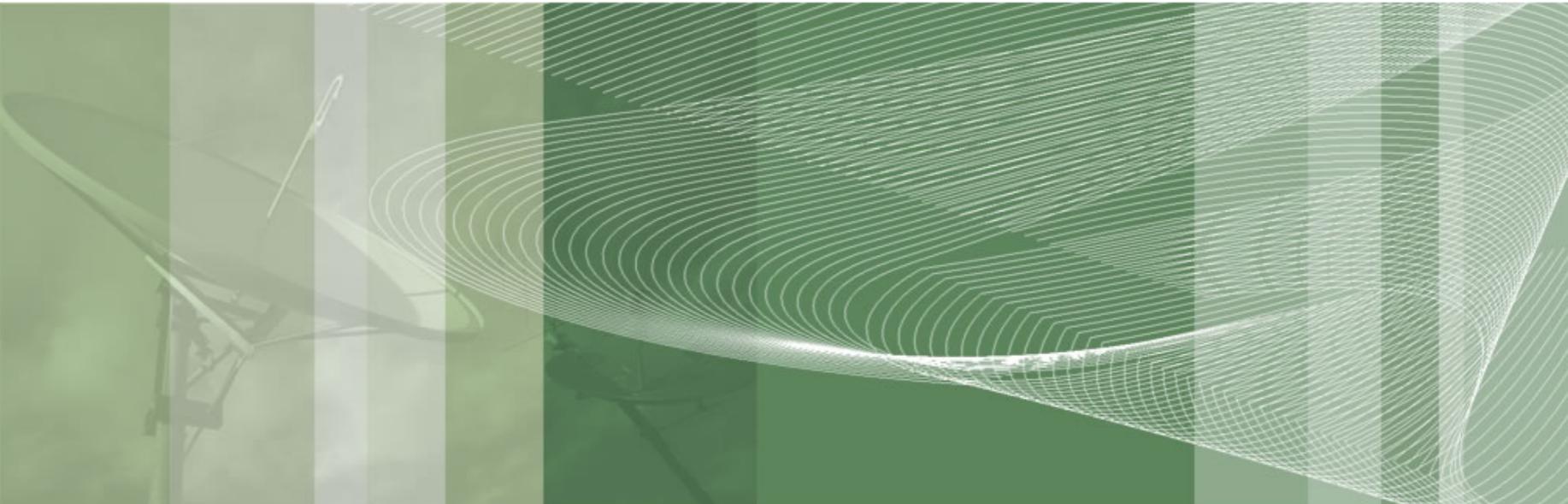


 POLITECNICO DI MILANO

Dipartimento di  
Elettronica e Informazione



# Prima esercitazione: algoritmi e tipi di dato

Matteo Ferroni

[matteo.ferroni@polimi.it](mailto:matteo.ferroni@polimi.it)

29/09/2017

# Chi sono

---



## Matteo Ferroni

- Ph.D. @ NECSTLab w/ Santa
- Ho lavorato su modelli di potenza per sistemi energy-aware
- Esercitatore di questo corso
- Web site: **matteo.ferroni.me**
- Email: **matteo.ferroni@polimi.it**

b o t  
t e g  
a 5 2 .it

# Agenda

---

- Un programma in C
- Algoritmi: esempio del DVD
- I tipi di dato
- Caratteri e codifica ASCII

# Un programma in C

inclusione librerie / per poter invocare funzioni utili (i/o, ...) /  
dichiarazione di **variabili globali** e **funzioni**

```
int main ( ) {
```

dichiarazione di **variabili locali**

```
    istruzione 1;      / tutti i tipi di operazioni, e cioè: /  
    istruzione 2;      / istr. di assegnamento /  
    istruzione 3;      / istr. di input / output /  
    istruzione 4;      / istr. di controllo (condizionali, cicli) /  
    ...
```

```
    istruzione N;
```

```
    return 0;  
}
```

parte dichiarativa **globale**

parte dichiarativa **locale**

parte **esecutiva**

# Un programma in C

---

Ma prima di pensare al codice...

**RAGIONATE  
SULL'ALGORITMO!**



Dove lo cerco?  
In quale *settore*?

