

Fondamenti di Informatica e Laboratorio T-AB  
Ingegneria Elettronica e Telecomunicazioni

---

# Lab 06

## Funzioni semplici

# Esercizio 1 - Funzioni

---

Codificare in C la funzione

```
int max(int a, int b)
```

che restituisce il massimo valore tra due interi.

Codificare in C la funzione

```
int max3(int a, int b, int c)
```

che restituisce il massimo valore fra tre interi, sfruttando la funzione max definita precedentemente.

Definire un possibile main che prende in ingresso i tre valori dall'utente e ne stampa il massimo.

# Esercizio 1 – Le due funzioni

---

```
int max(int a, int b) {  
    if (a>b)  
        return a;  
    else return b;  
}
```

```
int max3(int a, int b, int c) {  
    int max_di_due;  
    max_di_due = max(a,b);  
    return max(max_di_due,c);  
}
```

# Esercizio 1 – Il main

---

```
void main() {
    int a, b, c, d;

    printf("Inserire 3 tre interi: ");
    scanf("%d%d%d", &a, &b, &c);

    d = max2(a, b, c);
    printf("Il loro massimo e' : %d\n", d);
}
```

# Esercizio 1 – Il programma

---

```
#include <stdio.h>

void max(int a, int b) {
    ...
}

void max3(int a, int b, int c) {
    ...
}

void main() {
    ...
}
```

# Esercizio 1 – Una variante

---

```
int max(int a, int b) {  
    ...  
}
```

```
int max3(int a, int b, int c) {  
    return max(max(a,b),c);  
}
```

# Esercizio 2 - Funzioni

---

Si progettino e si realizzino due funzioni così definite:

```
float euro_to_dollari(float money)
float euro_to_lire(float money)
```

ognuna delle quali converte un valore in euro nella moneta corrispondente. A tal fine si supponga che:

1 € = 1.38 \$

1 € = 1936.27 £

Si progetti poi un programma che legge da input un valore intero, inteso come quantità di euro, e stampa la conversione in dollari ed in lire.

# Esercizio 3 - Funzioni

---

Si scriva una funzione

```
int somma_potenze(int a,int n);
```

che dati **a** e **n** deve calcolare  $\sum_{i=1}^n a^i$

A tal fine si scriva una funzione

```
int potenza(int x,int y);
```

che dati **x** e **y** deve calcolare **x<sup>y</sup>** usando come operazione primitiva il prodotto.



# Esercizio 4 - Funzioni

---

Creare una funzione `float square(float x)`. La funzione deve restituire il quadrato del parametro `x`.

Creare un'altra funzione, di nome `float cube(float x)`, che restituisce invece il cubo del valore `x`.

Progettare quindi e codificare un programma che legga un `float` da tastiera e restituisca il suo quadrato ed il suo cubo. Per calcolare il quadrato ed il cubo si devono utilizzare le due funzioni sopra definite.

# Esercizio 5 - Funzioni

---

Si scriva una funzione

```
int somma2(int n);
```

che dato  $n$  deve calcolare  $\sum_{i=1}^n \sum_{j=1}^i j$

A tal fine si sfrutti una funzione

```
int somma(int n);
```

che dato  $n$  deve calcolare  $\sum_{k=1}^n k$

# Esercizio 6 - Funzioni

---

Codificare in C la funzione

`int min_to_sec(int a)` che considera il parametro `a` come minuti e restituisce il numero di secondi corrispondente.

Codificare in C la funzione

`int ore_to_sec(int a)` che considera il parametro `a` come ammontare di ore, e restituisca il numero di secondi corrispondente. Si utilizzi la funzione definita precedentemente.

Definire un possibile main che prende in ingresso tre valori interi, rappresentanti ore, minuti e secondi della durata di un CD Audio. Il programma deve stampare il valore corrispondente in secondi.

# Esercizio 7 - Funzioni

---

Codificare in C la funzione

`int ipotenusa(int a, int b)` che, dati i cateti `a` e `b` di un triangolo rettangolo, restituisce il valore dell'ipotenusa.

A tal scopo si utilizzi il Teorema di Pitagora:

$$Ipotenusa = \sqrt{a^2 + b^2}$$

Per calcolare la radice quadrata si utilizzi la funzione di libreria `sqrt(x)`. Per utilizzare quest'ultima si aggiunga la direttiva `#include <math.h>` e l'opzione “**-lm**” per il linker.

Definire un possibile `main` che legga da tastiera due valori che rappresentino i cateti di un triangolo rettangolo, e stampi il valore dell'ipotenusa.

# Esercizio 8 - Funzioni

---

Codificare in C la funzione

`int perimetro(int a, int b, int c)` che, dati i lati `a, b, c` di un triangolo, ne calcola il perimetro.

Codificare in C la funzione

`float area(int a, int b, int c)`

che restituisce l'area di un triangolo i cui lati misurano `a, b, c`.

A tal scopo si usi la formula di Erone:

$$Area = \sqrt{p(p-a)(p-b)(p-c)}$$

Dove `p` è la metà del perimetro. A tal scopo si includa l'header `<math.h>` e si utilizzi la funzione `sqrt(x)`.

Definire un possibile `main` che prenda in ingresso i tre lati di un triangolo e stampi perimetro ed area.

# Esercizio 9 - Funzioni

---

Codificare in C la funzione `int primo(int x)` che restituisce:

1 se  $x$  è un numero primo

0 altrimenti.

Si utilizzi a tal proposito l'operatore modulo (%).

Si progetti un programma che legge da tastiera un numero  $N$ , e stampa a video tutti i numeri primi compresi tra 0 e  $N$ .