

2. Fondamenti del linguaggio digitale: sistemi di numerazione

Sicurezza dell'Informazione

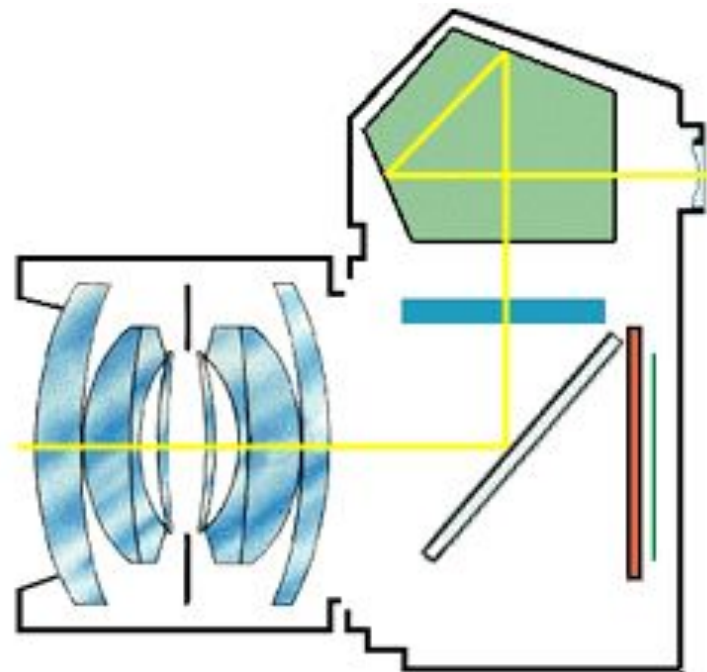
L'informazione nei computer



Viviamo in un mondo analogico, ma anche digitale. Tutti i video, immagini, audio e, più in generale, informazioni, vengono trasmesse ed elaborate in digitale.

Fotocamera analogica

Acquisizione: la luce proveniente dalla scena attraverso l'obiettivo e colpisce la pellicola fotografica, che è ricoperta da una sostanza chimica sensibile alla luce: in genere sali d'argento (alogeni d'argento) sospesi in una gelatina.



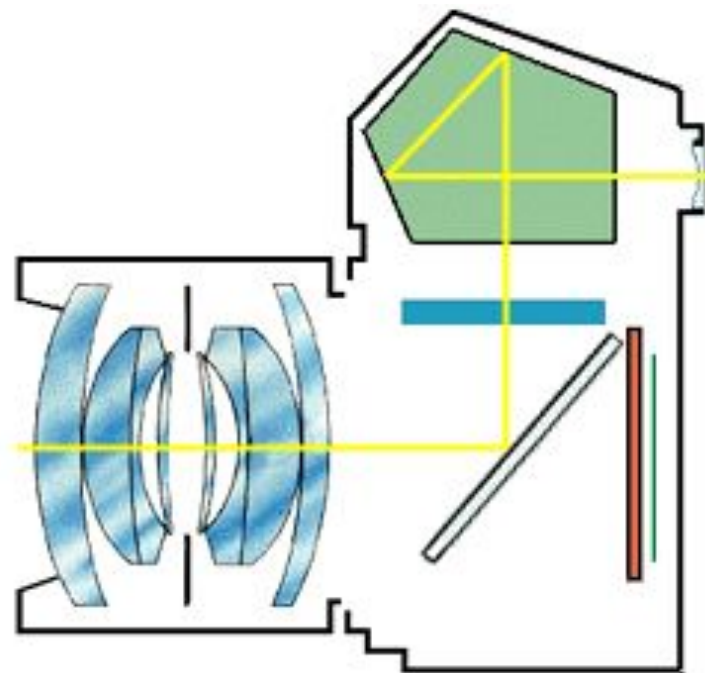
Fotocamera analogica

Reazione Chimica: Ogni punto della pellicola riceve una quantità diversa di luce (più o meno intensa a seconda della scena).

I cristalli di alogenuro d'argento reagiscono in proporzione alla luce ricevuta:

- Dove arriva molta luce, la sostanza si altera di più.
- Dove arriva poca luce, rimane quasi inalterata.

Si crea così un'immagine latente (cioè invisibile), che rappresenta in modo continuo l'intensità luminosa.



Fotocamera analogica

Sviluppo chimico: La pellicola viene poi sviluppata in camera oscura. I reagenti chimici fissano le parti colpite dalla luce e rimuovono quelle non esposte.