Novita'

Azione  
...

Indicatori di settore



Trovato il settore, come  
trovo il mio DVD?

Ordine alfabetico



- 1. Acquisisci il nome del settore (s) del DVD (d)**
  - 2. Vai al settore (s)**
  - 3. Cerca (t)**
  - 4. Prendi il DVD (d)**
- 
- **Acquisisci e Cerca sono anche loro *algoritmi***



- Contesto
  - Sei nel settore (s) corretto
  - Nel settore sono presenti diversi DVD
    - Il settore puo' essere visto come un insieme di DVD:  $s=\{d_0, d_1, \dots, d_n\}$
- Problema
  - Devo trovare il DVD ( $d_t$ ) tra tutti i DVD presenti nel settore (s)
  - **Aiuto:** il settore e' ordinato
    - I DVD sono ordinati alfabeticamente



- Noti
  - $s=\{d_0, d_1, \dots, d_n\}$ , ord. alfabeticamente
  - DVD cercato =  $d_t$

## 1. Ci sono DVD?

- Allora
  1. Leggo il titolo del primo DVD in  $s$
  2. Se il titolo e' il titolo cercato
    - Allora concludo la ricerca con successo
    - Altrimenti passo al DVD successivo
- Altrimenti concludo la ricerca con esito negativo

- Cosa succede se il settore e' vuoto?
- Come *funziona* la mia ricerca?
  - Se in s vi sono 10000 DVD e io cerco Zorro?
  - Scenario ancora peggiore
    - Voglio cercare il numero del mio professore di IEIM (Santambrogio) nella rubrica telefonica di Milano..
- Esistono diversi modi per risolvere un problema



- Noti
  - $s=\{d_0, d_1, \dots, d_n\}$ , ord. alfabeticamente
  - DVD cercato =  $d_t$

## 1. Ci sono DVD?

- Allora
  1. Leggo il titolo del DVD ( $d_x$ ) nel *mezzo* di  $s$
  2. Se il titolo di  $d_x$  e' il titolo cercato
    - Allora concludo la ricerca con successo
    - Altrimenti se  $d_x < d_t$ 
      - » allora ricomincio da 1 considerando la metà superiore di  $s$
      - » Altrimenti ricomincio da 1 considerando la metà inferiore di  $s$
- Altrimenti concludo la ricerca con esito negativo



