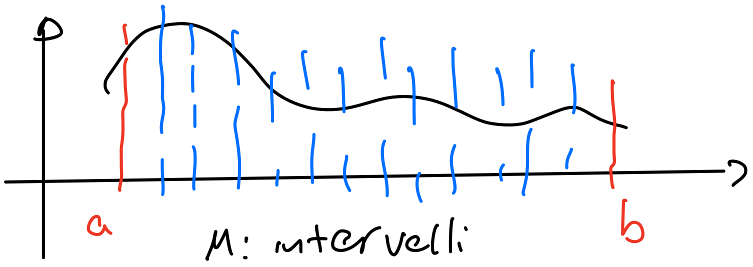
Integrali

$$I = \int_a^b f(x)\,dx$$



$$I = \sum_{i,0}^{M-1} I_i \qquad i \in [0, M-1]$$

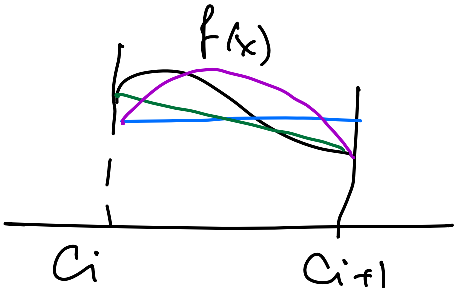
$$I_i = \int_{c_i}^{c_{i+1}} f(x)\,dx$$

$c_0 = a$

$c_M = b$

$$c_i = a + i\,\Delta x$$

$$\Delta x = (b-a)/M$$



$$I_i = \int_{c_i}^{c_{i+1}} f(x)\,dx$$

$f(x)$: note oppure sconosciute

conosciamo $f(x_i)$ in $x_i = \{x_1 -- x_n\}$

approssimare $f(x)$ con funzione $L(x, \vec{a})$

$P_2(x) = ax^2 + bx + c$ $\qquad L(x, a, a c)$

Spesso si ricorre a polinomi per approssimare $f(x)$

Polinomi di Lagrange

$$L_{n-1} = \sum_{i,1}^{n} f(x_i) \prod_{\substack{j=1 \\ j\neq i}}^{n} \frac{(x-x_j)}{x_i - x_j}$$

polinomio grado n

conosciamo $f(x_i)$ in $\{x_1 -- x_n\}$

$n=1$

$$L_0(x) = \sum_{1}^{1} {}_i f(x_i) \prod_{\substack{j=1 \\ j\neq i}}^{1} \frac{x-x_j}{x_i-x_j}$$

$$= f(x_1) \quad \text{costante}$$

$n=2$

$$L_1(x) = \sum_{1}^{2} {}_i f(x_i) \prod_{\substack{j=1 \\ i\neq j}}^{2} \frac{x-x_j}{x_i-x_j} =$$

$$= f(x_1) \prod_{\substack{j=1 \\ j\neq 1}}^{2} \frac{x-x_j}{x_i-x_j} + f(x_2) \prod_{\substack{j=1 \\ j\neq 2}}^{2} \frac{x-x_j}{x_i-x_j}$$

$$= f(x_1) \frac{x-x_2}{x_1-x_2} + f(x_2) \frac{x-x_1}{x_2-x_1}$$

$$= \underbrace{\left(\frac{f(x_1)-f(x_2)}{x_1-x_2}\right)}_{\substack{\text{costante} \\ m}} x + \underbrace{\frac{f(x_2)x_1 - f(x_1)x_2}{x_1-x_2}}_{\substack{\text{costante} \\ q}}$$

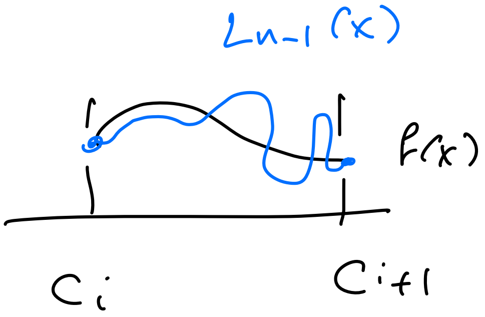
$$L_1(x_1) = \frac{f(x_1) - f(x_2)}{x_1-x_2} x_1 + \frac{f(x_2)x_1 - f(x_1)x_2}{x_1-x_2}$$

$$= \frac{1}{x_1-x_2}\left[f(x_1)x_1 - f(x_1)\, x_2 \right] =$$

$$= f(x_1) \frac{x_1-x_2}{x_1-x_2} = f(x_1)$$

$n=3$

$L_2(x)$:



$$\Delta_n = |f(x) - L_{n-1}(x)| = \leq \sup_{z\in[a,b]} |f^{(n)}(z)| \frac{\prod_{j=1}^{n} (x-x_j)}{n!}$$

$$I_i = \int_{c_i}^{c_{i+1}} f(x)\,dx \qquad I_i^{(n)} = \int_{c_i}^{c_{i+1}} L_{n-1}(x)\,dx$$

