

FONDAMENTI DELL'INFORMATICA

APPUNTI DI CORSO 23/24

C

## F.D.I • 2.2.

**Definizione** >  $\forall a, b \in \mathbb{Z} \ (b \neq 0 \rightarrow \exists q \in \mathbb{Z} (a = q \cdot b))$

Dati 2 numeri  $a, b$ , si dice che  $b$  divide  $a$  "b divide  $a$ " per indicare che  $b$  è divisore di  $a$ , ovvero se si facesse  $a/b$  non ci sarebbe resto.

Infatti, facendo la divisione euclidea tra  $a$  e  $b$  si ha che  $a = q \cdot b + r$ , ma se  $b | a$  allora  $r=0$ .

### Esercizio: dimostrazione teoremi >

Dati  $m, n, d \in \mathbb{Z}$

- a. Se  $d|m \wedge d|n \Rightarrow d|(m+n)$
- b. se  $d|m \wedge d|n \Rightarrow d|(m-n)$
- c. se  $d|m \Rightarrow d|(m \cdot n)$

a) Ipotesi:  $d|m \wedge d|n$  Tesi:  $d|(m+n)$

Dim

$$m+n = q_1 \cdot d$$

$$m = q_1 \cdot d$$

$$n = q_2 \cdot d$$

$$m+n = q_1 \cdot d + q_2 \cdot d \stackrel{\in \mathbb{Z} \text{ siccome esiste}}{\Rightarrow} d(\underbrace{q_1+q_2}_{} \cdot d)$$

b) Ipotesi:  $d|m \wedge d|n$  Tesi:  $d|(m-n)$

Dim

$$m = q_1 \cdot d, d = q_2 \cdot d \leftarrow \text{per def. ipotesi} \quad d|(m-n) = \exists q_3 \text{ t.c.}$$

$$\begin{aligned} m-n &= \overbrace{q_1 \cdot d - q_2 \cdot d}^{\text{Per ipotesi}} \Rightarrow m-n = d(\underbrace{q_1-q_2}_{} \cdot d) \quad \text{e } q_1-q_2 \in \mathbb{Z} \text{ allora } m-n = q_3 \cdot d \end{aligned}$$

c) Ipotesi:  $d|m$  Tesi:  $d|(m \cdot n)$

Dim  $m = q_1 \cdot d$

$$m \cdot n = q_1 \cdot d \cdot n$$

$$\begin{aligned} m \cdot n &= q_1 \cdot d \cdot n \stackrel{d \in \mathbb{Z}}{=} q_2 \\ m \cdot n &= \overbrace{(q_1 \cdot n)}^{(q_1 \cdot n) \in \mathbb{Z}} \cdot d \end{aligned}$$

NUMERI PRIMI E COMPOSTI  $\Rightarrow \forall n \in \mathbb{Z} (\ n \geq 2 \wedge \exists n \wedge \nexists_{k \in \mathbb{Z}} (n \neq k \wedge k \mid n))$

Per numeri primi si intendono tutti i numeri interi maggiori di 1 che sono divisibili per se stessi o per 1.

Tutti gli altri numeri ad esclusione di 1 sono detti composti.

DETERMINARE SE UN NUMERO È PRIMO  $\Rightarrow \forall n \in \mathbb{Z} (n \text{ composto} \Leftrightarrow \exists d \in \mathbb{Z} (2 \leq d \leq \sqrt{n}))$

Un primo modo per determinare se un numero è primo o meno può essere provare a dividere il numero per 2, 3, ..., n-1 e vedere se soddisfa le condizioni.

$$\lfloor n \rfloor = \text{Floor}(n)$$

In realtà, dato n, basta fermarsi a  $\lfloor \sqrt{n} \rfloor$ . In particolare, dato n esso è composito se e solo se esiste un divisore d tale che  $2 \leq d \leq \sqrt{n}$

Dimostrazione  $\Rightarrow$  Ipotesi: n è composto  $\Rightarrow \exists d \in \mathbb{Z} (2 \leq d \leq \sqrt{n})$

Se n è composto, allora  $n = k_0 \cdot k_1$  ovvero  $n = q_0 \cdot d$   
allora ci possono essere 2 casi:

