Esame di Metodi Computazionali della Fisica I
17 marzo 2010

C++

Nell'esercizio di C^{++} è prevista l'implementazione di una classe *Particella* per simulazioni di particelle interagenti.

Ogni particella è caratterizzata da una massa, una carica e da una posizione in uno spazio di dimensione $3+1$ (spazio + tempo) ed interagirà con le altre particelle con un certo potenziale. L'idea alla base del progetto consisterà nel virtualizzare la classe particella (classe madre) e specializzarla nei vari tipi di interazione (classi figlie).

Nella prima parte dell'esame sarà richiesto l'implementazione di una classe **XYZt** che rappresenterà la posizione della particella nello spazio tempo. Per questa classe sarà chiesto di implementare i principali operatori di somma e prodotto. Nella seconda parte verrà chiesto di progettare una classe **Particella** che rappresenterà una particella con potenziale di interazione generico.

Ogni parte è suddivisa in punti i quali possono essere obbligatori o facoltativi. I punti obbligatori debbono essere svolti in sequenza per il corretto funzionamento della libreria mentre i punti facoltativi (se non esplicitamente indicato) possono essere svolti in qualsiasi ordine. Il codice deve essere **correttamente compilabile** e accompagnato dal rispettivo **Makefile** funzionante (se il Makefile è assente saranno decurtati dei punti). Non saranno accettati codici non compilabili (abbiate cura di commentare la parte di codice che non riuscite sviluppare. In caso di indecisione sul voto tale parte sarà comunque parzialmente valutata). Il codice deve rispondere correttamente alle richieste previste nel main. In sede di valutazione il codice sarà comunque testato per verificarne l'affidabilità.

I punti obbligatori sono:

I Parte (a.1), (a.2), (a.3), (b.1), (b.2)

II Parte (c.1), (c.2), (c.3), (c.4), (c.5), (c.6), (d.1), (d.2), (d.3)

1 I Parte: classe XYZt

La classe **XYZt** rappresenta un vettore nello spazio-tempo. In quanto tale, dovrà memorizzare la posizione nello spazio tridimensionale ed il tempo (si consiglia di usare un **double pos[3]** per la posizione spaziale e **double time** per il tempo). Il numero di dimensioni è fisso a $3+1$. Le variabili devono essere dichiarate **private**.

(a) Costruttori/Distruttori:

(1 - Obb) Implementare il costruttore di default (che setta la posizione spazio-temporale a zero) ed il costruttore che accetta in ingresso quattro **double** (3 per lo spazio ed uno per il tempo).

(2 - Obb) Implementare il costruttore di copia.

(3 - Obb) Implementare il distruttore di default.

(4) Implementare il costruttore che accetta in ingresso un puntatore ad un double ed un double (rispettivamente per la posizione e per il tempo).

(b) Operatori:

(1 - Obb) Implementare gli operatori **X()**, **Y()**, **Z()**, **T()** che restituiscono rispettivamente la componente x, y, z e t del vettore spazio-tempo.

(2 - Obb) Implementare l'operatore di assegnamento tra due oggetti **XYZt**.

(3) Implementare l'operatore $*$ che calcola il prodotto scalare tra le componenti spaziali di due vettori **XYZt** se le loro componenti temporali coincidono. In caso contrario restituisce 0.

(4) Implementare l'operatore [] che restituisce il valore delle componenti spaziali (se la posizione richiesta è compresa tra 1 e 3) e temporale (se la posizione richiesta è 4). Altrimenti zero.

(5) Implementare gli operatori $+=$ e $-=$ tra oggetti **XYZt** che rispettivamente agiscono sommando e sottraendo le componenti del vettore spazio-tempo.

(6) Implementare gli operatori $+$ e $-$ tra oggetti **XYZt** che rispettivamente agiscono sommando e sottraendo le componenti del vettore spazio-tempo.

(7) Implementare l'operatore $-x$ (dove x è un oggetto di tipo **XYZt**) che cambia il segno *esclusivamente* alla parte spaziale.

2 II Parte: classe Particella

La classe **Particella** rappresenterà una particella con un'interazione generica. Successivamente, questa classe potrà essere specializzata (attraverso il meccanismo dell'ereditarietà) aggiungendo un'interazione più precisa. Una generica particella è caratterizzata da una posizione, una velocità, un'accelerazione (per semplicità assumiamo che tutte e tre le grandezze siano rappresentate da un oggetto di tipo **XYZt**), una massa (**double**) e una carica (**double**). Le variabili devono essere dichiarate **private**.

(c) Classe Particella (Madre):

(1 - Obb) Implementare il costruttore che accetta in ingresso la massa e la carica, e setta al valore di default la posizione, la velocità e l'accelerazione.

(2 - Obb) Implementare il costruttore che accetta in ingresso tutti i parametri.

(3 - Obb) Implementare gli operatori **M()** e **E()** che restituiscono rispettivamente la massa, la carica ed il tipo della particella.

(4 - Obb) Implementare gli operatori **Pos()**, **Vel()**, **Acc()** che restituiscono rispettivamente la posizione, la velocità e l'accelerazione.