Errore o resto del calcolo integrale

$$\delta_i^{(n)} = |I_i - I_i^{(n)}| \le \int_{c_i}^{c_{i+1}} |\Delta_n|\,dx$$

Fattore numerico

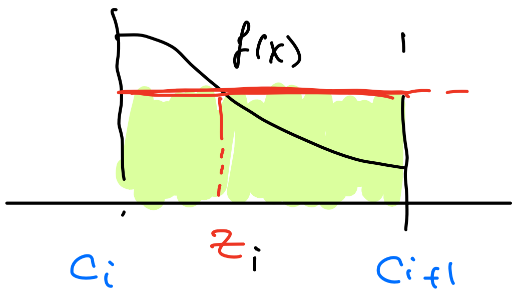
$$\delta^{(n)} = \sum_i \delta_i^{(n)} = A_n(f)\,\frac{D_n}{2^{n+1}}\,\frac{(b-a)^{n+1}}{M^n}$$

$$A_n(f) = \left|\sup_{z\in[a,b]} f^{(n)}(z)\right|$$

M: numero di intervalli

n: grado del polinomio
numero di punti in cui conosco $f(x_i)$

Metodo dei Rettangoli



$z_i \in [c_i, c_{i+1}]$

$c_i = f(z_i)$

z interna $[c_i, c_{i+1}]$

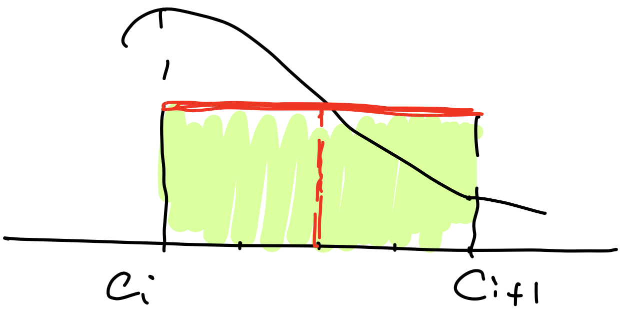
polinomio di grado $\emptyset$

$$I_i = c\cdot\Delta x = c\left(\frac{b-a}{M}\right)$$

$$I = \sum_{i=0}^{M-1} c_i\,\frac{(b-a)}{M} = \frac{b-a}{M}\sum_{i=0}^{M-1} c_i$$

Spesso usato se conosciamo $f(x)$ in alcuni punti z

Metodo del Punto di Mezzo



$$z_i = \frac{c_i + c_{i+1}}{2}$$

$$= a + i\,\Delta x + 0.5\,\Delta x$$

Polinomio di grado 0

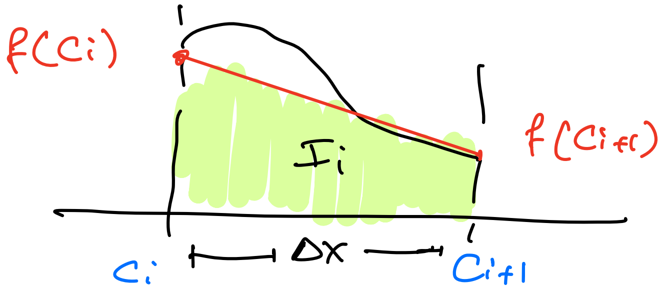
$$I = \sum_{i=0}^{n-1} I_i = \sum_{i=0}^{n-1} f(z_i)\cdot\Delta x$$

$$= \frac{b-a}{n} \sum_{i=0}^{n-1} f(z_i)$$

Errore

$$\delta^{(1)} \simeq \frac{1}{24} \left| \sup_{z\in[a,b]} f''(z) \right| \frac{(b-a)^3}{n^2}$$

