

TÉCNICAS DE PROGRAMACIÓN - TP

Mg. Miguel Guanira

Mg. Silvia Vargas

Capítulo 1: Almacenamiento de Datos en el Computador

- ▶ 1.1 Introducción
- ▶ 1.2. Memoria del Computador: Principal y Secundaria
- ▶ 1.3. ¿Cómo se almacena la información en el computador?
- ▶ 1.4. Representación de diversos tipos de información:
 - ▶ a) números enteros
 - ▶ b) números negativos
 - ▶ c) números reales
 - ▶ d) almacenamiento invertido o “back-words”
 - ▶ e) caracteres
 - ▶ f) cadena de caracteres
 - ▶ g) imágenes

1.1. Introducción



Programa (Definición RAE):

Conjunto unitario de instrucciones que permite a una computadora realizar funciones diversas, como el tratamiento de textos, el diseño de gráficos, la resolución de problemas matemáticos, el manejo de bancos de datos, etc.

Imagen tomada de: <https://computerhoy.com/reportajes/tecnologia/historia-lenguajes-programacion-428041>

1.2. Memoria del Computador



Memoria Principal

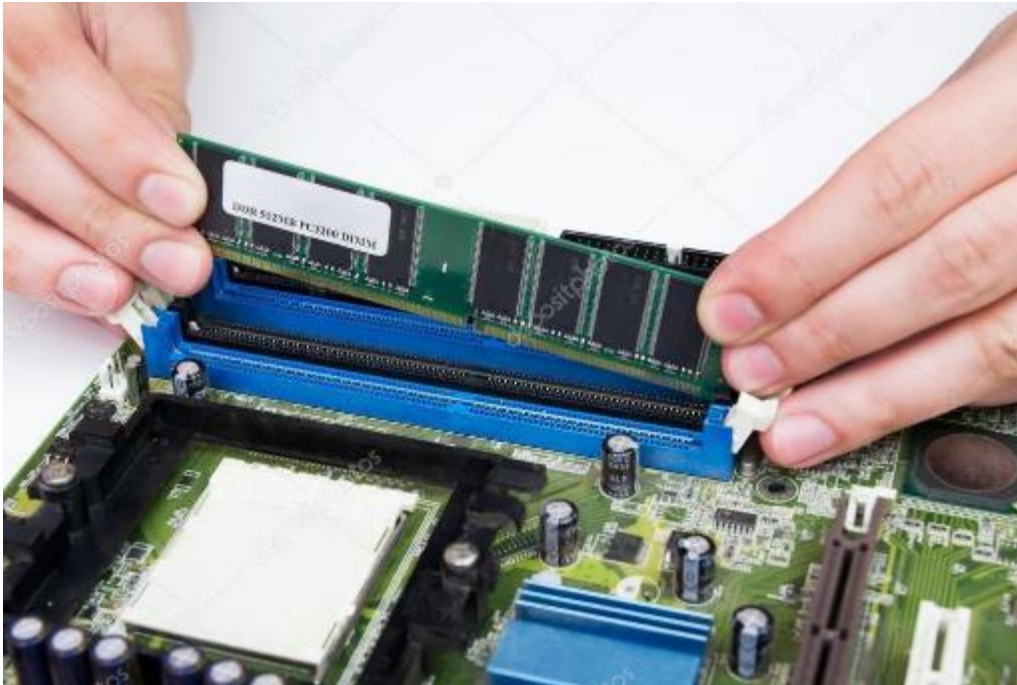


Memoria Secundaria

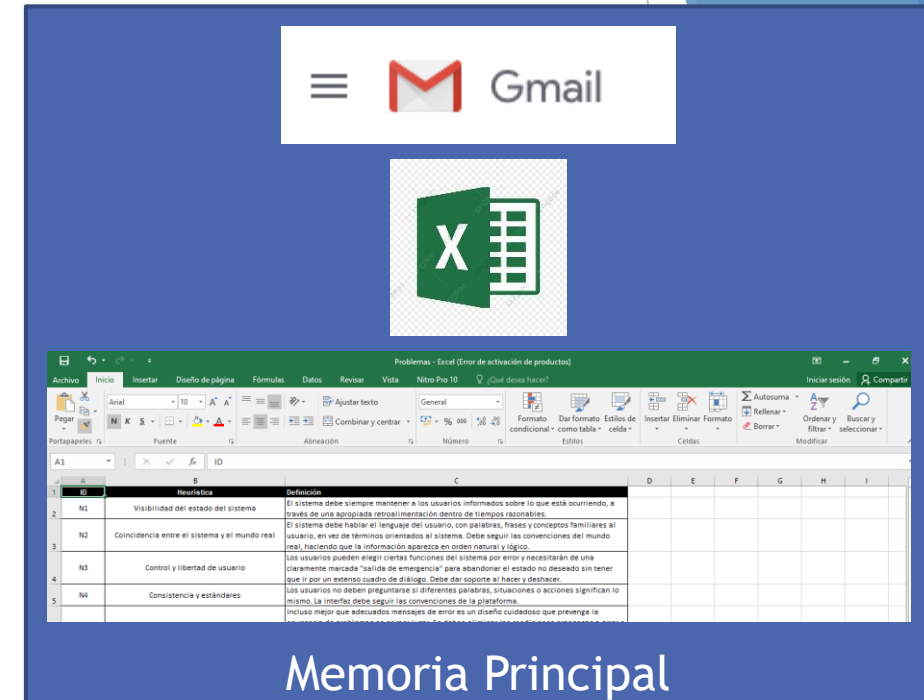


Imágenes tomada de: <https://www.mobitechzone.com/how-to-get-android-apps-on-a-chromebook/>
<https://thetecheducation.com/what-is-the-software-like-on-chromebook/>
<https://sites.google.com/site/partesinternasdeuncomputador/disco-duro>
<https://www.kemik.gt/comprar/memoria-usb-sandisk-cruzer-blade-16-gb/>

1.2.1. Memoria Principal



<https://componentesinformaticospma.blogspot.com/2019/05/aspectos-que-influyen-en-la-compra-de.html>