Ne risulta un negativo: le zone più illuminate diventano scure e viceversa.



Digitalizzazione

Creation of a Digital Image

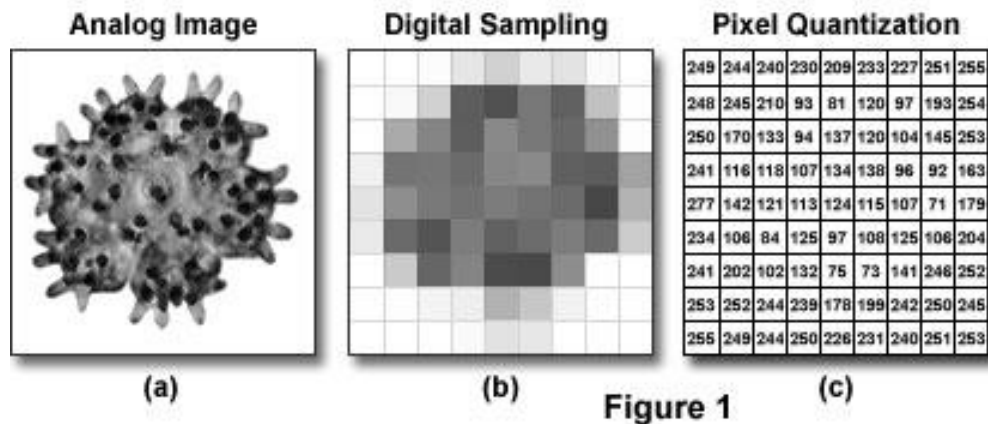


Figure 1

Spatial Resolution Effect on Pixelation in Digital Images

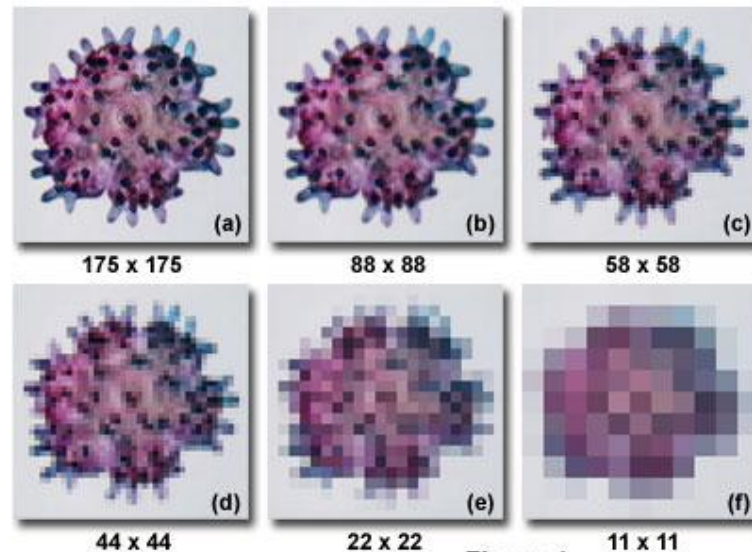





Figure 4

Alta risoluzione == Più informazione

Problema: come salviamo numeri?

Non possono essere astratti, ma **devono essere salvati fisicamente.**

-  una carica elettrica (nelle memorie/SSD)
-  una polarità magnetica (negli hard disk)
-  una riflessione di luce (nei CD e DVD)