Metodo del Trapezio



Polinomi grado 1

$n=1$

$$\Delta x = \frac{b-a}{n}$$

$$I_i \simeq \Delta x \cdot \frac{1}{2} \left[f(c_{i+1}) + f(c_i) \right]$$

area del trapezio

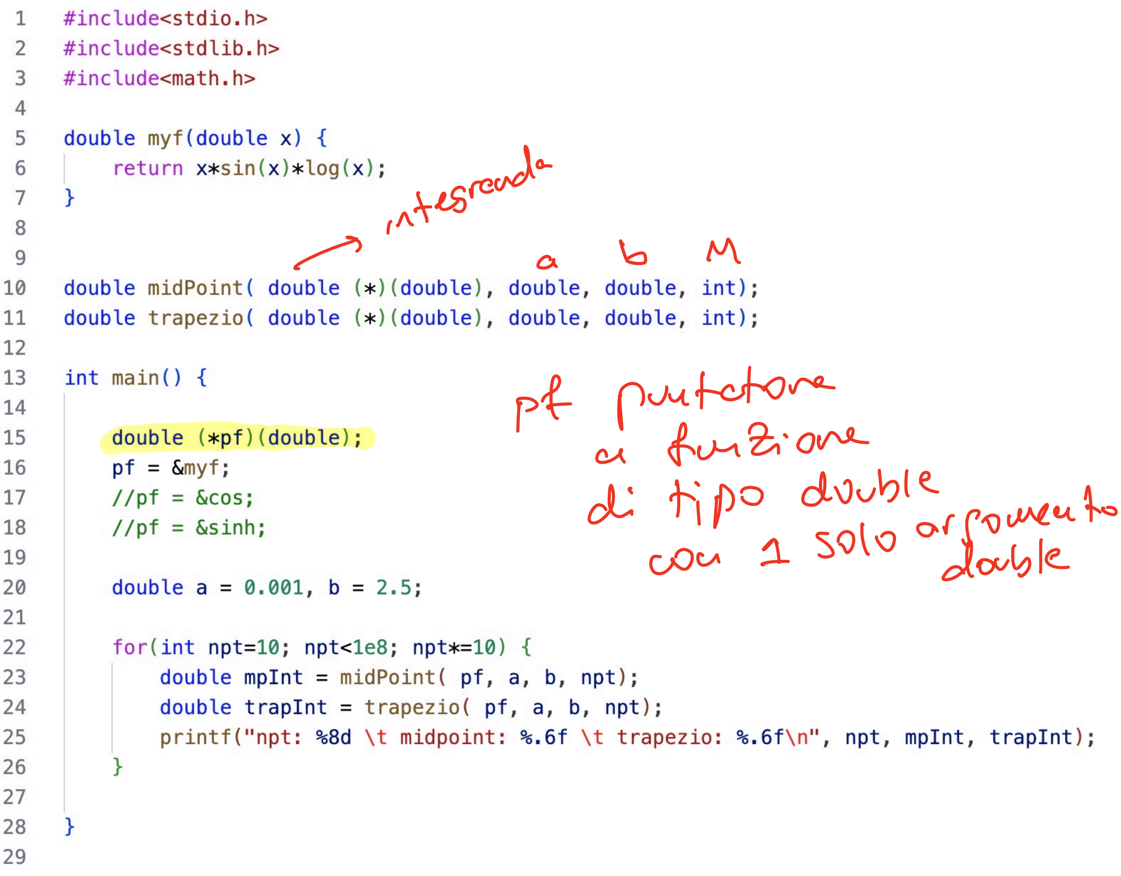
$$I = \sum_i I_i$$

$$\delta^{(2)} = \frac{1}{12} \left| \sup_{z\in[a,b]} f''(z) \right| \frac{(b-a)^3}{n^2}$$

$$c_i = a + i\,\Delta x$$

$$c_{i+1} = a + (i+1)\,\Delta x.$$

punto di mezzo $\quad z_i = a + i\,\Delta x + 0.5\,\Delta x$



```c
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

double myf(double x) {
    return x*sin(x)*log(x);
}


double midPoint( double (*)(double), double, double, int);
double trapezio( double (*)(double), double, double, int);

int main() {

    double (*pf)(double);
    pf = &myf;
    //pf = &cos;
    //pf = &sinh;

    double a = 0.001, b = 2.5;

    for(int npt=10; npt<1e8; npt*=10) {
        double mpInt = midPoint( pf, a, b, npt);
        double trapInt = trapezio( pf, a, b, npt);
        printf("npt: %8d \t midpoint: %.6f \t trapezio: %.6f\n", npt, mpInt, trapInt);
    }

}
```