-  $d \leq \sqrt{n}$ : caso risolto

-  $d > \sqrt{n}$ : In questo caso, si può semplificare l'equazione

$$n = q_0 \cdot d \Rightarrow \frac{n}{d} = q_0 \quad \text{ma siccome } d > \sqrt{n}, \text{ allora}$$

$$\frac{n}{d} < \frac{n}{\sqrt{n}} \quad \Leftrightarrow \frac{n}{\sqrt{n}} > \frac{n}{d} \quad \text{Dividendo per qualcosa}$$

$\frac{\sqrt{n}}{d} > \frac{n}{d} = q_0$

ovvero l'altro elemento sarà comunque minore di  $\sqrt{n}$

di più piccolo si ottiene qualcosa di più grande

Ipotesi:  $\exists d \in \mathbb{Z} (2 \leq d \leq \sqrt{n}) \wedge d \mid n$   $\Rightarrow$  tesi = n è composto

Se  $d \neq n \wedge d \mid n \Rightarrow n$  è composto verificata.

F.D.I • L2 • 2023-09-21

TEOREMA FONDAMENTALE DELL'ARITMETICA >  $\forall n \in \mathbb{Z}_{\geq 1} \quad (n = p_0 \cdot p_1 \cdot p_2 \cdots)$

Il Teorema fondamentale dell'aritmetica dice che ogni  $n \in \mathbb{Z} > 1$  può essere scritto come moltiplicazioni tra primi

TEOREMA: I numeri primi sono infiniti >  $\forall n \in \mathbb{Z} \quad (n \text{ è primo} \Leftrightarrow \exists m \in \mathbb{Z} \text{ minimo}$

Ipotesi: nessuno Tesi: +

Dato un numero primo  $p$ , è possibile definire una sequenza di primi  $\leq p$ :  $p_1, p_2, \dots, p_n$ . Si definisce un altro numero  $m$  dato dal prodotto degli  $m = p_1 \cdot p_2 \cdots p_n + 1$  elementi della sequenza +.  $m$  potrebbe non potrebbe essere primo.

Per il Teorema fondamentale dell'Arithmetica,  $m$  può essere definito come prodotto di primi.  $m = q_1 \cdot q_2 \cdot q_3 \cdots$

I primi che formano il prodotto, però, non essendo una sequenza, devono essere maggiori della sequenza che compone  $m$ .

$m \geq q_1 \geq p_j$  di conseguenza esistevano dei primi maggiori  
per ogni ogni ovvero sono infiniti.

Perciò essendo che

$$m = p_1 \cdot p_2 \cdot p_3 \cdots + 1$$

ogni di quelli deve dividere  $m$   
per cui i primi che possono dividere essi devono essere > in quanto non nella sequenza

FDI • L3

MASSIMO COMUNE DIVISORE >  $\forall m, n \in \mathbb{Z}_{\geq 1} \quad (\text{gcd} = \max\{k \in \mathbb{Z} \mid k \mid m \wedge k \mid n\})$

Dati  $m, n \in \mathbb{Z}_{\geq 1}$ , il massimo comune divisore o gcd (mcd) è il divisore più grande di  $m$  ed  $n$

DIMOSTRAZIONE > Ipotesi: nessuna Tesi:  $\text{gcd} = \max\{\text{div di } m, n\}$

$$m = q_0 \cdot q_1 \cdot q_2 \cdots$$

definendo un  $K = \min\{q_0, q_1, q_2, \dots\}$

non c'è resto

costituito solo

dai gg. esponenti

minori

$$m/K = q_0 \cdot q_1 \cdot q_2 \cdots$$

comunidivisore

semplificabile

per gg. gg. gg.

$$m/K = q_0 \cdot q_1 \cdot q_2 \cdots$$

esso potrà dividere sia  $m$  che  $n$  in quanto scritto nella forma  $n = p \cdot q$

se invece si aggiungono esponenti agli elementi del prodotto, allora l'operazione non è semplificabile per cui  $\text{lcm}(m, n) \mid K$  e conseguentemente  $K$  è il massimo divisore

