



POLITECNICO
MILANO 1863

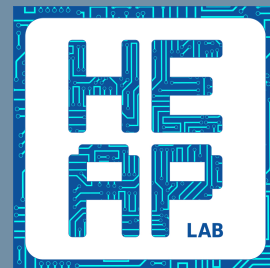
Informatica ed Elementi di Informatica Medica

A.A. 2017-18

Laboratorio n°1

Dott. Michele Zanella

Ing. Gian Enrico Conti



- Contatti:
 - gianenrico.conti@mail.polimi.it
 - michele.zanella@polimi.it
 - [HEAP Lab](#) – Campus Leonardo, via Golgi 39, Edificio 21, Piano 1, Ufficio 4, +39 02 2399 **9613**
(mandatemi una mail per accordarci su giorno e ora)
- Sito web del laboratorio:
 - <http://zanella.faculty.polimi.it/teaching/informatica-ed-elementi-di-informatica-medica/>
- **Nota per le mail:**
Oggetto: *[IEIM] il vostro oggetto*

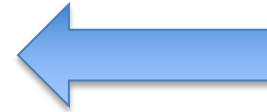
- Calendario laboratori

Data	Orario	Squadra	Aula	Resp.	Programma
16/03/18	14:15-17:15	A	L26.14	Conti	Lab 1
19/03/18	15:15-18:15	B	L26.14	Zanella	
23/03/18	14:15-17:15	A	L26.14	Conti	Lab 2
26/03/18	15:15-18:15	B	L26.14	Zanella	
06/04/18	14:15-17:15	A	L26.14	Conti	Lab 3
09/04/18	15:15-18:15	B	L26.14	Zanella	
13/04/18	14:15-17:15	A	L26.14	Conti	Lab 4
16/04/18	15:15-18:15	B	L26.14	Zanella	
20/04/18	14:15-17:15	A	L26.14	Conti	Lab 5
23/04/18	15:15-18:15	B	L26.14	Zanella	

- **Argomenti**

- Introduzione sull'ambiente di sviluppo, compilazione di un programma, debugging (gdb)
- Buone norme di programmazione in C
- Semplici algoritmi e programmi di esempio
- Operazioni con le stringhe
- Array e matrici
- I/O
- Puntatori
- File I/O
- Ricorsione
- (Elaborazione delle immagini)

- Processo con il quale una serie di istruzioni scritte in un determinato linguaggio di programmazione (*codice sorgente*, e.g. `.c` o `.h`) viene tradotto in istruzioni di un altro linguaggio (*codice oggetto*, e.g. `.exe` o `.o`), quest'ultimo comprensibile ed eseguibile dalla macchina.
- Tramite interfaccia grafica (IDE)
- Tramite interfaccia a riga di comando (CLI)



Compilazione (CLI - Linux)

- Recarsi tramite *Shell* nella cartella dove si trova il sorgente

```
$ gcc <file_sorgente.c> -o <file_eseguibile>
```

- Per utilizzare il debugger (i.e., gdb)

```
$ gcc <file_sorgente.c> -o <file_eseguibile> -g -O0
```

- Con file sorgenti multipli

```
$ gcc <file_1.c> <file_2.c> -o <file_eseguibile>
```

- Generare *Messaggi di Warning*

```
$ gcc -Wall <file.c> -o <file_eseguibile>
```

Esecuzione (CLI - Linux)

- Nella cartella dove è presente il file eseguibile

```
$ ./<file_eseguibile>
```

Errori comuni...

- Controllare di aver salvato il file sorgente come .c
- Controllare la cartella dove avete salvato il file
- Controllare i ';' e le *parentesi graffe*

Aprire un editor di testo (e.g., gedit):

```
int main(){  
    int a = 2;  
    return 0;  
}
```

Compilare ed eseguire

Primi passi (cont'd)

Partendo dal file scritto in precedenza, inseriamo degli errori e prendiamo familiarità con il debugging.

```
int main(){  
    int a = 2;  
    return 0;  
}
```

- Inserire un *breakpoint* alla riga 2 ed avviare il debug.
- Verificare il valore di *a* durante l'esecuzione

Esercizio 1: Calcolo IVA

Scrivere un programma che, dato un costo ivato di un articolo (in float), calcoli il costo senza IVA (IVA al 22%).

Esempio:

Inserisci: 122 -> *100.00 + IVA 22.00*

Hints:

- Il costo finale è determinato dal costo + IVA, dove IVA è il 22% del costo.

Esercizio 2: Volume di un cilindro

Scrivere un programma che, dati il raggio e l'altezza di un cilindro, ne calcoli il volume.

Volume del cilindro: $V = \pi r^2 h$

Hints:

- Consideriamo $\pi = 3,14$ per semplicità

Esercizio 3: Volume di un cono

Scrivere un programma che, dati il raggio e l'altezza di un cono, ne calcoli il volume.

Volume del cono: $V = \frac{\pi r^2 h}{3}$

Hints:

- Consideriamo $\pi = 3,14$ per semplicità

Esercizio 4: Quadrato perfetto

Dato in input un numero intero $A > 0$, verificare se A è un quadrato perfetto, cioè se esiste un numero B tale che $A = B*B$.

Hints:

- Utilizzare il flowchart prodotto dagli esercizi di esempio [qui](#) (Es 3.1)

Esercizio 5: MDC di un numero

Calcolare il Massimo Comune Divisore di due numeri interi positivi A e B.

Esempio:

90, 27 $\rightarrow$ 9

91, 27 $\rightarrow$ 1

Hints:

- Utilizzare il flowchart prodotto dagli esercizi di esempio [qui](#) (Es 3.2)

Esercizio 6: Anno bisestile

Scrivere un programma che, dato un anno, dica se è bisestile o meno.

Esempio:

Anno: 1777

-> *non è bisestile!*

Hints:

- Un anno è bisestile se è multiplo di 4. Se però è multiplo di 100 non è bisestile, con l'eccezione dei multipli di 400 che sono bisestili.
- Utilizzare l'operatore % per calcolare il resto della divisione intera
- Combinare le varie condizioni in costrutti condizionali sequenziali, annidati o mediante operatori AND e OR logici

Esercizio 7: Numeri primi di Mersenne

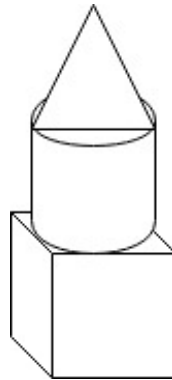
Dato in input un numero primo P , verificare che il numero di Mersenne $M = 2^P - 1$ sia anch'esso primo e nel caso stamparlo.

Hints:

- Utilizzare il flowchart prodotto dagli esercizi di esempio [qui](#) (Es 3.5)
- Avete già visto come verificare se un numero è primo

Esercizio 8: Volume complesso

Scrivere un programma che calcoli il volume di un solido formato da un cono, di altezza e raggio noto, posizionato sopra un cilindro, di altezza e raggio noto, il quale è infine posizionato sopra un cubo di lato noto.



Hints:

- Riutilizzare le formule precedenti