integranda

a b M

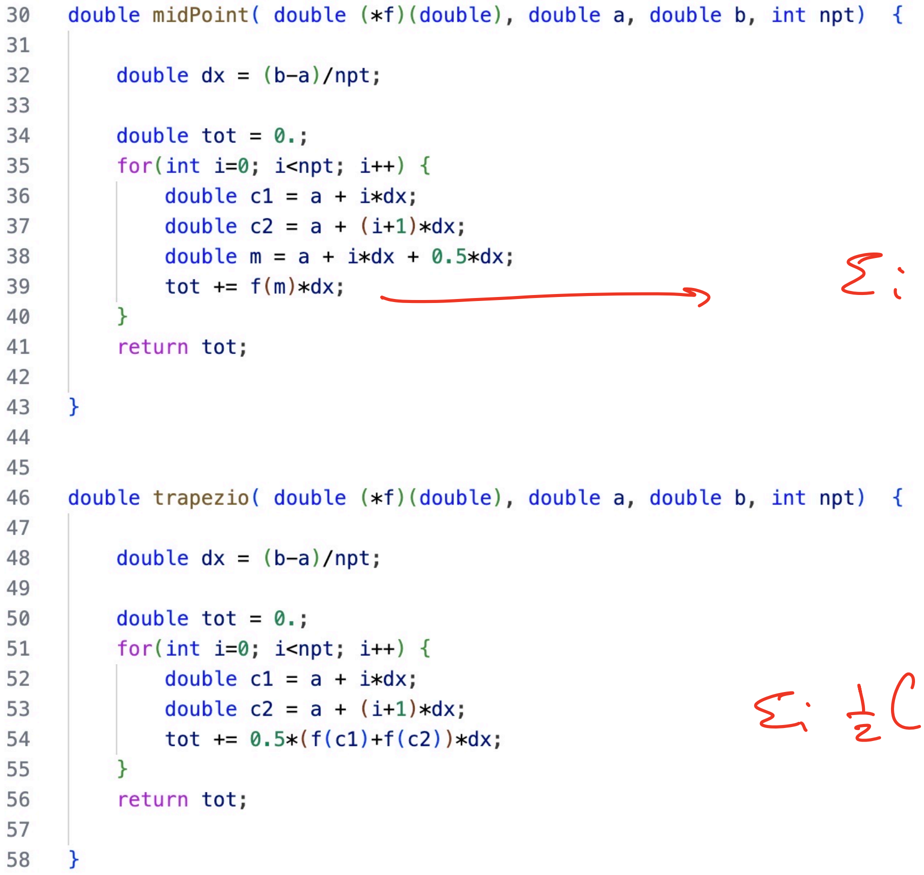
pf puntatore a funzione di tipo double con 1 solo argomento double

puntatore a variabili

double x;

double* P;

P = &x;

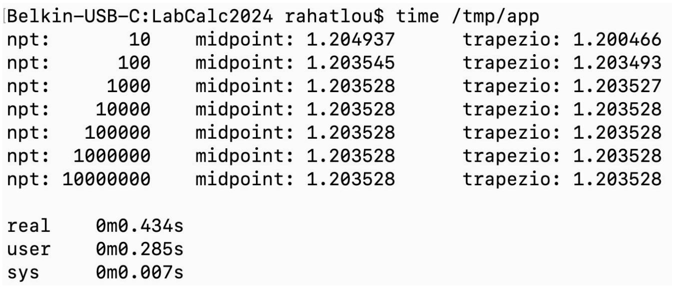


```c
double midPoint( double (*f)(double), double a, double b, int npt)  {

    double dx = (b-a)/npt;

    double tot = 0.;
    for(int i=0; i<npt; i++) {
        double c1 = a + i*dx;
        double c2 = a + (i+1)*dx;
        double m = a + i*dx + 0.5*dx;
        tot += f(m)*dx;
    }
    return tot;

}


double trapezio( double (*f)(double), double a, double b, int npt)  {

    double dx = (b-a)/npt;

    double tot = 0.;
    for(int i=0; i<npt; i++) {
        double c1 = a + i*dx;
        double c2 = a + (i+1)*dx;
        tot += 0.5*(f(c1)+f(c2))*dx;
    }
    return tot;

}
```

$\sum_i f(z_i)\Delta x$

z_i : punto di mezzo

$\sum_i \frac{1}{2}(f(c_{i+1}) + f(c_i)) \cdot \Delta x$



```
Belkin-USB-C:LabCalc2024 rahatlou$ time /tmp/app
npt:       10     midpoint: 1.204937      trapezio: 1.200466
npt:      100     midpoint: 1.203545      trapezio: 1.203493
npt:     1000     midpoint: 1.203528      trapezio: 1.203527
npt:    10000     midpoint: 1.203528      trapezio: 1.203528
npt:   100000     midpoint: 1.203528      trapezio: 1.203528
npt:  1000000     midpoint: 1.203528      trapezio: 1.203528
npt: 10000000     midpoint: 1.203528      trapezio: 1.203528

real    0m0.434s
user    0m0.285s
sys     0m0.007s
```