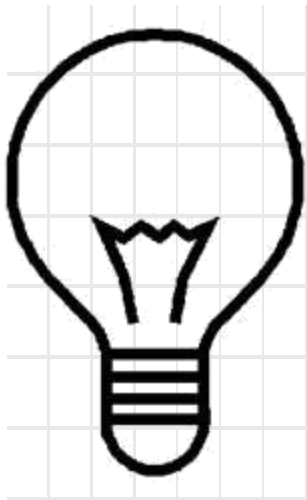


Per rendere una informazione comprensibile ad un computer è necessario esprimerla sotto forma di una sequenza di bit ovvero sotto forma di 0 e 1. Consideriamo il semplice caso di un disegno in bianco e nero come quello che vediamo nella figura che segue:



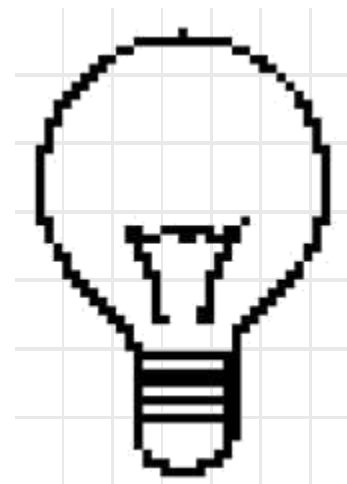
Per **estrarre** la sequenza di bit che rappresenti il suddetto disegno possiamo procedere in questo modo:

- per prima cosa dividiamo il disegno in quadratini molto piccoli, chiamati **pixel**, sovrappoendogli una griglia



- ad ogni quadratino della griglia diamo il colore **nero** se il contenuto di nero al suo interno supera quello del bianco, e viceversa diamo il colore **bianco** se il contenuto del bianco al suo interno supera quello del nero.

La successiva figura rappresenta proprio la **nuova immagine** che otterremo dopo aver portato a termine la suddetta operazione



A questo punto, ad ogni quadratino o pixel della griglia associamo uno **0** se il suo contenuto è il bianco e **1** in caso contrario.

In tal modo otterremo una **lunga sequenza di 0 e 1** che codifica l'immagine della lampadina in **formato digitale**.

La nostra immagine di partenza, dopo questa operazione si è dunque trasformata in una sorta di **mappa composta di 0 e 1**, ed infatti il nome tecnico che si usa per descrivere questa sequenza di bit è proprio **bitmap**.

Ovviamente, **più fitta è la griglia** che sovrapponiamo all'immagine e più la nostra rappresentazione digitale risulterà **fedele all'originale**.

D'altra parte, **una griglia più fitta** significa **un maggior numero di pixel** e quindi **un maggiore numero di bit** che dovranno essere utilizzati per descrivere digitalmente l'immagine.

#000000		#330000		#660000	
#000033		#333300		#660033	
#000066		#336600		#660066	
#000099		#339900		#660099	
#0000CC		#33CC00		#6600CC	
#0000FF		#33FF00		#6600FF	
#003300		#333330		#663300	
#003333		#333333		#663333	
#003366		#333366		#663366	
#003399		#333399		#663399	
#0033CC		#3333CC		#6633CC	
#0033FF		#3333FF		#6633FF	
#006600		#336600		#666600	
#006633		#336633		#666633	
#006666		#336666		#666666	
#006699		#336699		#666699	
#0066CC		#3366CC		#6666CC	
#0066FF		#3366FF		#6666FF	
#009900		#339900		#669900	
#009933		#339933		#669933	
#009966		#339966		#669966	
#009999		#339999		#669999	
#0099CC		#3399CC		#6699CC	
#0099FF		#3399FF		#6699FF	
#00CC00		#33CC00		#66CC00	
#00CC33		#33CC33		#66CC33	
#00CC66		#33CC66		#66CC66	
#00CC99		#33CC99		#66CC99	
#00CCCC		#33CCCC		#66CCCC	
#00CCFF		#33CCFF		#66CCFF	
#00FF00		#33FF00		#66FF00	
#00FF33		#33FF33		#66FF33	
#00FF66		#33FF66		#66FF66	
#00FF99		#33FF99		#66FF99	
#00FFCC		#33FFCC		#66FFCC	
#00FFFF		#33FFFF		#66FFFF	

Per un'immagine di partenza viceversa a colori, per ogni quadretto, o meglio per ogni pixel, anziché utilizzare solo uno 0 ed un 1, utilizzeremo una **combinazione di 0 e 1** in base a una **tabella di codifica dei colori**.

La successiva tabella di codifica dei colori è ad esempio composta di **64 colori**, e quindi ad ogni pixel potremo assegnare un numero **da 0 a 63 in binario** che individuerà uno solo dei colori della tabella stessa.

#990000		#CC0000		#FF0000	
#990033		#CC0033		#FF0033	
#990066		#CC0066		#FF0066	
#990099		#CC0099		#FF0099	
#9900CC		#CC00CC		#FF00CC	
#9900FF		#CC00FF		#FF00FF	
#993300		#CC3300		#FF3300	
#993333		#CC3333		#FF3333	
#993366		#CC3366		#FF3366	
#993399		#CC3399		#FF3399	
#9933CC		#CC33CC		#FF33CC	
#9933FF		#CC33FF		#FF33FF	
#996600		#CC6600		#FF6600	
#996633		#CC6633		#FF6633	
#996666		#CC6666		#FF6666	
#996699		#CC6699		#FF6699	
#9966CC		#CC66CC		#FF66CC	
#9966FF		#CC66FF		#FF66FF	
#999900		#CC9900		#FF9900	
#999933		#CC9933		#FF9933	
#999966		#CC9966		#FF9966	
#999999		#CC9999		#FF9999	
#9999CC		#CC99CC		#FF99CC	
#9999FF		#CC99FF		#FF99FF	
#99CC00		#CCCC00		#FFCC00	
#99CC33		#CCCC33		#FFCC33	
#99CC66		#CCCC66		#FFCC66	
#99CC99		#CCCC99		#FFCC99	
#99CCCC		#CCCCCC		#FFCCCC	
#99CCFF		#CCCCFF		#FFCCFF	
#99FF00		#CCFF00		#FFFF00	
#99FF33		#CCFF33		#FFFF33	
#99FF66		#CCFF66		#FFFF66	
#99FF99		#CCFF99		#FFFF99	
#99FFCC		#CCFFCC		#FFFFCC	
#99FFFF		#CCFFFF		#FFFFFF	

:: TAVOLOZZA UNIVERSALE PER I BROWSER :: www.gdesign.it

Dato che per esprimere in binario numeri fino a 63 sono necessari **sei bit**, vorrà allora dire che **ad ogni pixel dell'immagine corrisponderà una combinazione di sei bit.**

Ovviamente serviranno tanti più bit quanti più colori utilizziamo.

Se infatti usiamo **8 bit per ogni pixel**, i colori che possiamo utilizzare nella nostra immagine saranno **256**.

Se invece usiamo **16 bit per ogni pixel**, i colori che possiamo utilizzare nella nostra immagine saranno **65536**, e così via.

La maggior parte delle schede grafiche utilizza ormai **24 bit per ogni pixel** o addirittura **32 bit per ogni pixel**.

Le immagini codificate pixel per pixel con il procedimento appena visto, danno corpo, come si è già accennato, ai cosiddetti **file bitmap**, ed appunto caratterizzati dall'estensione **.BMP**

I dispositivi fisici per digitalizzare le immagini sono: lo **scanner** e la **fotocamera digitale**.