Tuttavia, possiamo avere solo due stati: **1** e **0**

Esempio: carica elettrica o assenza di carica elettrica

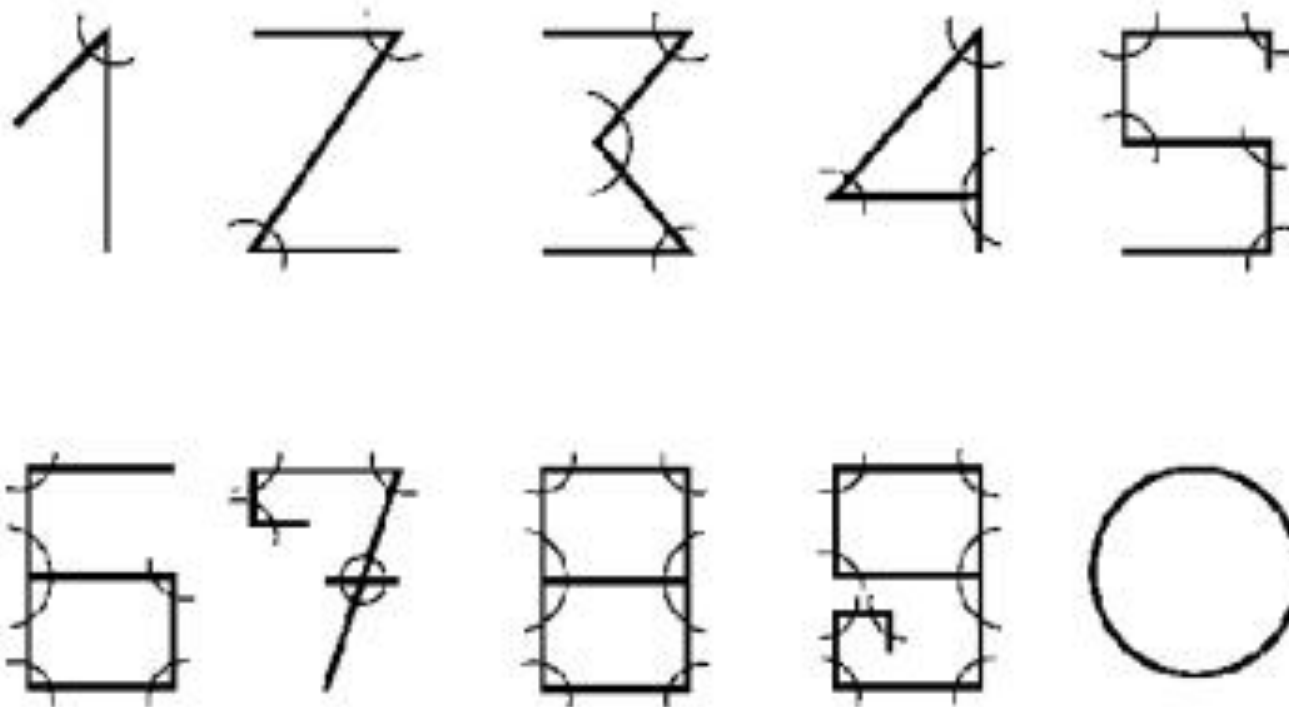
Il problema dei due simboli: 1 e 0

Dobbiamo rappresentare i numeri con solo due simboli: **Sistema di codifica Binaria** o **Sistema binario**

Perché si chiama sistema binario?

Perché si chiama decimale il sistema decimale?

Simboli numerici moderni

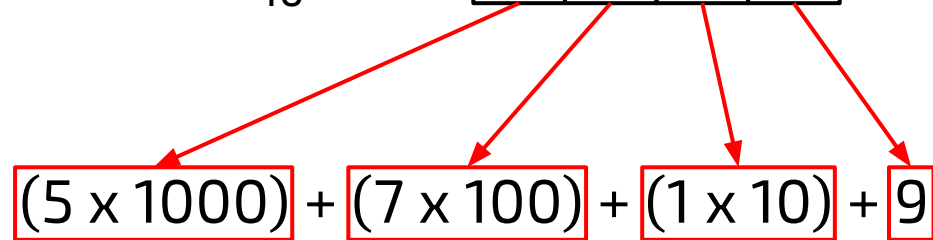


Sistema Decimale

$$5719_{10} = \begin{array}{|c|c|c|c|} \hline 5 & 7 & 1 & 9 \\ \hline \end{array}$$

Sistema Decimale

$$5719_{10} = \begin{array}{cccc} 3 & 2 & 1 & 0 \\ \boxed{5} & \boxed{7} & \boxed{1} & \boxed{9} \end{array}$$



$$\boxed{(5 \times 1000)} + \boxed{(7 \times 100)} + \boxed{(1 \times 10)} + \boxed{9}$$

Sistema Decimale

$$5719_{10} = \begin{array}{c} 3 \quad 2 \quad 1 \quad 0 \\ \boxed{5} \boxed{7} \boxed{1} \boxed{9} \end{array}$$

