Laboratorio di Calcolo per Fisici,
Esame del 14/07/2025
A.A. 2024/2025

Nome ______________________ Cognome ______________________

Matricola ______________________ ☐ Ritirato/a

Lo scopo di questa esercitazione è scrivere un programma in C e uno script in python seguendo la traccia riportata di seguito. Si tenga presente che:

1. Per svolgere il compito si hanno a disposizione 3 ore.
2. Si possono usare libri di testo, prontuari e gli appunti ma non è ammesso parlare con nessuno né utilizzare cellulari, tablet o laptop, pena l'annullamento del compito.
3. Il programma va scritto e salvato esclusivamente sul computer del laboratorio, a cui si deve accedere utilizzando come username **studente** e come password **informatica**
4. **Tutti i file vanno salvati in una cartella chiamata `ELCGEN_NOME_COGNOME` nella home directory**, dove `NOME` e `COGNOME` indicano rispettivamente il tuo nome e cognome. Ad esempio lo studente *Marco Rossi* deve creare una cartella chiamata `ELCGEN_MARCO_ROSSI` contenente tutti i file specificati nel testo. **Tutto ciò che non si trova all'interno della cartella suddetta non verrà valutato.** In tutti i programmi e script inserisci all'inizio un commento con il tuo nome, cognome e numero di matricola.
5. **Dovete consegnare il presente testo indicando nome, cognome e numero di matricola** (vedi sopra), barrando la casella "Ritirato/a" se ci si vuole ritirare, ovvero se non si vuole che la presente prova venga valutata.

Si consideri un'equazione di secondo grado del tipo:

$$x^2 + bx + c = 0 \tag{1}$$

Come noto le due radici (o **zeri**) di questa equazione si possono trovare come segue:

$$x_{1,2} = \frac{-b \pm \sqrt{\Delta}}{2} \tag{2}$$

dove $\Delta = b^2 - 4c$, e si assume $\Delta \geq 0$. Tuttavia questa soluzione, che nel seguito indicheremo come metodo A, può essere molto inaccurata numericamente. Un'alternativa numericamente migliore è quella di scrivere le due soluzioni x_1, x_2 nel seguente modo (metodo B):

$$x_1 = \begin{cases} -\frac{b+\sqrt{\Delta}}{2} & \text{if } b \geq 0 \\ \frac{-b+\sqrt{\Delta}}{2} & \text{if } b < 0 \end{cases}$$

e

$$x_2 = \begin{cases} 0 & \text{if } x_1 = 0 \\ c_0/x_1 & \text{if } x_1 \neq 0 \end{cases} \tag{3}$$

Lo scopo di questo prova d'esame è scrivere un codice che generi casualmente `NZERI` coppie di zeri reali (x_1, x_2), calcoli l'equazione di secondo grado di cui sono soluzione, e poi la risolva con il metodo A e il metodo B. Si ricorda che i coefficienti b e c dell'equazione 1 si possono trovare a partire da x_1 e x_2 applicando le formule:

$$\begin{aligned} b &= -(x_1 + x_2) \\ c &= x_1 \cdot x_2 \end{aligned} \tag{4}$$

Dopodiché il programma confronterà le due coppie di zeri ottenuti numericamente con quelli esatti, valutando l'errore relativo ai due metodi (ϵ^A/ϵ^B). Ripetendo quest'operazione un numero `NZERI` di volte, si può ottenere una stima della diversa precisione dei due metodi, calcolando la frequenza con cui si trova un certo errore ϵ^A o ϵ^B, e graficandone l'istogramma.

Definendo x_1^A e x_2^A le coppia di zeri ottenuti numericamente con il metodo A, dove $x_1^A < x_2^A$, e x_1^e e x_2^e la coppia di zeri esatti generati casualmente, con $x_1^e < x_2^e$, l'errore relativo ϵ^A si definisce come:

$$\epsilon^A = \left|\frac{x_1^A - x_1^e}{x_1^e}\right| + \left|\frac{x_2^A - x_2^e}{x_2^e}\right|; \tag{5}$$

chiaramente la definizione è analoga per ϵ^B.