(5 - Obb) Dichiarare una funzione virtual pura **_Interazione()** (nella sezione **private**) che accetta in ingresso un oggetto di tipo **Particella** e restituisce un **double** che rappresenta il potenziale di interazione tra le particelle. Questa sarà l'unica funzione che dovrà essere specializzata per ereditarietà in quanto è l'unica differenza che caratterizza particelle di diverso tipo.

(6 - Obb) Implementare l'operatore $*$ tra due oggetti di tipo **Particella** che restituisce il potenziale di interazione un oggetto di tipo **XYZt**) che settano rispettivamente la posizione, la velocità e l'accelerazione. (sfruttando la funzione definita al punto 5).

(7) Implementare gli operatori **SetPos()**, **SetVel()**, **setAcc()** (che chiedono come parametro

(d) Classe Elettrone (Figlia):

Le classe **Elettrone** deve ereditare **public** dalla classe **Particella**. Due elettroni interagiscono con un potenziale del tipo $V = \frac{q_1 \cdot q_2}{|\vec{r_1}(t) - \vec{r_2}(t)|}$. Gli unici parametri liberi sono la posizione, la velocità e l'accelerazione.

(1 - Obb) Implementare il costruttore di default (che setta al valore di default la posizione, la velocità e l'accelerazione) e assegna la massa e la carica dell'elettrone (rispettivamente $m = 1$ ed $e = -1$).

(2 - Obb) Implementare il costruttore che accetta in ingresso la posizione, la velocità e l'accelerazione.

(3 - Obb) Implementare la funzione **_Interazione()** (dichiarata nella classe madre virtual) come indicato (se i tempi non coincidono l'interazione è nulla ..).

3 III Parte: Main

(e) Main:

(1) Creare un file main che crei due elettroni.

(2) Stampare a video il potenziale di interazione.

Mathematica

Si consideri l' equazione differenziale

$$\frac{d^2\theta(t)}{dt^2} + g\frac{d\theta(t)}{dt} + \omega^2\theta(t) \;=\; \frac{f_0}{m}\cos(\omega_f t) \tag{1}$$

- Si ponga $f_0 = 0$ e si discuta la soluzione nei 3 casi: 1) $g = 3\omega$, 2) $g = 2\omega$, 3) $g = \omega/4$.
  (Si utilizzi il comando `Limit` *nel secondo caso.)*
  Posto $\omega = 1$, si disegnino in un solo grafico, per $t \in [0, 20]$, le soluzioni corrispondenti a $g = 1/4, 2, 3$, con condizioni al contorno $\theta'(0) = 0,\ \theta(0) = 1$.
  Si visualizzi l'intero intervallo in y delle soluzioni.

- Si disegni in un grafico 3D la soluzione, fissati $g = 1, f_0 = 1, m = 1, \omega = 1$, con condizioni al contorno $\theta'(0) = 0,\ \theta(0) = 1$, ponendo sugli assi x e y rispettivamente $t \in [0, 20]$ e $\omega_f \in [0, 3]$.

- Si confrontino nello stesso grafico, con $t \in [0, 30]$, le seguenti tre curve 2D:
  a) $f_0 = 0, g = 0, \omega = 1$;
  b) $f_0 = 1, m = 1, g = 1, \omega_f = \omega = 1$;
  c) $f_0 = 1, m = 1, g = 1, \omega_f = 2, \omega = 1$.
  Si commenti la differenza tra il caso a) e i casi b) e c).
  Si commenti il caso b) in relazione al plot 3D del punto precedente.

- Si fattorizzi il seguente polinomio:

$$\begin{aligned} p_{100}(x) \;=\;& x + x^2 + x^3 + x^4 + x^5 + x^6 + x^7 + x^8 + x^9 + x^{10} + \\ & + x^{11} + x^{12} + x^{13} + x^{14} + x^{15} + x^{16} + x^{17} + x^{18} + x^{19} + x^{20} + \\ & + x^{21} + x^{22} + x^{23} + x^{24} + x^{25} + x^{26} + x^{27} + x^{28} + x^{29} + x^{30} + \\ & + x^{31} + x^{32} + x^{33} + x^{34} + x^{35} + x^{36} + x^{37} + x^{38} + x^{39} + x^{40} + \\ & + x^{41} + x^{42} + x^{43} + x^{44} + x^{45} + x^{46} + x^{47} + x^{48} + x^{49} + x^{50} + \\ & + x^{51} + x^{52} + x^{53} + x^{54} + x^{55} + x^{56} + x^{57} + x^{58} + x^{59} + x^{60} + \\ & + x^{61} + x^{62} + x^{63} + x^{64} + x^{65} + x^{66} + x^{67} + x^{68} + x^{69} + x^{70} + \\ & + x^{71} + x^{72} + x^{73} + x^{74} + x^{75} + x^{76} + x^{77} + x^{78} + x^{79} + x^{80} + \\ & + x^{81} + x^{82} + x^{83} + x^{84} + x^{85} + x^{86} + x^{87} + x^{88} + x^{89} + x^{90} + \\ & + x^{91} + x^{92} + x^{93} + x^{94} + x^{95} + x^{96} + x^{97} + x^{98} + x^{99} + x^{100} \end{aligned} \tag{2}$$