MINIMO COMUNE MULTIPLO >  $\text{lcm}(m, n) = \frac{m \cdot n}{\text{gcd}}$

Dati  $m, n \in \mathbb{Z}_{\geq 1}$ , il minimo comune multiplo

è il minimo numero divisibile sia da  $m$  che da  $n$

DIMOSTRAZIONE > Ipotesi: nessuna Tesi:  $\text{lcm} = \min\{k \in \mathbb{Z} \mid m \mid k \wedge n \mid k\}$

Analogamente alla dim. del gcd:

$$m = q_0^{a_1} \cdot q_1^{a_2} \cdot q_2^{a_3} \cdots$$

$$\exists K = q_0^{\max(a_1, b_1)} \cdot q_1^{\max(a_2, b_2)} \cdots$$

$$K/m = \frac{q_1^{a_2} \cdot q_2^{a_3} \cdots}{q_0^{a_1} \cdot q_1^{a_2} \cdots} \checkmark \quad m \mid K$$

$$n = q_0^{b_1} \cdot q_1^{b_2} \cdot q_2^{b_3} \cdots$$

Prendendo il massimo si è certi che  $m \mid K \wedge n \mid K$   
il min in quanto minoranti esponenti  $m \mid K \wedge n \mid K$

PRODOTTO TRA MDC E LCM  $\Rightarrow \text{gcd}(m,n) \cdot \text{lcm}(m,n) = m \cdot n$

La proprietà dice che per ogni  $m, n \in \mathbb{Z}_{\geq 0}$ , il loro prodotto è equivalente al prodotto tra il gcd e lcm.

### DIMOSTRAZIONE >

① Teorema:  $\min(x,y) + \max(x,y) = x+y$

$$- x = y: \quad x+y = x+y \quad \text{OK}$$

$$- x > y: \quad y+x = x+y \quad \text{OK}$$

$$- y < x: \quad x+y = x+y \quad \text{OK}$$

NB: le logiche le formule inverse  
per trovare gcd e lcm

② Teorema:  $\text{gcd}(m,n) \cdot \text{lcm}(m,n) = m \cdot n$

$$\frac{\min(a,b)}{q_1} \cdots \frac{\min(a,b)}{q_k} \cdot \frac{\max(a,b)}{q_1} \cdots \frac{\max(a,b)}{q_k} = \frac{\min(a,b) \cdot \max(a,b)}{\text{gcd}(a,b)} \cdot \frac{\max(a,b) \cdot \min(a,b)}{\text{lcm}(a,b)}$$

li moltiplico tra di loro:

$$\frac{\min(a,b) + \max(a,b)}{q_1} \cdots \frac{\min(a,b) + \max(a,b)}{q_k} = \frac{\min(a,b) + \max(a,b)}{q_1} \cdots \frac{\min(a,b) + \max(a,b)}{q_k}$$

per il Teorema 1:

$$\frac{a+b}{q_1} \cdots \frac{a+b}{q_k} = \frac{a}{q_1} \cdots \frac{a}{q_k} + \frac{b}{q_1} \cdots \frac{b}{q_k} \quad \text{che si possono rappresentare come} \\ m \qquad \qquad \qquad n = m \cdot n \quad \text{OK}$$

### PORTE LOGICHE >

Le principali porte logiche utilizzate sono:

		AND
A	B	AND
0	0	0
0	1	0
1	0	0
1	1	1

L'operazione di AND è una moltiplicazione binaria

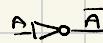


		OR
A	B	OR
0	0	0
0	1	1
1	0	1
1	1	1

L'operazione di OR è un'addizione binaria



		NOT
A		NOT
0		1
1		0



		XOR
A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

L'operazione di XOR è una OR esclusiva ovvero esclude la possibilità che  $A=1 \wedge B=1$



## F2P FCOP &gt;

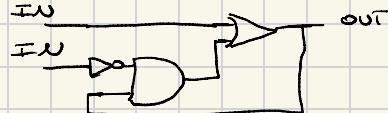
Il Flip Flop è un'unità fondamentale formata dai circuiti fondamentali in grado di memorizzare 1 bit.