# Graficamente: Noti



POLITECNICO  
DI MILANO

DIPARTIMENTO DI ELETTRONICA E INFORMAZIONE



$s = \{d_0, d_1, \dots, d_{11}\}$

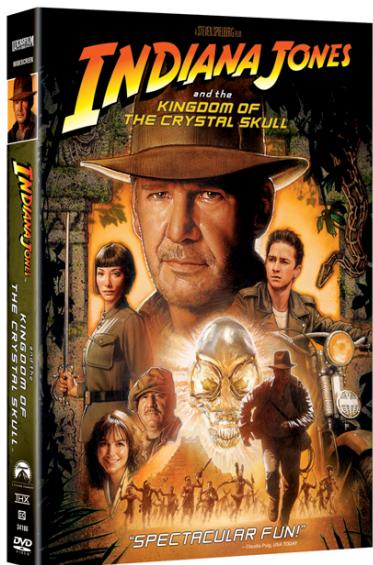




# Leggo $d_x$ e lo confronto con $d_t$

$$d_x = d_6$$

$$d_6 = d_t ?$$



NO

$$d_6 < d_t$$



# Ricomincio sulla meta' superiore



POLITECNICO  
DI MILANO

DIPARTIMENTO DI ELETTRONICA E INFORMAZIONE



d<sub>6</sub>

d<sub>7</sub>

d<sub>8</sub>

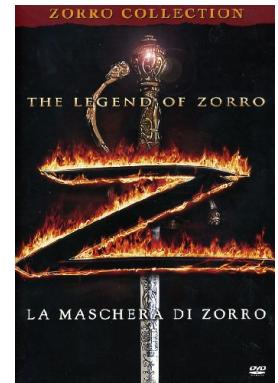
d<sub>9</sub>

d<sub>10</sub>

d<sub>11</sub>



$$d_x = d_9$$





# Leggo $d_x$ e lo confronto con $d_t$

$d_x = d_9$

$$d_9 = d_t ?$$



SI

$d_9$



Termino con successo la  
mia ricerca comprando  $d_9$



- Problema
  - Richiedi all'utente la sua altezza in centrimetri e mostrala a video in metri
- Pseudocodice
  - 1.Scrivi “quanto sei alto?”
  - 2.Leggi altezzacm
  - 3.Altezzam = alteccacm/100
  - 4.Scrivi “sei alto: altezzam”



- Pseudocodice

1. Scrivi “quanto sei alto?”
2. Leggi altezzacm
3. Altezzam = altezzacm/100
4. Scrivi “sei alto: altezzam”

```
#include <stdio.h>

int main () {
    int altezzacm;
    int altezzam;

    printf("Quanto sei alto? ");
    scanf("%d",&altezzacm);
    altezzam = altezzacm/100;
    printf("Sei alto %d m\n",altezzam);

    return 0;
}
```



```
#include <stdio.h>

int main () {
    int altezzacm;
    int altezzam;

    printf("Quanto sei alto? ");
    scanf("%d", &altezzacm);
    altezzam = altezzacm/100;
    printf("Sei alto %d m\n", altezzam);

    return 0;
}
```

Tra “” ci si aspetta una sequenza di caratteri

Provate ad inserire  
-1347

Si sono premuti 6 caratteri:

-, 1, 3, 4, 7, enter

Abbiamo quindi inserito una sequenza di caratteri!

%d dice al calcolatore di *interpretare* questa sequenza come un intero

- Pseudocodice

1. Scrivi “quanto sei alto?”
2. Leggi altezzacm
3. Altezzam = altezzacm/100
4. Scrivi “sei alto: altezzam”

```
#include <stdio.h>

int main () {
    int altezzacm;
    int altezzam;

    printf("Quanto sei alto? ");
    scanf("%d",&altezzacm);
    altezzam = altezzacm/100;
    printf("Sei alto %d m\n",altezzam);

    return 0;
}
```



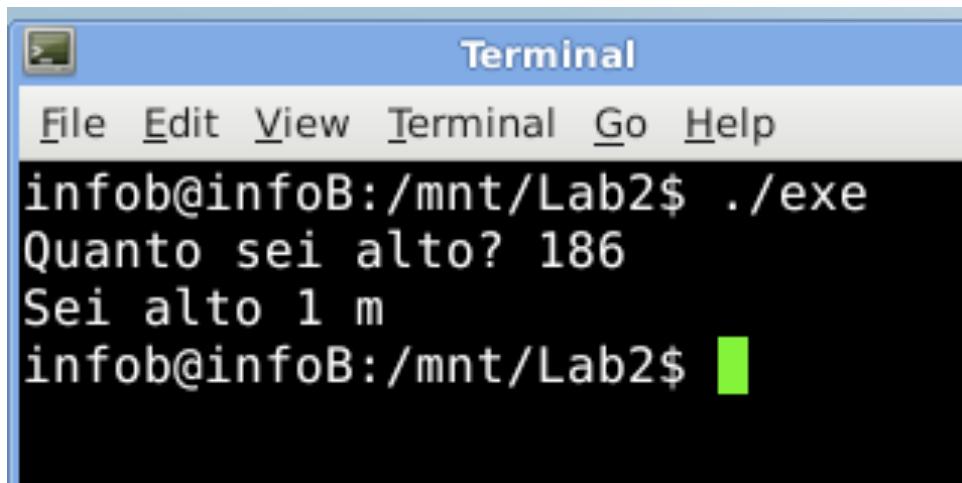
# Un primo errore

```
#include <stdio.h>

int main () {
    int altezzacm;
    int altezzam;

    printf("Quanto sei alto? ");
    scanf("%d",&altezzacm);
    altezzam = altezzacm/100;
    printf("Sei alto %d m\n",altezzam);

    return 0;
}
```



The screenshot shows a terminal window titled "Terminal". The menu bar includes "File", "Edit", "View", "Terminal", "Go", and "Help". The command "infob@infoB:/mnt/Lab2\$ ./exe" is entered, followed by the user's input "Quanto sei alto? 186". The program outputs "Sei alto 1 m". The terminal prompt "infob@infoB:/mnt/Lab2\$" is visible again at the end.