$(5 \times 10^3) + (7 \times 10^2) + (1 \times 10^1) + (9 \times 10^0)$

Sistema Binario

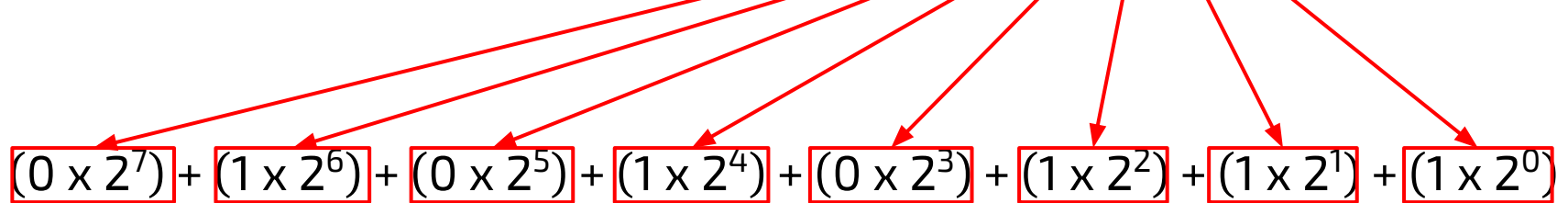
$$01010111_2 =$$

7	6	5	4	3	2	1	0
0	1	0	1	0	1	1	1

Sistema Binario

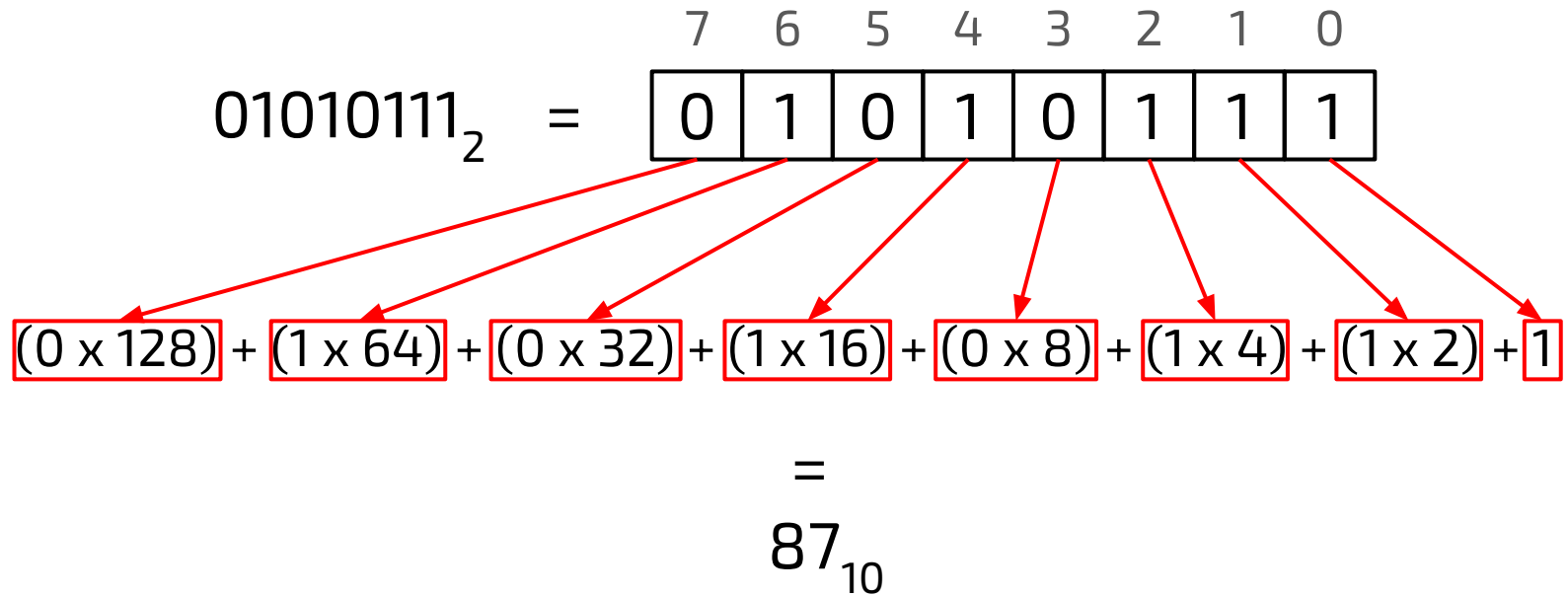
$$01010111_2 =$$

7	6	5	4	3	2	1	0
0	1	0	1	0	1	1	1



$(0 \times 2^7) + (1 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$

Sistema Binario



Da binario a decimale (intuitivo)

Vogliamo convertire il numero 61_{10} in binario


$$61_{10} = \begin{array}{cccccccc} 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \end{array}$$

Qual è la potenza di 2 più vicina a 61 che sia anche minore o uguale a 61?