► **Prima parte (Programma in c):**

Si realizzi un programma in C, chiamato `nome_cognome.c` (tutto minuscolo, senza eventuali spazi, accenti o apostrofi) che calcoli la frequenza degli errori relativi per i due approcci numerici A e B e salvi tali frequenze in un file da utilizzare nella seconda parte per generare un grafico in python. In particolare il programma dovrà:

1. Definire tramite delle macro il numero di coppie di zeri "esatti" da generare, `NZERI` = 10000, e il numero di bin per l'istogramma, `BINS` = 20.
2. **Tutti i numeri reali contenuti nel programma vanno salvati in doppia precisione.**
3. Generare casulalmente coppie di zeri x_1^e e x_2^e usando le seguenti formule:

$$x_1^e = 10^{30} + (\xi_1 - 0.5) \cdot 10^{20} \tag{6}$$
$$x_2^e = 10^{-30} + \xi_2 \cdot 10^{20} \tag{7}$$

   dove ξ_1 e ξ_2 sono due numeri casuali distribuiti uniformemente nell'intervallo $[0, 1)$.
4. Calcolare i coefficienti b e c della corrispondente equazione quadratica utilizzando l'equazione (4).
5. Risolvere l'equazione quadratica ottenuta usando prima il metodo A - eq. (2) - e poi il metodo B - eq. (3), ottenendo così due coppie di zeri approssimati x_1^A, x_2^A e x_1^B, x_2^B.
6. Ordinare le coppie di zeri esatti e approssimati, in modo che il primo sia minore del secondo.
7. Calcolare gli errori relativi ϵ^A e ϵ^B delle due diverse soluzioni numeriche usando l'Eq. 5.
8. Se l'errore relativo calcolato è diverso da 0, aggiornare l'istogramma corrispondente, che chiameremo `istoA` e `istoB`, incrementando di uno il k-esimo bin, dove k è definito da:

$$k = \lfloor \log_{10}(\epsilon) + \texttt{BINS} \rfloor \tag{8}$$

   dove $\lfloor x \rfloor$ indica la parte intera di x, $\log_{10}$ è il logaritmo in base 10 (da calcolarsi con la funzione `log10`) e `BINS` è il numero di bin dell'istogramma. Con questa definizione, k più grandi corrispondono a errori più grandi. Per esempio, se l'errore relativo ϵ_A per una determinata coppia di zeri è pari a 10^-6 va incrementato di uno il valore contenuto nel 14-esimo bin. *Suggerimento*: prima di aggiornare l'istogramma controllate che $k \geq 0$ e $k <$ `BINS`.
9. Stampare per le prime dieci coppie di zeri i valori esatti x_1^e e x_2^e generati casualmente al punto 2, e il valori di ϵ^A e ϵ^B corrispondenti, con 15 cifre significative. *Suggerimento:* per stampare un numero con 15 cifre significative potete usare `%.15G`.
10. Stampare su di un file chiamato `isto.dat` i due istogrammi `istoA` e `istoB` ottenuti dopo aver generato `NZERI` coppie di zeri. Il file deve contenere 3 colonne: la prima conterrà i valori di k e la seconda e la terza con i valori dei due istogrammi divisi per `NZERI`, cioè la frequenza con cui l'errore relativo è finito nel k-esimo bin per ciascuno dei due metodi. *Suggerimento*: Ci si aspetta che `istoA` abbia un picco intorno a $k = 13$, mentre `istoB` dovrebbe averlo intorno a $k = 4$.

Nello scrivere il programma si richiede che vengano implementate almeno le seguenti funzioni:

- `ordina_zeri()` che ordina i due zeri passati come argomento tramite un opportuno array, mettendo il più piccolo prima.
- `calc_err()` che calcola l'errore relativo ϵ^A/ϵ^B degli zeri ottenuti numericamente tramite l'equazione (5). Tale funzione prende come argomento due array, uno contenente gli zeri esatti e l'altro quelli ottenuti numericamente, e restituisce ϵ.
- `solve_quadratic()` che trova gli zeri con la formula A - Eq. (2). Tale funzione prende come argomenti due array: uno con i coefficienti dell'equazione quadratica e un altro dove verrano salvati i valori degli zeri calcolati numericamente.
- `solve_quadratic_best()` che trova gli zeri con il metodo B - Eq. (3). Tale funzione prende come argomenti due array: uno con i coefficienti dell'equazione quadratica e un altro dove verrano salvati i valori degli zeri calcolati numericamente.
- `updisto()` che prende come parametri un array contenente l'istogramma e un valore di ϵ, e utilizza l'equazione (8) per incrementare il bin dell'istogramma corrispondente al valore di ϵ.

► **Seconda parte (Python):** Utilizzando il file `isto.dat` generato eseguendo il programma scritto nella prima parte, creare con `python` uno script `nome_cognome.py` un grafico che mostri gli istogrammi delle frequenze per il metodo A e B, includendo un'opportuna legenda, label per gli assi x e y e un titolo. Salvare l'immagine in formato PNG con nome `istogramma.png`.