

Laboratorio di Calcolo: Prova d'esame del 5 Luglio 2012

Scrivere un programma, chiamato $\langle \text{cognome} \rangle_ \langle \text{nome} \rangle. \text{c}$ (avendo eliminato caratteri speciali da nome e cognome, esempio: Marco D'Alì si scrive **dali_marco.c**), che simuli il moto casuale di 100 particelle. Ogni particella parte dalla posizione $x_0 = 0$. e compie $NPassi$ ($NPassi \in [500, 5000]$) secondo la legge

$$x_{k+1} = x_k - F(x_k) * dt + \sqrt{2. * dt * \gamma} * \xi_{k+1} \quad ; \quad k = 0, \dots, NPassi - 1$$

dove x_k è la posizione al passo k , $dt = 10^{-4}$, $\gamma \in [0.1, 3.]$ e ξ_{k+1} è un numero pseudo-aleatorio con distribuzione Gaussiana a media nulla e varianza pari a uno. Si ha inoltre $F(x_k) = -(1. - x_k^2) * x_k$.

Il programma deve:

- 2 1. Descrivere brevemente cosa fa.
- 3 2. Chiedere in input i valori di $NPassi$ (intero) e γ (double), verificare che ciascuno di essi sia nell'intervallo richiesto e, in caso contrario, stampare un messaggio di errore e chiedere un nuovo valore.

- 3 ciclo giusto 3. Simulare, secondo la legge descritta sopra, il moto di ogni particella mediante un ciclo sul numero di passi. Il ciclo deve contenere

3 (a) una funzione, **Force**, che ad ogni passo restituisce il valore di $F(x_k)$.

2 interfaccia (b) una funzione, **Gauss**, che ad ogni passo calcola ξ_{k+1} . La funzione deve restituire una variabile di tipo double (che è il valore di ξ a quel passo) mediante la seguente procedura:

2 r1

2 r2 a) Generare due variabili reali uniformemente distribuite: $r_1 \in [0, 1]$ e $r_2 \in (0, 1.]$

1 return

1 include math b) Restituire la variabile $\xi = \sin(2\pi r_1) * \sqrt{-2\ln(r_2)}$.

1 dimensione array
1 valore corretto

4. Salvare, nell'array unidimensionale *Posizione*, la posizione finale di ogni particella.
5. Calcolare, in un'apposita funzione **Pos**, il valore minimo e massimo della posizione finale delle particelle e stamparli sullo schermo.

1 interfaccia

2 passaggio array

2 calcolo min

2 cal max

2 printf