Da binario a decimale (intuitivo)

Vogliamo convertire il numero 61_{10} in binario

$$61_{10} = \begin{array}{ccccccc} 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ \hline 0 & 0 & 1 & ? & ? & ? & ? & ? \end{array}$$


 $32 = 2^5$

Questa cifra ha un contributo di 32. Restano da identificare le cifre che hanno un contributo di $61 - 32 = 29$

Da binario a decimale (intuitivo)

Vogliamo convertire il numero 61_{10} in binario


$$61_{10} = \begin{array}{cccccccc} & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ \hline & 0 & 0 & 1 & ? & ? & ? & ? & ? \end{array}$$

Di nuovo: qual è la potenza di 2 più vicina a 29 che sia anche minore o uguale a 29?

Da binario a decimale (intuitivo)

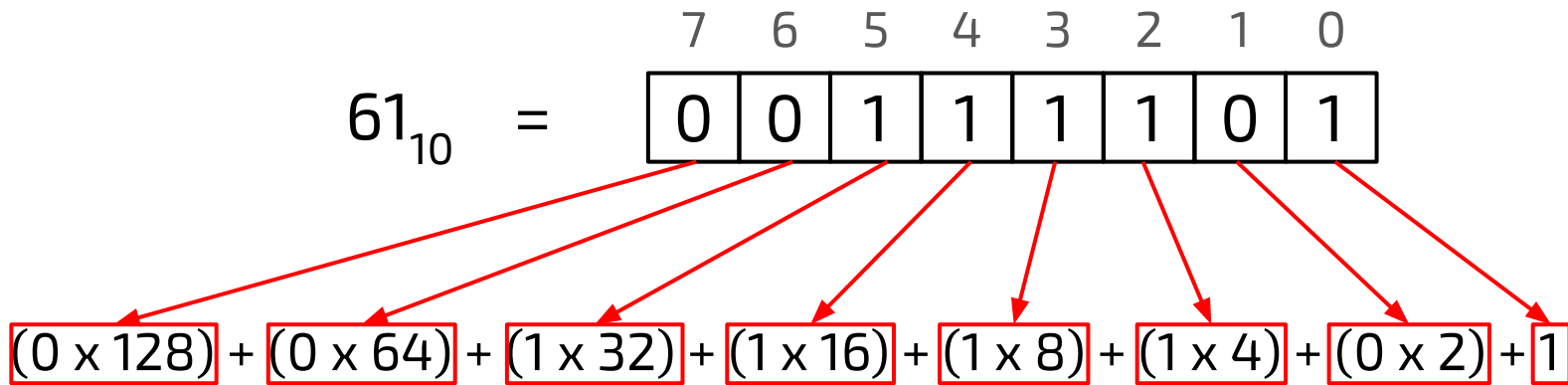
Vogliamo convertire il numero 61_{10} in binario

$$61_{10} = \begin{array}{cccccccc} 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & ? & ? & ? & ? \end{array}$$