- Se encuentra dentro del computador, donde se coloca el programa que se ejecuta.
- Volátil.

1.2.2. Memoria Secundaria

- Almacenamiento permanente.
- Para procesarla se debe colocar en la memoria principal.



Problemas - Excel (Error de activación de producto)

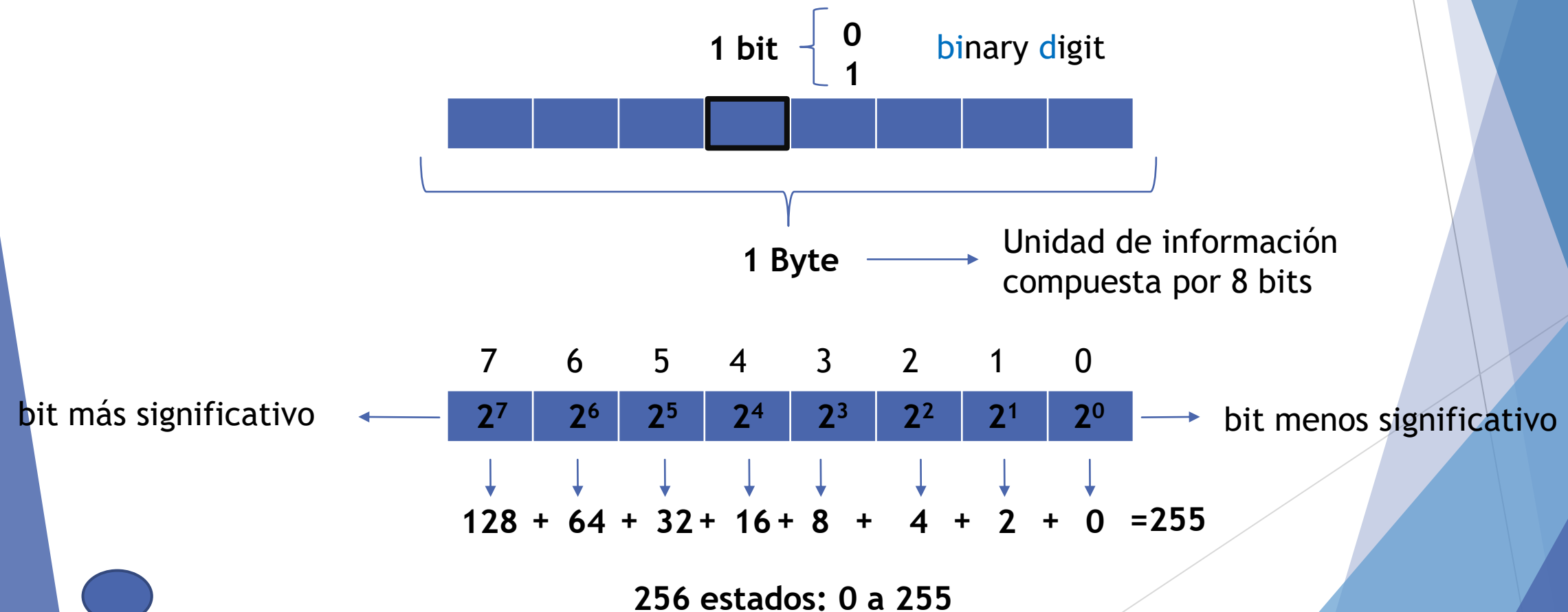
Heurística		Definición
N1	Visibilidad del estado del sistema	El sistema debe siempre mantener a los usuarios informados sobre lo que está ocurriendo, a través de una apropiada retroalimentación dentro de tiempos razonables.
N2	Coincidencia entre el sistema y el mundo real	El sistema debe hablar el lenguaje del usuario, con palabras, frases y conceptos familiares al usuario, en vez de términos orientados al sistema. Debe seguir las convenciones del mundo real, haciendo que la información aparezca en orden natural y lógico.
N3	Control y libertad de usuario	Los usuarios pueden elegir ciertas funciones del sistema por error y necesitarán de una claramente marcada "salida de emergencia" para abandonar el estado no deseado sin tener que ir por un extenso cuadro de diálogo. Debe dar soporte al hacer y deshacer.
N4	Consistencia y estándares	Los usuarios no deben preguntarse si diferentes palabras, situaciones o acciones significan lo mismo. La interfaz debe seguir las convenciones de la plataforma. Incluso mejor que adecuados mensajes de error es un diseño cuidadoso que prevenga la ocurrencia de situaciones de error.

