles consisten en las letras mayúsculas y minúsculas (26 de cada una), los 10 dígitos decimales (0 a 9) y 32 caracteres correspondientes a los signos de puntuación y símbolos especiales.

Tabla 4. Código ASCII

	$b_7 b_6 b_5$							
$b_4 b_3 b_2 b_1$	**000**	**001**	**010**	**011**	**100**	**101**	**110**	**111**
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	“	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	‘	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	−	=	M	]	m	}
1110	SO	RS	.	>	N	∧	n	~
1111	SI	US	/	?	O	—	o	DEL

Caracteres de control

NUL	Nulo	DLE	Escape de enlace de datos
SOH	Inicio de cabecera	DC1	Control de dispositivo 1
STX	Inicio de texto	DC2	Control de dispositivo 2
ETX	Fin de texto	DC3	Control de dispositivo 3
EOT	Fin de transmisión	DC4	Control de dispositivo 4
ENQ	Pregunta	NAK	Acuse negativo
ACK	Acuse	SYN	Inactividad sincrónica
BEL	Campana	ETB	Fin de bloque de transmisión
BS	Retroceso	CAN	Cancelar
HT	Tabulación horizontal	EM	Fin de medio
LF	Salto de línea	SUB	Sustituto
VT	Tabulación vertical	ESC	Escape
FF	Salto de página	FS	Separador de archivos
CR	Retorno de carro	GS	Separador de grupos
SO	Desplazamiento afuera	RS	Separador de registros
SI	Desplazamiento adentro	US	Separador de unidades
SP	Espacio	DEL	Borrar

También hay un código ASCII extendido de 8 bit en el cual los 128 elementos adicionales representan caracteres alfabéticos no ingleses, símbolos de monedas no inglesas, símbolos matemáticos, caracteres para generar gráficos, etc.

Les recomiendo el video sobre códigos binarios: “Como trabaja un circuito digital con letras y números, códigos numéricos y alfabéticos (clase 13)”.

código de barras

CODIGO DE BARRAS (UPC)

0 7 1 | 8 9 2 6 | 4 8 7 8 4 | 8

País de Origen | Código del fabricante | Identificación del producto | Número de control

El código de barras es un grupo rectangular de líneas paralelas, con números impresos en la parte inferior, utilizado para identificar productos de todo tipo (ropa, alimentos, revistas, etc.) en cualquier parte del mundo. Este código proporciona información acerca del país de origen, el fabricante, el peso, etc. del producto.

El código de barras se denomina también código universal de productos o UPC (universal product code). Sin importar su complejidad, un UPC está formado por una sucesión de barras delgadas y gruesas (1's y 0's) que representan un número de trece (13) cifras. Este número es algo así como la cédula de ciudadanía del producto.

Las tres primeras cifras identifican el país de origen y las cuatro siguientes el fabricante. Las cinco cifras restantes son propias del producto y especifican características como referencia, peso, talla, precio, etc. Con estas cinco cifras, el fabricante puede distinguir hasta cien mil artículos. La última cifra se utiliza para efectos de control.

En los supermercados y establecimientos comerciales donde se utiliza el sistema de identificación UPC, el código de barras del producto se hacer pasar por un escáner o lector óptico situado en el punto de pago. Los circuitos digitales del escáner interpretan la información recibida y la transmiten a un computador central.

El computador central se programa para procesar la información anterior, de acuerdo con las necesidades del comerciante o del fabricante. Por ejemplo, puede utilizarse para agilizar la facturación, mantener actualizado los inventarios u

También se ha propuesto su utilización en la marcación de billetes, con el fin de poder seguirles el rastro. Esta medida contrarrestaría el llamado lavado de dólares. El código de barras se puede también emplear para identificar papeles como boletas, tiquetes, documentos, etc. Y detectar su autenticidad.

El principal promotor del código de barras a nivel mundial es la Asociación Europea de Numeración de Artículos (EAN). Este organismo cuenta con filiales en varios países, encargadas de asignar localmente el código UPC a fabricantes y comerciantes. Un ejemplo es el Instituto Colombiano de Codificación y Automatización Comercial (IAC).

Método para detectar y corregir errores

Un error en un sistema digital es la alteración de datos (bits) a partir de su valor correcto. La principal causa de errores en la transmisión es el **ruido eléctrico** el cual consiste en una señal eléctrica indeseable que produce fluctuaciones en el voltaje o la corriente, esas señales indeseables están presentes en todos los sistemas eléctricos en diversos grados.

Método de paridad para detección de errores

Uno de los esquemas más simples usados para este fin es el **método de paridad** que consiste en el **bit de paridad** el cual es un bit que se agrega al grupo de bit del código que se está transmitiendo. Dicho bit de paridad se hace 0 o 1 dependiendo del número de **unos** que contenga el grupo. De acuerdo con eso hay 2 métodos distintos.

Método de paridad par

obtener datos por memorizados sobre la demanda y preferencia de los consumidores con relación a ciertos productos.

El código de barras es tan eficiente que se utiliza incluso para identificar equipaje de pasajeros en aeropuertos de mucho tráfico. Si una maleta se extravía en un viaje y aparece en otra parte del mundo, al pasar las barras por un lector óptico la aerolínea podrá identificar automáticamente a quién le corresponde y en qué ruta se extravió.

El valor del bit de paridad en el transmisor se elige de modo que el número total de unos en el grupo, incluido el bit de paridad **sea par**, normalmente el bit de paridad se coloca en la izquierda del grupo como MSB.

Método de paridad impar

Similar al anterior pero el valor del bit de paridad se elige de modo que el número total de unos en el grupo **sea impar**. En cualquiera de los dos casos el bit de paridad se convierte en parte del código. El método de paridad solo detecta errores impares de bits, pero está demostrado estadísticamente que en grupos de pocos bits normalmente solo ocurre un error.

Método para la corrección de errores: Código de Hamming

Estudiar uds. este tema.

Ejemplos

Tocci: páginas 70 y 71, ejemplos: 2-13 a 15.

Floyd: ejemplos: 2.38 a 44.

Ejercicios: darle solución a alguno de los siguientes problemas.

Tocci: preguntas de repaso (páginas 63, 69 y 72), problemas al final del capítulo: 2-2 a 29.

Floyd: Problemas relacionados (páginas 101,105,108 a 111), revisión de la sección 2.11 (página 103), sección 2.12 (página 111), problemas al final del capítulo: 56 a 63.