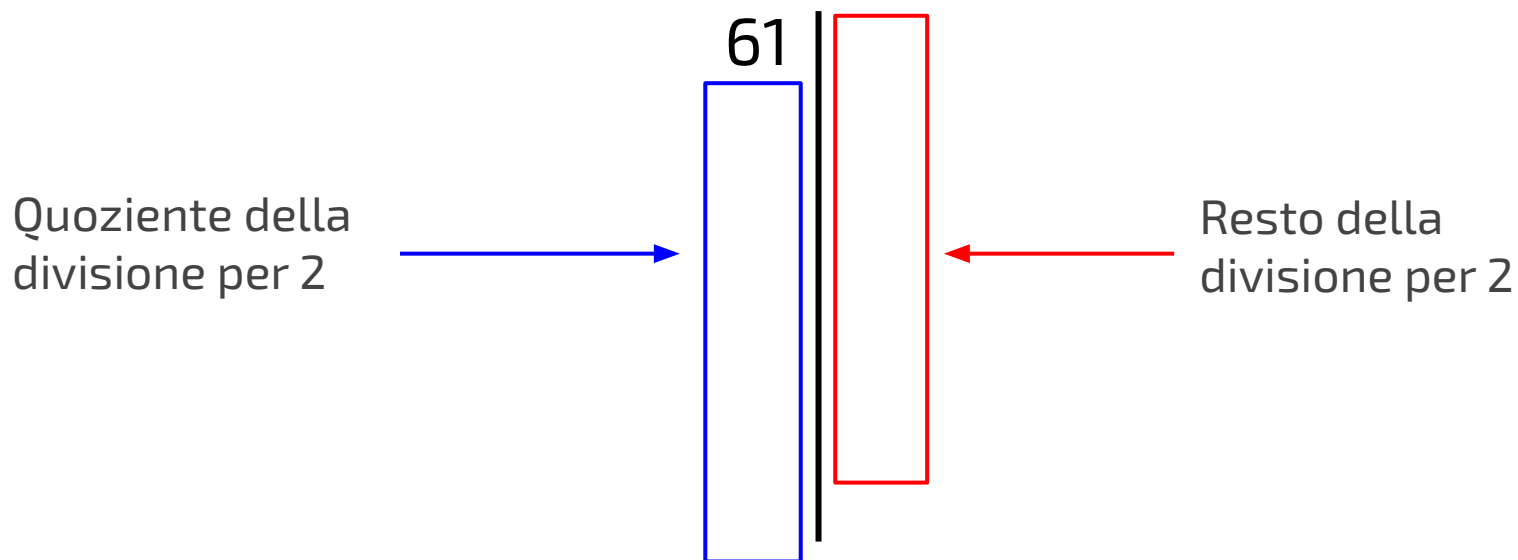

 $16 = 2^4$

Questa cifra ha un contributo di 16. Restano da identificare le cifre che hanno un contributo di $61 - 32 - 16 = 13$

Da binario a decimale (intuitivo)




Da binario a decimale (formale)




Da binario a decimale (formale)

61		1
30		0
15		1
7		1
3		1
1		1
0		

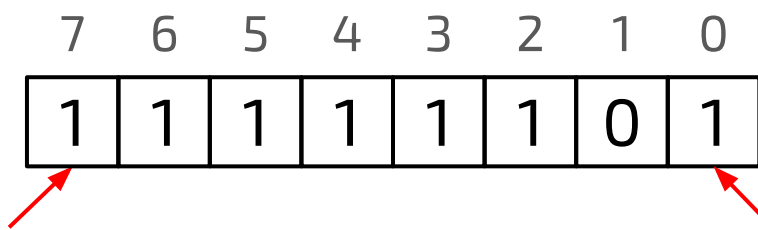


5	4	3	2	1	0
1	1	1	1	0	1



Sistema di numerazione binario nell'informatica

In informatica, le cifre binarie sono chiamate **bit (Binary digITs)**



MSB (Most Significant Bit)

LSB (Least Significant Bit)

Un bit descrive l'**informazione minima** (vero/falso)

Un gruppo di 8 bit si chiama **byte**

Un valore di byte può variare da 0_{10} (00000000_2) a 255_{10} (11111111_2).

Per esprimere numeri più grandi consideriamo sequenze di bit o byte più grandi

Sistema di numerazione binario nell'informatica

Leggere sequenze di bit non è banale. Se proprio dobbiamo leggere sequenze di bit andrebbero raggruppate. 10 non è un esponente di 2, quindi il sistema decimale risulterebbe scomodo.

Per questo motivo usiamo il **sistema esadecimale**.

Sistema Esadecimale

				3	2	1	0
0_{16}	=	0_{10}	=	0	0	0	0
1_{16}	=	1_{10}	=	0	0	0	1
2_{16}	=	2_{10}	=	0	0	1	0
3_{16}	=	3_{10}	=	0	0	1	1
4_{16}	=	4_{10}	=	0	1	0	0
5_{16}	=	5_{10}	=	0	1	0	1
6_{16}	=	6_{10}	=	0	1	1	0
7_{16}	=	7_{10}	=	0	1	1	1

				3	2	1	0
8 ₁₆	=	8 ₁₀	=	1	0	0	0
9 ₁₆	=	9 ₁₀	=	1	0	0	1
A ₁₆	=	10 ₁₀	=	1	0	1	0
B ₁₆	=	11 ₁₀	=	1	0	1	1
C ₁₆	=	12 ₁₀	=	1	1	0	0
D ₁₆	=	13 ₁₀	=	1	1	0	1
E ₁₆	=	14 ₁₀	=	1	1	1	0
F ₁₆	=	15 ₁₀	=	1	1	1	1