Memoria Principal

<https://componentesinformaticospma.blogspot.com/2019/05/aspectos-que-influyen-en-la-compra-de.html>

<https://thetecheducation.com/what-is-the-software-like-on-chromebook/>

1.3. ¿Cómo se almacena la información en el computador?



1.4. Representación de diversos tipos de información

476

- 476

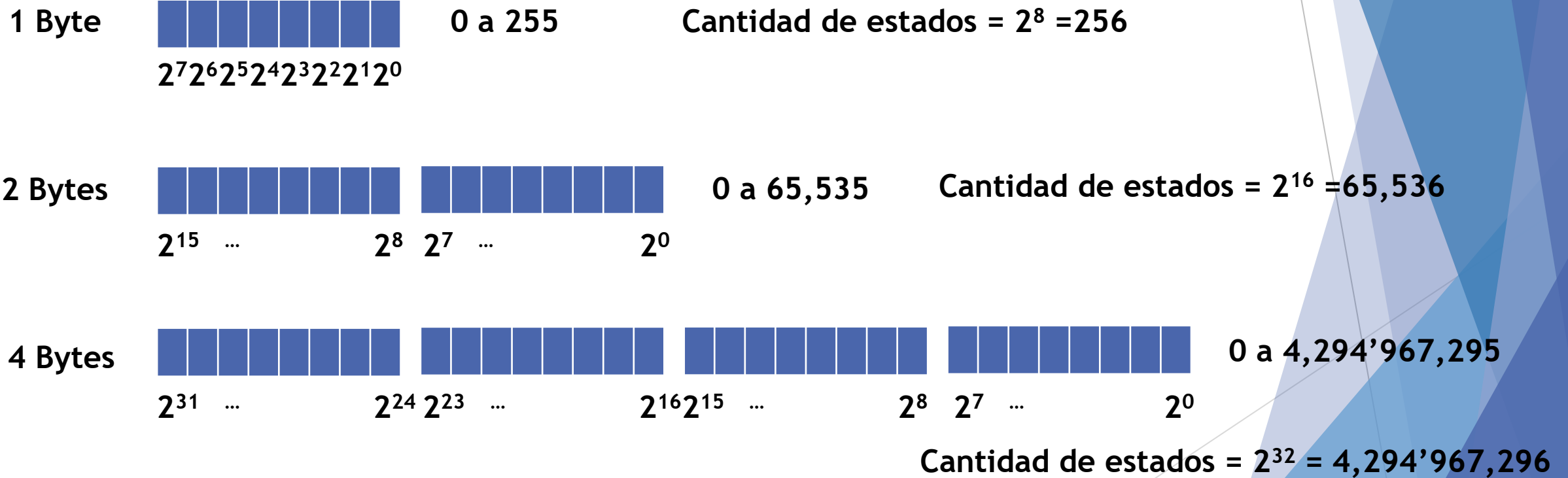
- 476.25

A a Z .

JUAN PÉREZ



a) números enteros



Reglas de suma

Suma de bits:

$$0 + 0 = 0$$

$$1 + 0 = 1$$

$$0 + 1 = 1$$

$$1 + 1 = 0, \text{ se acarrea } 1$$

Suma de bytes:

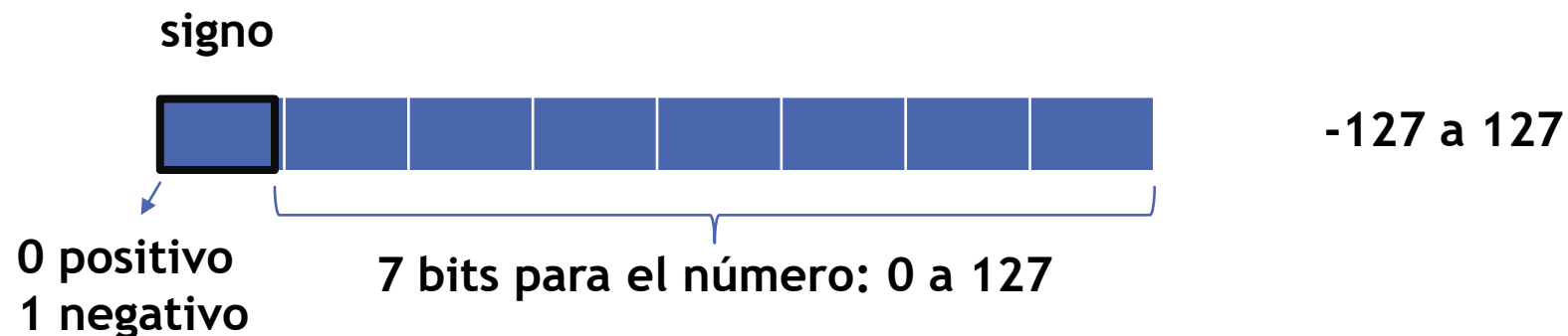
$$\begin{array}{r} 1) \quad 0000 \ 0000 \ + \\ \quad 0000 \ 0001 \\ \hline \quad 0000 \ 0001 \end{array}$$

