

Laboratorio di Programmazione e Calcolo

Canale 2

Appunti del corso

Simone Cacace e Giuseppe Visconti

Dipartimento di Matematica
Sapienza Università di Roma

Anno Accademico 2025–2026

Laboratorio – Matematica al calcolatore

Esercizi

E0. Valutazione di polinomi

Scrivere un programma che legge da tastiera i coefficienti di un polinomio di grado n e il valore x , e calcola il valore del polinomio usando sia la forma diretta sia lo schema di Horner.

Spiegazione matematica. Un polinomio di grado n si scrive:

$$P(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n.$$

La valutazione diretta richiede n potenze e n moltiplicazioni, mentre lo *schema di Horner* riscrive il polinomio come:

$$P(x) = a_0 + x(a_1 + x(a_2 + \cdots + x(a_{n-1} + xa_n) \dots)).$$

Questo riduce il numero di operazioni e migliora la stabilità numerica.

Suggerimento: confrontare i due metodi per valori grandi di x o per polinomi con coefficienti di segno alterno, dove gli errori di arrotondamento possono amplificarsi.

E1. Serie di Taylor

Approssimare e^x usando la serie di Taylor fino a che il termine aggiunto è minore di una tolleranza fissata dall'utente.

Spiegazione matematica. La serie di Taylor di e^x centrata in 0 è:

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}.$$

Il termine k -esimo vale $t_k = \frac{x^k}{k!}$. Si interrompe la somma quando $|t_k| < \varepsilon$, dove ε è la tolleranza scelta.

Suggerimento: calcolare ogni termine a partire dal precedente:

$$t_{k+1} = t_k \cdot \frac{x}{k+1},$$

così si evitano potenze e fattoriali. Confrontare il risultato con la funzione `exp(x)` di `math.h`.

E2. Limiti notevoli numerici

Scrivere un programma che calcoli numericamente i seguenti limiti notevoli:

$$\lim_{x \rightarrow 0} \frac{\sin(x)}{x}, \quad \lim_{x \rightarrow 0} \frac{1 - \cos(x)}{x^2}, \quad \lim_{x \rightarrow 0} (1 + x)^{1/x}.$$

Spiegazione matematica. I tre limiti teorici valgono:

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1, \quad \lim_{x \rightarrow 0} \frac{1 - \cos x}{x^2} = \frac{1}{2}, \quad \lim_{x \rightarrow 0} (1 + x)^{1/x} = e.$$

Tuttavia, quando x è molto piccolo, le sottrazioni tra numeri simili (come $1 - \cos x$) generano perdita di significato numerico.

Suggerimento: usare valori di x decrescenti (ad esempio $10^{-1}, 10^{-2}, \dots, 10^{-8}$) e stampare il valore della funzione e l'errore rispetto al limite teorico. Per il secondo limite confrontare le due forme:

$$f_1(x) = \frac{1 - \cos(x)}{x^2}, \quad f_2(x) = \frac{2 \sin^2(x/2)}{x^2},$$

e notare che f_2 è numericamente più stabile.

E3. Calcolo di serie

Scrivere un programma che calcoli le somme parziali delle seguenti serie notevoli:

1. Serie geometrica: $\sum_{k=0}^{\infty} q^k = \frac{1}{1-q}$, con $|q| < 1$
2. Serie di Leibniz per π : $\sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1} \cdot 4 = \pi$

Spiegazione matematica. Ogni serie converge a un valore noto S . Le somme parziali

$$S_n = \sum_{k=0}^n a_k$$

si avvicinano progressivamente a S . Gli errori assoluto e relativo sono:

$$E_a = |S_n - S|, \quad E_r = \frac{|S_n - S|}{|S|}.$$

Suggerimento: allocare un array per i valori S_n , e stampare per ogni n il valore della somma parziale e gli errori. Osservare la velocità di convergenza delle tre serie: la geometrica converge rapidamente, la di Leibniz molto lentamente.

E4. Massimo comune divisore e minimo comune multiplo

Implementare un programma che calcola il massimo comune divisore (MCD) e il minimo comune multiplo (m.c.m.) di due numeri interi positivi inseriti dall'utente.

Spiegazione matematica. L'algoritmo di Euclide calcola il MCD di a, b eseguendo divisioni successive:

$$\begin{cases} r_0 = a, & r_1 = b, \\ r_{k+1} = r_{k-1} \bmod r_k, \end{cases}$$

fino a $r_k = 0$. L'ultimo resto non nullo è il MCD. Il minimo comune multiplo si ottiene da:

$$\text{m.c.m.}(a, b) = \frac{a \cdot b}{\text{MCD}(a, b)}.$$

Suggerimento: verificare il corretto funzionamento anche per numeri primi o multipli diretti (ad esempio 15 e 45).

E5. Stima di π con Monte Carlo

Un'applicazione classica dei numeri pseudocasuali è la stima di π mediante il metodo Monte Carlo. Si consideri il quadrato $[0, 1] \times [0, 1]$ e il quarto di cerchio $x^2 + y^2 \leq 1$. Generare punti uniformemente nel quadrato e calcolare la proporzione di punti che cadono nel quarto di cerchio. Verificare che è approssimativamente $\pi/4$.

Spiegazione matematica. Il rapporto tra l'area del quarto di cerchio e quella del quadrato è:

$$\frac{A_{\text{cerchio}}}{A_{\text{quadrato}}} = \frac{\pi r^2 / 4}{r^2} = \frac{\pi}{4}.$$

Se N_{cerchio} è il numero di punti nel quarto di cerchio e N il numero totale di punti, allora:

$$\pi \approx 4 \cdot \frac{N_{\text{cerchio}}}{N}.$$

Suggerimento: usare `rand()` per generare $x, y \in [0, 1]$. Contare quanti punti soddisfano $x^2 + y^2 \leq 1$. Aumentando N , la stima si stabilizza attorno a π

E7. Grafica

Se interessati, dotare di veste grafica gli esercizi da E0 a E3.

Suggerimenti:

- Esportare i dati su file e plottarli con `gnuplot`.
- Per E0, disegnare $P(x)$ in un intervallo di valori di x .
- Per E1 e E3, mostrare la convergenza della serie o dell'approssimazione.
- Provare la versione con pipe per aggiornare il grafico in tempo reale.