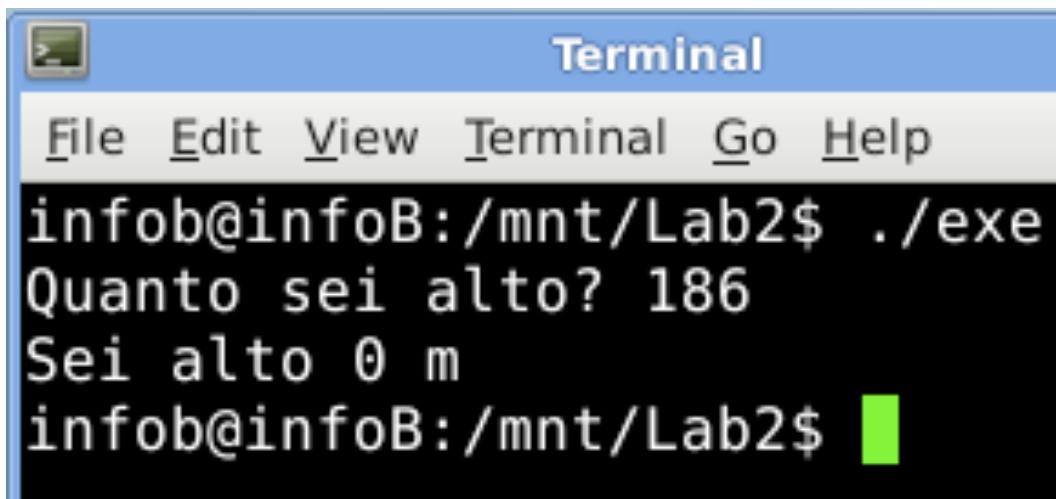
# Un secondo errore

```
#include <stdio.h>

int main () {
    int altezzacm;
    float altezzam;

    printf("Quanto sei alto? ");
    scanf("%d", &altezzacm);
    altezzam = altezzacm/100;
    printf("Sei alto %d m\n", altezzam);

    return 0;
}
```



Terminal

File Edit View Terminal Go Help

infob@infoB:/mnt/Lab2\$ ./exe

Quanto sei alto? 186

Sei alto 0 m

infob@infoB:/mnt/Lab2\$ █



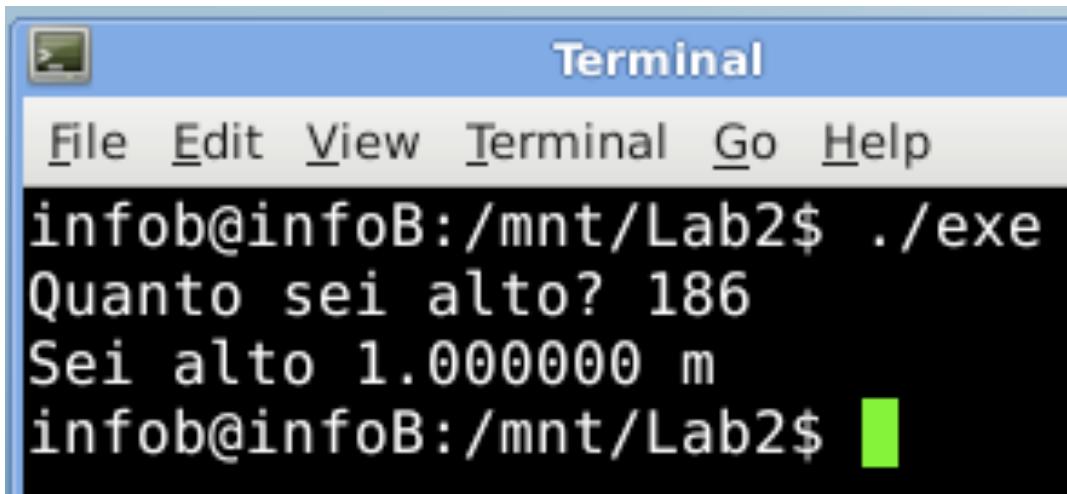
# Un terzo errore

```
#include <stdio.h>

int main () {
    int altezzacm;
    float altezzam;

    printf("Quanto sei alto? ");
    scanf("%d",&altezzacm);
    altezzam = altezzacm/100;
    printf("Sei alto %f m\n",altezzam);

    return 0;
}
```



```
Terminal
File Edit View Terminal Go Help
infob@infoB:/mnt/Lab2$ ./exe
Quanto sei alto? 186
Sei alto 1.000000 m
infob@infoB:/mnt/Lab2$ █
```