$$\begin{array}{r} \quad \quad \quad 11 \\ 2) \quad 0000 \ 0011 \ + \\ \quad 0000 \ 0001 \\ \hline \quad 0000 \ 0100 \end{array}$$

$$\begin{array}{r} \quad \quad \quad 11 \\ 3) \quad 1111 \ 1111 \ + \\ \quad 0000 \ 0001 \\ \hline \quad 0000 \ 0000 \ , \text{ se acarrea } 1 \end{array}$$

b) números negativos ^(1/3)

Método 1: Empleo de un bit para representar el signo



1) $18 + 7 = 25$

Binary representation:

$$\begin{array}{r} 2^4 + 2^1 \\ 2^2 + 2^1 + 2^0 \end{array} \quad \begin{array}{r} 0001 \ 0010 \\ 0000 \ 0111 \end{array} \quad \begin{array}{r} 0001 \ 0010 + \\ 0000 \ 0111 \end{array}$$

$0001 \ 1001$

$2^4 + 2^3 + 2^0 = 25$ ✓

2) $18 + (-7) = 11$

Binary representation:

$$\begin{array}{r} 0001 \ 0010 + \\ 1000 \ 0111 \end{array}$$

$1001 \ 1001$

$2^4 + 2^3 + 2^0 = -25$ ✗

b) números negativos (2/3)