Da Esadecimale a Decimale

$$\text{EF25}_{16} = 0\text{xEF25} = \begin{array}{c} \begin{array}{cccc} 3 & 2 & 1 & 0 \\ \boxed{\text{E}} & \boxed{\text{F}} & \boxed{2} & \boxed{5} \end{array} \\ \begin{array}{c} \swarrow \quad \searrow \quad \swarrow \quad \searrow \\ \boxed{(14 \times 16^3)} + \boxed{(15 \times 16^2)} + \boxed{(2 \times 16^1)} + \boxed{(5 \times 16^0)} \end{array} \end{array}$$

Da Esadecimale a Decimale

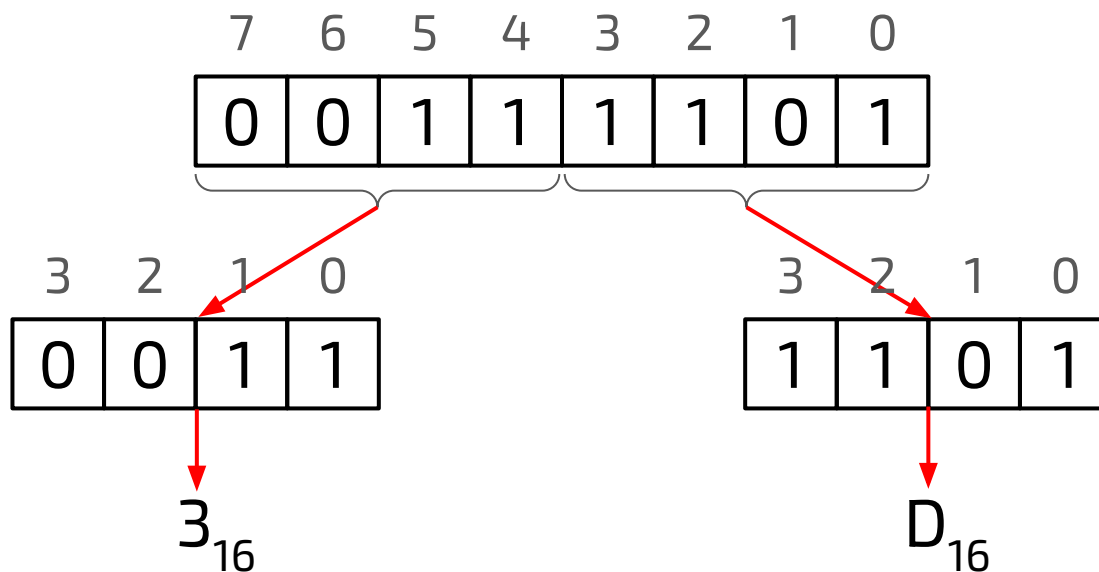
$$\begin{array}{ccccccc} & & & 3 & 2 & 1 & 0 \\ \text{EF25}_{16} & = & 0\text{xEF25} & = & \boxed{\text{E}} & \boxed{\text{F}} & \boxed{2} & \boxed{5} \\ & & & \swarrow & \downarrow & \searrow & \swarrow & \searrow \\ & & & \boxed{(14 \times 4096)} & + & \boxed{(15 \times 256)} & + & \boxed{(2 \times 16)} & + & \boxed{5} \\ & & & & & & & & & = \\ & & & & & & & & & 61221_{10} \end{array}$$