FUNZIONAMENTO > Il Flip Flop ha 2 inputs

- CASO 0,0 - SIGNLE

In caso si passi 0,0 nei 2 ingressi

- Nel 2° ingresso, lo 0 sarà trasformato in 1 in modo che ritorni il valore proveniente dall'output con l'AND



- Nel 3° ingresso, lo 0 basterà all'OR per ritornare il valore proveniente dall'output

- CASO 1,0 - SET

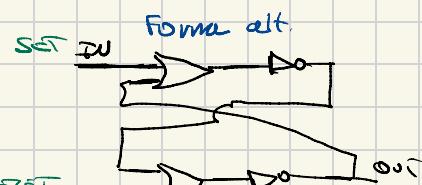
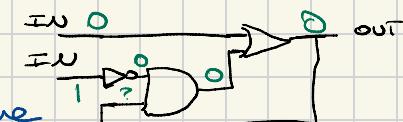
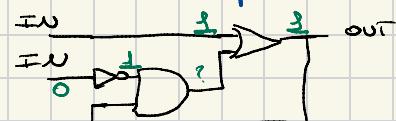
In caso si passi 1,0 nei 2 ingressi:

- Nel primo ingresso, il valore nell'OR sarà 1 siccome l'uscita finale è un OR, indipendentemente dal valore del secondo ingresso, l'output sarà 1

- CASO 0,1 - RESET / RESET

In caso si passi 0,1 nei 2 ingressi:

- Nel secondo ingresso, il valore prima dell'AND sarà 0 di conseguenza il risultato sarà 0
- Nel primo ingresso, venire come "parametri" 0 e 0 (ris. dell'AND) dà in uscita come risultato 0



### NOTAZIONE ESADECIMALE >

È possibile utilizzare la notazione esadecimale 0x... per poter rappresentare 4 o più bit. In particolare 1 cifra HEX è in grado di rappresentare 4 bit

bit	HEX
0000	0x0
0001	0x1
0010	0x2
0011	0x3
0100	0x4
0101	0x5
0110	0x6
0111	0x7
1000	0x8
1001	0x9
1010	0xA
1011	0xB
1100	0xC
1101	0xD
1111	0xE

## UNITÀ DI MISURA &gt;

Essendo il bit, l'unità più piccola di informazione in informatica, esso può essere raggruppato in Bytes, ovvero 8 bit.

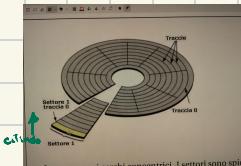
Sono anche in uso i seguenti prefissi per multipli binari:

Nome	Fattore	Equivalenti	F. Equivalenti
(Ki)	Kibi	$2^{10}$	Kilo
(Mi)	Mebi	$2^{20}$	Mega
(Gi)	Gibi	$2^{30}$	Giga
(Ti)	Tebi	$2^{40}$	Tera
(Pi)	Petabi	$2^{50}$	Peta
(Ei)	Exabi	$2^{60}$	Exa
(Zi)	Zettabi	$2^{70}$	Zetta
(Yi)	Yottabi	$2^{80}$	Yotta

## DISCHI MAGNETICI &gt;

I dischi magnetici sono una tipologia di memoria lette attraverso delle testine che passano nelle concavità per leggerne i dati. È formato da:

- Tracce: "cerchi" concentrici formati dalle concavità. La Traccia 0 è quella più esterna
- Settore: "spicchi" creati da concavità verticali che dividono il disco in cerchi. Dalle varie pile di dischi, sono le tracce alle stesse distanze dal centro
- Cilindri: Sono come i cerchi concentrici. I settori sono sposti



## METRICHE DEI DISCHI MAGNETICI >

Dai dischi magnetici è possibile leggere le seguenti metriche:

- Seek Time: Tempo impiegato per spostare le testine di lettura / scrittura da una traccia all'altra.

- Rotational Delay: Tempo impiegato per fare metà di una rotazione completa. Solitamente usato per ruotare delle testine di lettura e scrittura una volta impostata la traccia.

- Access Time: Rotational Delay + Seek Time

- Transfer Rate: Tempo complessivo per trasferire in/out i dati del disco



## COMPACT DISKS >

I cd, a differenza dei dischi magnetici, sono composti da una spirale di parti, leggibili da un laser.

In particolare, viene sfruttata la riflettività degli elementi del disco per determinare se è 1 o 0.

In fase di scrittura, invece, si utilizza un laser più potente per poter trasformare le parti del CD in assorbenti o riflettenti.