Método 2: Complemento

25 0001 1001
-25 1110 0110

5 0000 0101
-5 1111 1010

-127 a 127

1) 18 + 0001 0010 +
7 0000 0111

25 0001 1001

2) 18 + 0001 0010 +
-7 1111 1000

11 0000 1010 = 10

3) -5 + 1111 1010 +
-7 1111 1000

-12 1111 0010 = -13

0000 1101
 $2^3 + 2^2 + 2^0 = 13$

b) números negativos (3/3)

Método 3: Complemento a 2 = Complemento + 1

25 0001 1001
-25 1110 0111

5 0000 0101
-5 1111 1011

-127 a 127

1) 18 + 0001 0010 +
7 0000 0111

25 0001 1001 ✓

2) 18 + 0001 0010 +
-7 1111 1001

11 0000 1011 = 11 ✓

3) -5 + 1111 1011 +
-7 1111 1001

-12 1111 0100 = -12 ✓

1111 0011

0000 1100 = 12

c) números reales (1/2)

$$157.381_{(10)} = 1.57381 \times 10^2$$

$$0.000135_{(10)} = 1.35 \times 10^{-4}$$

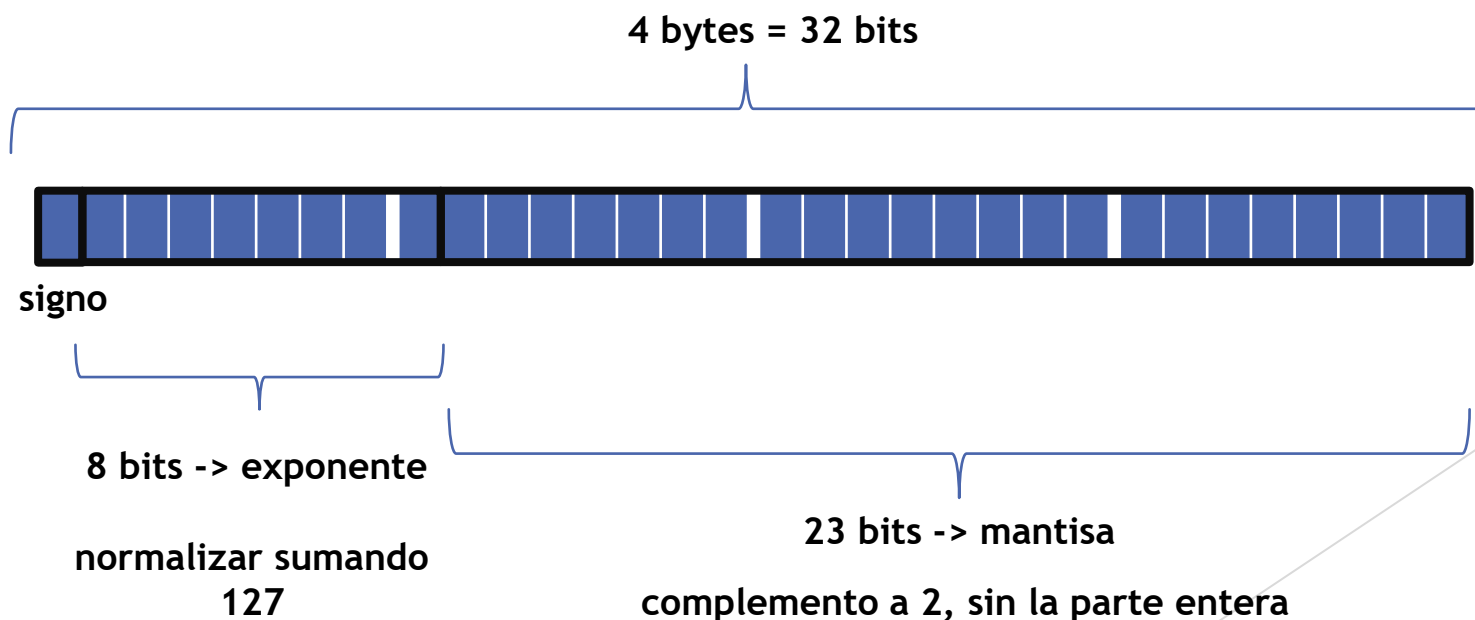
$$10101.1101_{(2)} = 1.01011101 \times 2^4$$

$$R = m \times b^e$$

$m = \text{mantisa}$

$b = \text{base}$

$e = \text{exponente}$



c) números reales (2/2)