# Soluzione corretta



```
#include <stdio.h>

int main () {
    int altezzacm;
    float altezzam;
    float rapporto = 100;

    printf("Quanto sei alto? ");
    scanf("%d",&altezzacm);
    altezzam = altezzacm/rapporto;
    printf("Sei alto %f m\n",altezzam);

    return 0;
}
```

```
Terminal
File Edit View Terminal Go Help
infob@infoB:/mnt/Lab2$ ./exe
Quanto sei alto? 186
Sei alto 1.860000 m
infob@infoB:/mnt/Lab2$
```

L'importanza dei tipi di dato



# Tipi di dato “classici” in C

- In C esistono diversi tipi di dato built-in, tra cui
  - int: numeri interi
  - float: numeri con virgola (singola precisione)
  - double: numeri con virgola (doppia precisione)
  - char: caratteri (sono interi che possono variare tra 0-255)
- Inoltre il C fornisce anche la possibilità di definire dei nuovi tipi di dato



# Tipo carattere e codifica ASCII

```
#include <stdio.h>
```

```
int main () {
    char carattere;

    printf("Inserisci un carattere: ");
    scanf("%c",&carattere);
    printf("Il carattere inserito e': %c\n",carattere);
    printf "... ma anche: %d\n",carattere);
```

```
return 0;
}
```

```
Terminal
File Edit View Terminal Go Help
infob@infoB:/mnt/Lab2$ ./exe
Inserisci un carattere: d
Il carattere inserito e': d
... ma anche: 100
infob@infoB:/mnt/Lab2$ █
```