## MEMORIE FLASH >

Per memoria flash si intende la famiglia di storage fornita da USB, micro SD, SD, SSD, <sup>TF</sup>TF<sub>TF</sub>

A differenza della RAM, esse conservano l'informazione degli elettroni, riuscendo a conservarla per qualche anno, senza alimentazione all'interno di celle di ossido di silicio.

"ASCII"

ASCII > 01001000 = "H"

Usato per la codifica dell'alfabeto. Ogni simbolo è rappresentato da 1 Byte, dei quali 7 sono solitamente usati.

1.043

## EVOLUZIONE IN ISO → UNICODE (OTP) >

Per accomodare esigenze linguistiche si è poi deciso di sfruttare l'ultimo bit con lo standard ISO. Più avanti è nato il progetto Unicode, formato da 21 bit, che per semplicità sono formattate negli 8 bit dei quali il 1°

per chiedere altri bit

BINARY NOTATION  $> 0000000 \rightarrow 0, 0000001 \rightarrow 1, 0000010 \rightarrow 2 \dots$

Siccome è inefficiente rappresentare i numeri usando ASCII, si rappresentano usando la notazione binaria degli stessi.

Con 1 singolo bit si possono rappresentare 256 numeri differenti.

OPERAZIONI BINARIE: somma >

La somma di 2 numeri in binario viene fatta in maniera simile a:

$$- 1+0=1 \quad 0+1=1$$

$$- 0+0=0$$

$$- 1+1=0 \quad \text{con riporto di } 1 \quad \text{come se si facesse } \begin{array}{r} 1 \\ 0 \\ + \\ 0 \\ \hline 1 \end{array}$$

$$\begin{array}{r} 1010 \\ 0011 \\ \hline 1101 \end{array}$$

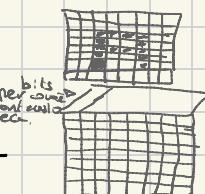
## RAPPRESENTAZIONE DELLE IMMAGINI >

Siccome ogni informazione rappresentata al computer è formata da 0/1 anche le immagini sono rappresentabili in diversi modi.

### - Tramite Bitmap / Raster

Tradotto in "Mappe di bit", prevede di rappresentare l'immagine in bianco e nero <sup>in una matrice</sup> usando i bit che coloreranno le celle dello stesso (1), nero(0).

In caso si volesse rappresentare anche i colori, la maniera più semplice sarebbe rendere la matrice dei bit in bi-dimensionale.



### - Tramite rappresentazione vettoriale

La rappresentazione vettoriale o suo sfrutta le equazioni per poter rappresentare le immagini.

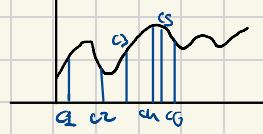
Vengono usate principalmente per loghi, font, CAD e hanno le caratteristiche di non perdere qualità quando si scalano in quanto i dettagli sono presenti nella formula.

## RAPPRESENTAZIONE DEL SUONO ➤

I suoni può essere codificato con le seguenti tecniche

- Tramite "campionamento"

Dato "l'onda musicale", si misura <sup>con</sup> ~~con~~ <sup>44.000</sup> suona metrica (es: altezza) nel tempo <sup>n</sup> volte in 16 bit o 32 per lo stereo.



BB: I dati tra un campionamento e l'altro sono pari.

- Tramite Midi

Detto Musical Instrument Digital Interface, effettua l'encoding delle note suonate del singolo strumento per n tempo.

## CONVERSIONI: BIN ↔ DEC ➤

Le conversioni da Binario a Decimale e vice versa si effettuano con le seguenti tecniche:

## BIN → DEC ➤

- Per ogni posizione

- Moltiplicare il numero per  $2^x$  dove  $x$  è la posizione.

$$\begin{array}{r} 10100 \\ \times 2^3 2^2 2^1 2^0 \\ \hline 10100 \end{array}$$

## DEC → BIN ➤

- Dato il numero, dividerlo per 2

- Se non ha resto = 0
- Se ha resto = 1

- Il risultato va letto al contrario.

$$\begin{array}{r} 20 \\ \hline 2 | 0 \\ 2 | 0 \\ 2 | 0 \\ 2 | 0 \\ 0 | 1 \end{array} \quad = 10100$$