$$0.875_{(10)} = 0.111_{(2)}$$

$$.2^{-1} 2^{-2} 2^{-3}$$

$$2^{-1} = 0.5$$

$$2^{-2} = 0.25 \quad +$$

$$2^{-3} = 0.125$$

$$0.875$$

$$734.875_{(10)} = 1011011110.111_{(2)}$$

$$7.34875 \times 10^2 = 1.011011110111 \times 2^9$$

Mantisa: 1.011011110111

Normalizando el Exponente

$$\text{Exponente} + 127 = 9 + 127 = 136_{(10)} = 10001000_{(2)}$$

0 1 0 0 0 1 0 0 0 0 1 1 0 1 1 1 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0

signo

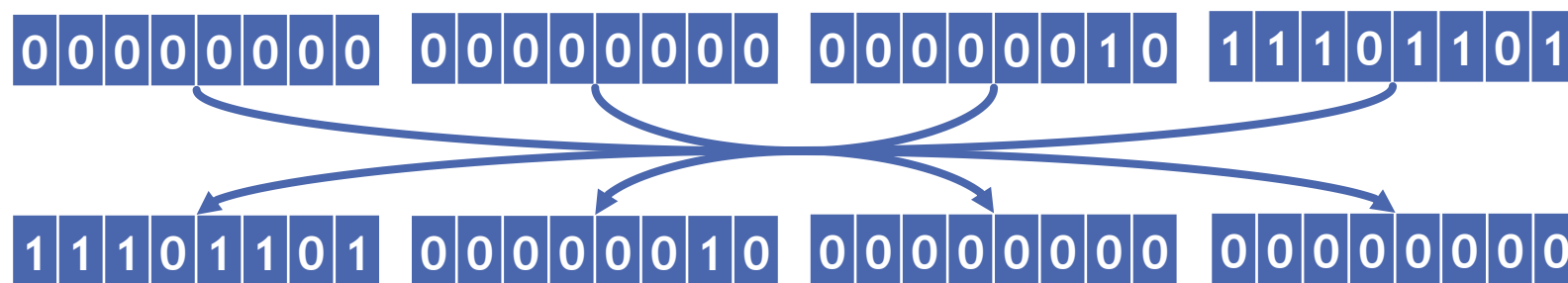
8 bits -> exponente

23 bits -> mantisa

d) almacenamiento Invertido - “back-words”

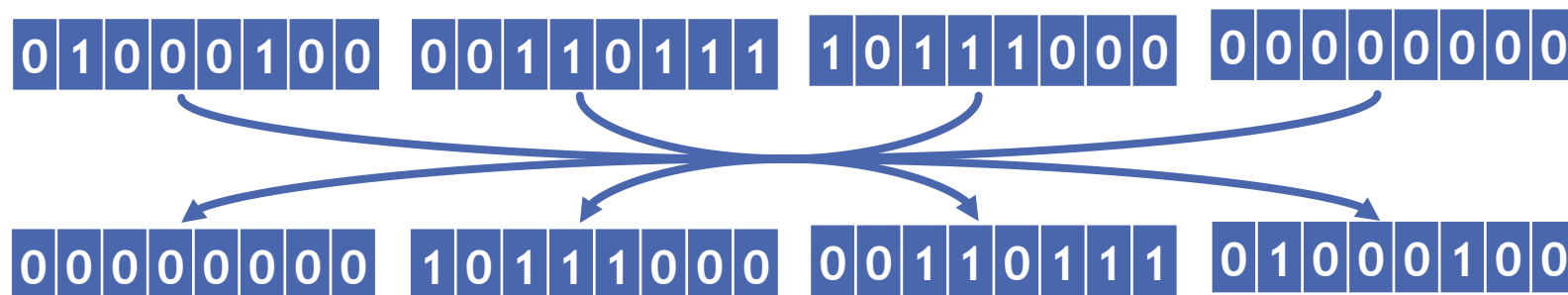
$\xrightarrow{234}$
 $\xleftarrow{234+567}$

