Laboratorio di Calcolo, Esercitazione Valutata del 22/12/2025

I turno, Anno accademico 2025-26

Nome: ______________________________ Cognome: ______________________________
Matricola: ______________________________ ☐ Ritirata/o

Lo scopo di questa prova d'esame è di 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. **Tutti i file vanno salvati in una cartella chiamata `EVDIC25T_NOME_COGNOME` nella home directory**, dove `NOME` e `COGNOME` indicano rispettivamente il vostro nome e cognome. Ad esempio lo studente *Nicolò Maria De Rossi Salò* deve creare una cartella chiamata `EVDIC25T_NICOLOMARIA_DEROSSISALO` 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 inserite all'inizio un commento con il vostro nome, cognome e numero di matricola.
4. **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.
5. **Per consegnare il compito** dovrete eseguire, all'interno della cartella creata in precedenza (come spiegato al punto 3), il seguente comando da terminale: `cp * /media/sf_esame/`

Consideriamo delle particelle caratterizzate da un orientamento nello spazio. L'orientazione di ciascuna particella sia descritta dall'angolo $\theta \in [0, \pi]$ rispetto a una direzione privilegiata fissata.
Ad ogni orientazione θ è associata una energia adimensionale

$$U(\theta) = -\cos\theta. \tag{1}$$

La probabilità che una particella assuma un angolo θ è descritta dalla distribuzione di probabilità

$$P(\theta, \lambda) = \frac{\lambda\, e^{\lambda \cos\theta} \sin\theta}{e^{\lambda} - e^{-\lambda}}, \tag{2}$$

dove $\lambda > 0$ è un parametro adimensionale che controlla il grado di allineamento delle particelle lungo la direzione privilegiata. Per stimare numericamente l'energia media, $\langle U \rangle$, si può utilizzare il metodo Monte Carlo, generando N_t valori di θ distribuiti secondo $P(\theta, \lambda)$ e calcolando

$$\langle U \rangle \simeq \frac{1}{N_t} \sum_{i=1}^{N_t} U(\theta_i). \tag{3}$$

Per generare i valori θ_i con la distribuzione richiesta si utilizzi il seguente algoritmo:

1. calcolare il valore massimo $P_{\max}(\lambda) = P(\theta_{\max}, \lambda) = \max_{\theta \in [0,\pi]}\{P(\theta, \lambda)\}$, dove $\theta_{\max}$ è il valore di θ che massimizza la distribuzione $P(\theta, \lambda)$.
2. generare in modo casuale e uniforme due numeri $x \in [0, \pi)$ e $y \in [0, P_{\max}(\lambda))$;
3. se $y < P(x, \lambda)$ accettare $\theta_i = x$, altrimenti tornare al punto precedente.

► **Esercizio in C** Scrivere un programma `NOME_COGNOME.c` che implementi il metodo Monte Carlo descritto per studiare la convergenza dell'energia media $\langle U \rangle$ al variare del numero di estrazioni N_t, per un dato valore della costante λ.
In particolare, il programma dovrà:

1. Definire, tramite opportune direttive `#define`, il numero di punti `NPTS = 5000` utilizzati per determinare il valore massimo della distribuzione $P(\theta, E)$; `NTBEG=128`; `NTB=2`; `NUMNT=13`.
2. Utilizzare il generatore di numeri casuali `drand48` ed inizializzarlo con il valore 42.
3. Chiedere all'utente di inserire il valore della costante λ, assicurandosi che sia nell'intervallo $[0.1, 10]$.
4. Dichiarare un array `Uarr[]` di tipo e dimensioni opportune per memorizzare i valori di $\langle U \rangle$ per $N_t = \texttt{NTBEG} \times \texttt{NTB}^i$ con i intero in $[0, \texttt{NUMNT}]$.
5. Salvare i risultati su un file chiamato `energia.dat`, stampando su due colonne il valore di N_t e il corrispondente valore di $\langle U \rangle$.
6. Stampare a schermo, con 4 cifre decimali, il valore massimo $P_{\max}(\lambda)$ e il valore di θ per cui esso viene assunto.

Suggerimento: conviene calcolare $P_{\max}(\lambda)$ una sola volta dopo richiesto l'inserimento di λ.
Nello scrivere il programma si dovranno definire almeno le seguenti funzioni:

1. Una funzione `ptheta`, che prenda come argomenti θ e λ, calcoli $P(\theta, \lambda)$ e ne restituisca il valore.
2. Una funzione `trova_massimo` che prenda come argomenti λ e due puntatori tramite i quali "restituirà" (passaggio *by reference*) il valore massimo della funzione $P(\theta, \lambda)$ e il corrispondente valore di $\theta_{\max}$.
3. Una funziona `theta_prob` che prenda come argomenti λ e $P_{\max}(\lambda)$ e restituisca un valore di θ generato con probabilità $P(\theta, \lambda)$, come discusso in precedenza.
4. Una funzione `stampa_energia` che prenda come argomento `Uarr[]` che scriva sul file `energia.dat` N_t e i valori corrispondenti di $\langle U \rangle$ calcolati precedentemente. Questi ultimi devono essere stampati con 4 cifre dopo la virgola.

► **Esercizio in Python** Dopo aver verificato il corretto funzionamento del programma in C e averlo eseguito inserendo $\lambda = 5.0$, scrivere uno script `NOME_COGNOME.py` che legga i dati contenuti nel file `energia.dat` e riporti in un grafico il valore di $\langle U \rangle$ in funzione di N_t (in scala logaritmica sull'asse delle ascisse, *Suggerimento:* se avete importato le librerire `matplotlib` con il comando `import matplotlib.pyplot as plt`, potete usare il comando `plt.xscale('log')`). Il grafico deve essere completo di titolo ed etichette degli assi e va salvato nel file `NOME_COGNOME.png`.