

Laboratorio di Calcolo per Fisici,
Esercitazione valutata del 19/12/2024 — I TURNO
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 ELCDIC_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 ELCDIC_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.

Una particella puntiforme di massa m è attaccata ad un estremo di una molla di costante elastica k il cui altro estremo è tenuto fisso. La particella, che è soggetta solo alla forza esercitata dalla molla, si muoverà di moto armonico con periodo $T_h = 2\pi\sqrt{m/k}$. La sua posizione $x(t)$ rispetto alla posizione di equilibrio della molla, e la sua velocità $v(t)$ al tempo t saranno date dalle seguenti equazioni:

$$\begin{aligned}x(t) &= x_0 \cos(wt) + \frac{v_0}{w} \sin(wt) \\v(t) &= -wx_0 \sin(wt) + v_0 \cos(wt)\end{aligned}\tag{1}$$

dove x_0 e v_0 sono, rispettivamente, la posizione e la velocità al tempo $t = 0$ e $w = \sqrt{k/m}$. L'energia meccanica del sistema pari a

$$\mathcal{H}(t) = \frac{1}{2}m v(t)^2 + \frac{1}{2}k x(t)^2\tag{2}$$

è una quantità conservata durante il moto della particella, ovvero il suo valore non cambia nel tempo. L'evoluzione nel tempo della particella (a intervalli di tempo Δt), ovvero l'andamento della sua posizione e della sua velocità, si può anche ottenere, a partire da x_0 e v_0 , in maniera numerica (approssimata) attraverso il seguente algoritmo iterativo (*Algoritmo di Eulero semi-implicito*):

$$\begin{aligned}v_n &= v_{n-1} - \frac{k}{m}x_{n-1}\Delta t \\x_n &= x_{n-1} + v_n\Delta t\end{aligned}\tag{3}$$

dove $n \geq 1$, $x_n = x(n\Delta t)$, $v_n = v(n\Delta t)$, ovvero x_n e v_n saranno la posizione e la velocità della particella al tempo $t = n\Delta t$. In tale algoritmo Δt è detto passo d'integrazione e tanto più questo è piccolo e tanto più accurata sarà la soluzione numerica rispetto a quella analitica. Si noti che per implementare tale algoritmo è sufficiente utilizzare due variabili double x_e e v_e che verranno aggiornate ad ogni iterazione come indicato nell'Eq. (3), avendo però cura di aggiornare prima v_e e di utilizzare quindi il nuovo valore di v_e per calcolare il nuovo valore di x_e .

► **Prima parte:**

Si realizzi un programma in C, chiamato `nome_cognome.c` (tutto minuscolo, senza eventuali spazi, accenti o apostrofi), che calcoli la posizione della particella e l'energia del sistema in funzione del tempo in maniera analitica e approssimata (utilizzando l'algoritmo di Eulero semi-implicito) per una particella di massa $m = 0.9$ e una molla di costante elastica $k = 1.3$. In tale programma si dovrà implementare un ciclo tramite il quale calcolare le posizioni e velocità della particella secondo l'Eq. (3). Ad ogni

iterazione andranno anche calcolate la posizione e la velocità della particella analitiche, e ogni 10 iterazioni si dovranno salvare la posizione numerica e quella analitica della particella in un file chiamato `traiettoria.dat`, e l'energia analitica e numerica del sistema in un file chiamato `energia.dat`. In tali file nella prima colonna andrà salvato il tempo e nella seconda e terza colonna i valori numerici ed analitici delle grandezze suddette (cioè posizione ed energia).

In particolare il programma dovrà:

1. Impostare il valore x_0 della posizione iniziale a 1 e quello della velocità iniziale v_0 a un numero casuale compreso tra $v_{min} = 0.5$ e $v_{max} = 1.5$.
2. Chiedere in input il valore di Δt , reiterando la richiesta di immissione se $\Delta t < 0.001$ oppure $\Delta t > 0.01$.
3. Aggiornare i valori delle variabili x_e e v_e secondo l'algoritmo in Eq. (3) per un numero di iterazioni n_{steps} pari alla parte intera di $T_h/\Delta t$.
4. Calcolare i valori analitici della posizione e della velocità secondo l'Eq. (1).
5. Calcolare i valori dell'energia secondo l'Eq. (2) utilizzando la soluzione analitica e quella numerica.
6. Salvare ogni 10 iterazioni sui file `traiettoria.dat` e `energia.dat` i valori numerici e analitici delle posizioni e dell'energia con 6 cifre dopo la virgola, nel formato discusso in precedenza.
7. I valori v_{min} , v_{max} , k e m devono essere definiti con delle macro.

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

- `input()` che richiederà l'immissione di Δt , che dovrà essere passato tramite un puntatore a `double`.
- `updSemiEuler()` che aggiornerà i valori delle variabili x_e e v_e secondo l'Eq. (3). Tali variabili saranno passate a tale funzione tramite due puntatori `double`. Questa funzione dovrà avere come argomento anche il passo d'integrazione Δt .
- `analytic()` che calcolerà il valore della posizione e della velocità analitici secondo l'Eq. (1). Tale funzione prenderà come argomenti due puntatori a `double` per poter restituire i valori analitici della posizione e della velocità. Inoltre, la funzione dovrà prendere come argomenti i valori x_0 e v_0 e il tempo all' n -esima iterazione $t = n\Delta t$.
- `energy()` che deve restituire come un `double` l'energia meccanica del sistema calcolata secondo l'Eq. (2). Tale funzione avrà come argomenti posizione e velocità della particella.

► Seconda parte:

Eseguire il programma sviluppato nella prima parte con $dt = 0.005$ e utilizzare il file `energia.dat` per creare con `python` un grafico che mostri le curve dell'energia, utilizzando una linea continua per i dati analitici e dei simboli per quelli numerici. Il grafico deve avere la legenda e opportune etichette per gli assi cartesiani. Lo script `python` si dovrà chiamare `nome_cognome.py`.