749 se almacena como 947



back-words a nivel de bytes

743.875



back-words a nivel de bytes

CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER	CODINGO	HEX	CHARACTER
0	00		32	20		64	40	@	96	60		128	80	Ç	160	A0	à	192	C0	À	224	E0	à						
1	01	☺	33	21	!	65	41	A	97	61	a	129	81	ü	161	A1	á	193	C1	Á	225	E1	á						
2	02	☹	34	22	"	66	42	B	98	62	b	130	82	ë	162	A2	â	194	C2	Â	226	E2	â						
3	03	☹	35	23	#	67	43	C	99	63	c	131	83	ä	163	A3	ã	195	C3	Ã	227	E3	ã						
4	04	*	36	24	\$	68	44	D	100	64	d	132	84	å	164	A4	ä	196	C4	Ä	228	E4	ä						
5	05	+	37	25	%	69	45	E	101	65	e	133	85	ä	165	A5	å	197	C5	Å	229	E5	å						
6	06	*	38	26	&	70	46	F	102	66	f	134	86	ä	166	A6	æ	198	C6	Æ	230	E6	æ						
7	07	.	39	27	'	71	47	G	103	67	g	135	87	ç	167	A7	ç	199	C7	Ç	231	E7	ç						
8	08	=	40	28	(72	48	H	104	68	h	136	88	è	168	A8	è	200	C8	È	232	E8	è						
9	09	°	41	29)	73	49	I	105	69	i	137	89	é	169	A9	é	201	C9	É	233	E9	é						
10	0A	®	42	2A	*	74	4A	J	106	6A	j	138	8A	ê	170	AA	ê	202	CA	Ê	234	EA	ê						
11	0B	©	43	2B	+	75	4B	K	107	6B	k	139	8B	ÿ	171	AB	ÿ	203	CB	ÿ	235	EB	ÿ						
12	0C	☺	44	2C		76	4C	L	108	6C	l	140	8C	ÿ	172	AC	ÿ	204	CC	ÿ	236	EC	ÿ						
13	0D	☹	45	2D	-	77	4D	M	109	6D	m	141	8D	ÿ	173	AD	ÿ	205	CD	=	237	ED	ÿ						
14	0E	☺	46	2E	.	78	4E	N	110	6E	n	142	8E	À	174	AE	«	206	CE	»	238	EE	«						
15	0F	☹	47	2F	/	79	4F	O	111	6F	o	143	8F	À	175	AF	»	207	CF	±	239	EF	±						
16	10	▶	48	30	0	80	50	P	112	70	p	144	90	E	176	B0	±	208	D0	±	240	F0	±						
17	11	◀	49	31	1	81	51	Q	113	71	q	145	91	æ	177	B1	≡	209	D1	≡	241	F1	±						
18	12	☺	50	32	2	82	52	R	114	72	r	146	92	Æ	178	B2	≡	210	D2	≡	242	F2	±						
19	13	☹	51	33	3	83	53	S	115	73	s	147	93	ø	179	B3	☺	211	D3	☺	243	F3	±						
20	14	☺	52	34	4	84	54	T	116	74	t	148	94	ø	180	B4	☺	212	D4	☺	244	F4	☺						

ASCII: American Standard Code for Information Interchange

e) caracteres (2/3)

```
char letra=65;
```

```
cout<<letra<<endl; /*mostrará A en la pantalla*/
```

```
letra='A';
```

```
cout<<letra<<endl; /*mostrará A en la pantalla*/
```

```
letra+=10;
```

```
cout<<letra<<endl; /*mostrará K en la pantalla*/
```



e) caracteres (3/3)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
0010	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	CS	RS	US
0020		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
0030	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0040	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0050	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
0060	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
0070	p	q	r	s	t	u	v	w	x	y	z	{		}	~	¸
0080	XXX	XXX	SPH	NBH	IND	NEL	SSA	ESA	HTS	HTJ	VTB	PLD	PLU	NI	SS2	SS3
0090	DC5	PU1	PU2	STB	COH	MW	SPA	EPA	BOB	XXX	SCI	CSI	ST	OSC	PM	APC

TABLA DE CARACTERES UNICODE

<https://unicode-table.com/es/#control-character>

f) cadena de caracteres

- ▶ Arreglos
- ▶ Punteros, espacios de memoria gestionados dinámicamente

```
char cadena[40]="TECNICAS DE PROGRAMACION";
```

```
/*al finalizar de asignar estos caracteres se asigna el caracter 0 o '\0'*/  
/*este carácter indica el final de los caracteres válidos del arreglo*/
```