# **WAT.**

**TUTTO È *NUMERI***  
**IL CALCOLATORE RAGIONA**  
**SU QUESTI DATI**





# Tipo carattere e codifica ASCII



DIPARTIMENTO DI ELETTRONICA E INFORMAZIONE

000	NUL	033	!	066	B	099	c	132	ä	165	Ñ	198	â	231	þ
001	Start Of Header (SOH)	034	"	067	C	100	d	133	à	166	¤	199	Ã	232	þ
002	Start Of Text (STX)	035	#	068	D	101	e	134	å	167	°	200	Ł	233	Ú
003	End Of Text (ETX)	036	\$	069	E	102	f	135	ç	168	¿	201	ƒ	234	Ú
004	End Of Transmission (EOT)	037	%	070	F	103	g	136	ê	169	®	202	݂	235	܂
005	Enquiry	038	&	071	G	104	h	137	ë	170	߱	203	܂	236	܂
006	Acknowledge (ACK)	039		072	H	105	i	138	è	171	߲	204	܂	237	܂
007	Bell	040	(	073	I	106	j	139	ି	172	߳	205	=	238	-
008	Backspace (BS)	041	)	074	J	107	k	140	ି	173	ି	206	ଫ	239	-
009	Horizontal Tab	042	*	075	K	108	l	141	ି	174	«	207	ମ	240	-
010	Line Feed (LF)	043	+	076	L	109	m	142	ା	175	»	208	ସ	241	୧
011	Vertical Tab	044	,	077	M	110	n	143	ା	176	କ	209	ଡ	242	-
012	Form Feed (FF)	045	-	078	N	111	o	144	ଇ	177	କ	210	ୟ	243	୩
013	Carriage Return (CR)	046	.	079	O	112	p	145	ସେ	178	କ	211	ୟ	244	୮
014	Shift Out	047	/	080	P	113	q	146	ଏୟେ	179	ି	212	ୟ	245	୬
015	Shift In	048	0	081	Q	114	r	147	୦	180	ି	213	ି	246	ି
016	Dataline Escape (DLE)	049	1	082	R	115	s	148	୥	181	ା	214	ି	247	୯
017	DC 1 (XON)	050	2	083	S	116	t	149	୦	182	ା	215	ି	248	୯
018	DC 2	051	3	084	T	117	u	150	୻	183	ା	216	ି	249	-
019	DC 3 (XOFF)	052	4	085	U	118	v	151	୹	184	୧୦	217	ି	250	-
020	DC 4	053	5	086	V	119	w	152	ୟ	185	ି	218	ି	251	୧
021	Negative Acknowledge (NAK)	054	6	087	W	120	x	153	୭ୋ	186	ି	219	ି	252	୩
022	Synchronous Idle	055	7	088	X	121	y	154	୭୪	187	ି	220	ି	253	୩
023	End Of Transmission Block	056	8	089	Y	122	z	155	୦	188	ି	221	ି	254	ି
024	Cancel	057	9	090	Z	123	{	156	୰୍ଲେ	189	ି	222	ି	255	-
025	End Of Medium	058	:	091	[	124		157	୰୍ଲେ	190	୰୍ଲେ	223	ି		
026	Substitute	059	:	092	\	125	}	158	୴	191	ି	224	୶		
027	Escape (ESC)	060	<	093	]	126	~	159	ି	192	ି	225	ି		
028	File Separator	061	=	094	^	127 (DEL)	ି	160	ା	193	ି	226	୶		
029	Group Separator	062	>	095	-	128	ୱ	161	ି	194	ି	227	୶		
030	Record Separator	063	?	096	ୱ	129	୹	162	୰୍ଲେ	195	ି	228	୶		
031	Unit Separator	064	@	097	a	130	ୱ	163	୰୍ଲେ	196	ି	229	୶		
032	SPACE (SP)	065	A	098	b	131	ା	164	ି	197	ି	230	ି		



Più da vicino...

068	D	101	e
069	E	102	f
070	F	103	g
071	G	104	h
072	H	105	i

- Scrivere un programma che, letto un carattere minuscolo, lo riporta in maiuscolo
- Osservazione
  - Proviamo a stampare il valore ASCII di un carattere minuscolo e del suo corrispettivo maiuscolo



```
#include <stdio.h>
#include <stdint.h>

int32_t main () {
    uint8_t c1, c2;

    printf("Inserisci un carattere minuscolo: ");
    scanf("%c",&c1);

    printf("Inserisci lo stesso carattere, ma in maiuscolo: ");
    scanf(" %c",&c2);

    printf("Il carattere inserito e' %c e in ASCII vale %u\n", c1, c1);
    printf("La sua rappresentazione maiuscola e' %c e in ASCII vale %u\n\n", c2, c2);

    return 0;
}
```

```
[Marcos-MacBook-Air:Desktop santa$ ./exe
Inserisci un carattere minuscolo: g
Inserisci lo stesso carattere, ma in maiuscolo: G
Il carattere inserito e' g e in ASCII vale 103
La sua rappresentazione maiuscola e' G e in ASCII vale 71
```

```
Marcos-MacBook-Air:Desktop santa$ ./exe
Inserisci un carattere minuscolo: a
Inserisci lo stesso carattere, ma in maiuscolo: A
Il carattere inserito e' a e in ASCII vale 97
La sua rappresentazione maiuscola e' A e in ASCII vale 65
```

```
[Marcos-MacBook-Air:Desktop santa$ ./exe
Inserisci un carattere minuscolo: b
Inserisci lo stesso carattere, ma in maiuscolo: B
Il carattere inserito e' b e in ASCII vale 98
La sua rappresentazione maiuscola e' B e in ASCII vale 66
```



Dato un carattere minuscolo, *dato*  
Il corrispettivo maiuscolo è: *dato-32*