Da Decimale a Esadecimale

48879	F		
3054	E		
190	E		
11	B		
0			

0xBEEF

Da Binario a Esadecimale



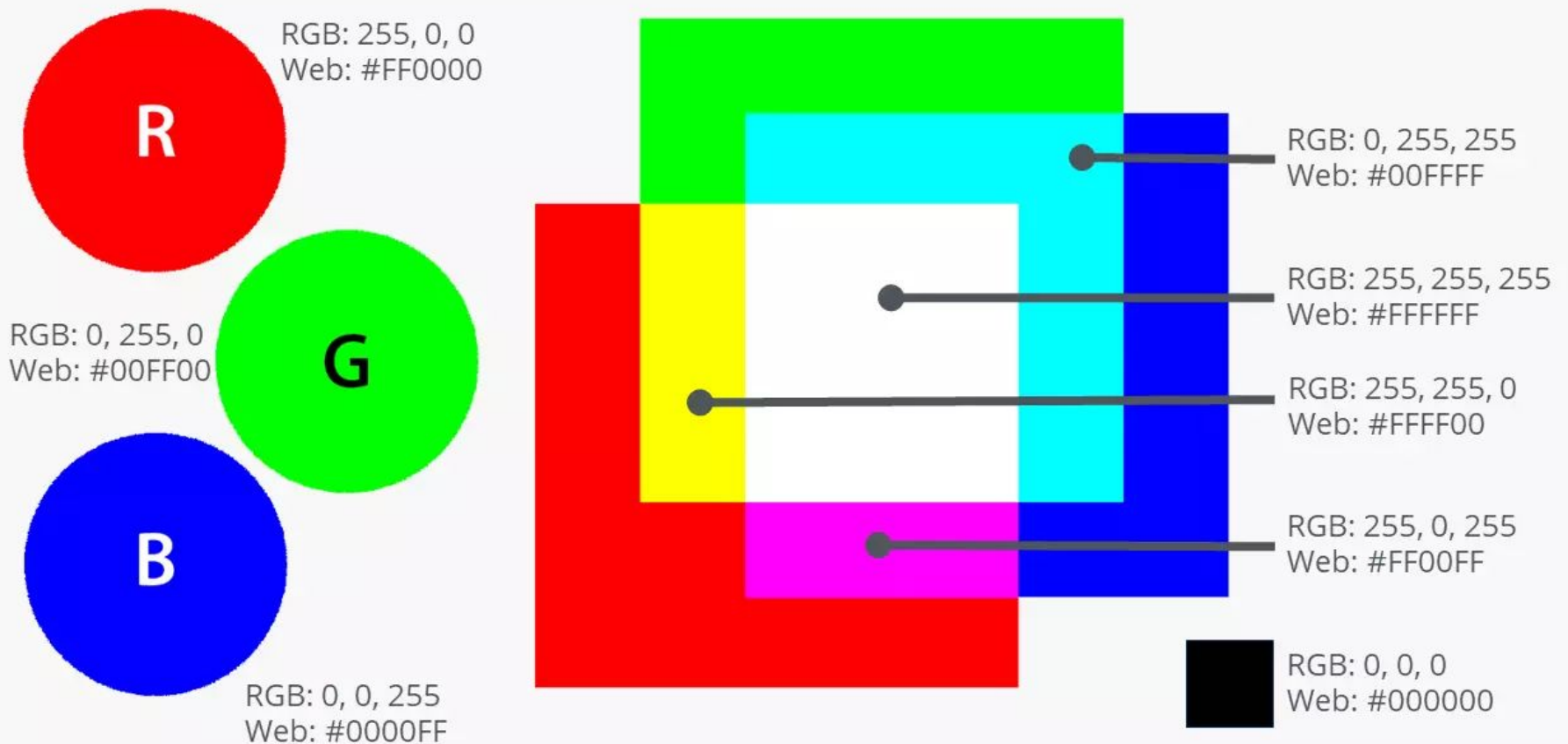
Gruppi di 2 cifre esadecimali compongono un byte

E i caratteri?

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

E le immagini?



IONOS

RGB: Additive Color Mixing

Operatori fra bit (logici): AND \wedge

Comunemente chiamata **congiunzione**. Una congiunzione è vera, **se entrambe le premesse sono vere**. Se 1 rappresenta vero e 0 rappresenta falso.

Bit #1	Bit #2	Risultato
0	0	0
0	1	0
1	0	0
1	1	1

Operatori fra bit (logici): OR \vee

Comunemente chiamata **disgiunzione inclusiva**.
Una disgiunzione inclusiva è vera, **se almeno una premessa è vera**. Se 1 rappresenta vero e 0 rappresenta falso.

Bit #1	Bit #2	Risultato
0	0	0
0	1	1
1	0	1
1	1	1

Operatori fra bit (logici): XOR \oplus

Comunemente chiamata **disgiunzione esclusiva**.

Una disgiunzione esclusiva è vera, **se esattamente una delle due premesse è vera**. Se 1 rappresenta vero e 0 rappresenta falso.

Bit #1	Bit #2	Risultato
0	0	0
0	1	1
1	0	1
1	1	0

L'operatore XOR è invertibile

AND:

$$? \wedge 0 = 0$$

OR:

$$? \vee 1 = 1$$

XOR:

$$? \oplus 0 = 0$$

$$? \oplus 0 = 1$$

$$? \oplus 1 = 0$$

$$? \oplus 1 = 1$$