T	E	C	N	I	C	A	S		D	E		P	R	O	G	R	A	M	A	C	I	O	N	'\0'	...
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	
										1										2					

cadena[0]

cadena[23]

cadena[24]

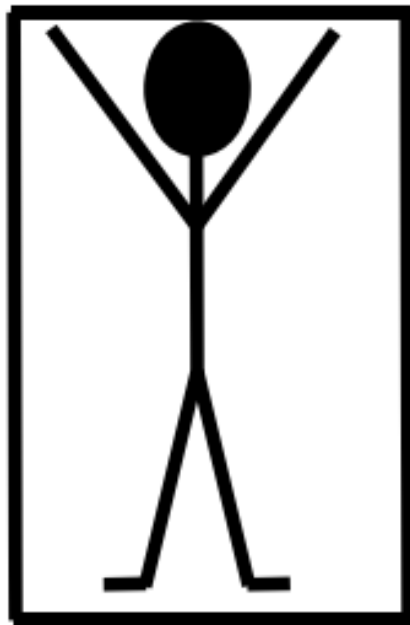
g) Imágenes (1/3)

- ▶ Píxel o Pixel: picture element

elemento más pequeño de los que componen una imagen digital

1 pixel- 1 bit:

- 0 blanco
- 1 negro



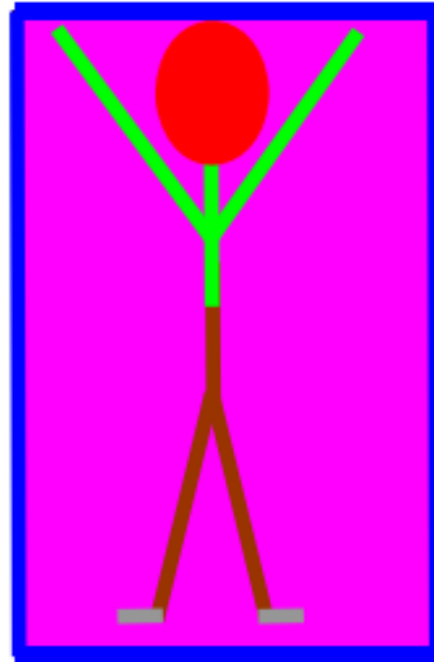
Valor de cada píxel: 1 píxel = 1 bit

[illegible]

g) Imágenes (2/3)

1 pixel- 3 bit

color	Valor decimal	Valor de cada pixel (3 bits)
blanco	0	000
azul	1	001
verde	2	010
fucsia	3	011
rojo	4	100
marrón	5	101
guinda	6	110
plomo	7	111

[illegible][illegible]

g) imágenes (3/3)

► Formato Imagen

- Cantidad de colores, número de bits por pixel
- número de píxeles por línea
- número de líneas

► Formato BMP

Bytes	Información
0, 1	Tipo de fichero "BM"
2, 3, 4, 5	Tamaño del archivo
6, 7	Reservado
8, 9	Reservado
10, 11, 12, 13	Inicio de los datos de la imagen
14, 15, 16, 17	Tamaño de la cabecera del bitmap
18, 19, 20, 21	Anchura (píxeles)
22, 23, 24, 25	Altura (píxeles)
26, 27	Número de planos
28, 29	Tamaño de cada punto
30, 31, 32, 33	Compresión (0=no comprimido)
34, 35, 36, 37	Tamaño de la imagen
38, 39, 40, 41	Resolución horizontal
42, 43, 44, 45	Resolución vertical
46, 47, 48, 49	Tamaño de la tabla de color
50, 51, 52, 53	Contador de colores importantes

¿Qué hemos aprendido hoy?

- ▶ ¿Qué es la memoria principal y la memoria secundaria? ¿Cuáles son sus principales características?
- ▶ ¿Qué se utiliza para almacenar la información en el computador?
- ▶ ¿Qué métodos se utilizan para representar números negativos?
- ▶ ¿Para qué se utilizan las tablas ASCII y UNICODE?
- ▶ ¿Cómo se puede almacenar una cadena de caracteres?
- ▶ ¿Qué es el pixel y como se utiliza para el almacenamiento de imágenes?
- ▶ ¿Para qué se usa el almacenamiento invertido “back-words”?