Prova d'esame di Laboratorio di Calcolo I
per il corso di laurea in Matematica
4 Febbraio 2010

Tema d'esame: studio di alcune proprietà dei polinomi di Legendre.

Descrizione del metodo di calcolo

I polinomi di Legendre P_n sono le funzioni che risolvono la seguente equazione differenziale (anch'essa detta di Legendre):

$$\frac{\mathrm{d}}{\mathrm{d}x}\left[\left(1-x^2\right)\frac{\mathrm{d}}{\mathrm{d}x}P_n(x)\right]+n(n+1)P_n(x)=0\ , \tag{1}$$

$\forall\ n\ \geq\ 0$. I polinomi di Legendre possono essere definiti (e, quindi, *calcolati*) in modo ricorsivo, come segue. Siano

$$P_0(x)=1\ ,\qquad P_1(x)=x \tag{2}$$

e, $\forall\ n\geq 2$,

$$nP_n(x)=(2n-1)xP_{n-1}(x)-(n-1)P_{n-2}(x)\ . \tag{3}$$

Dalle definizioni riportate nelle formule (2)–(3), si dimostra facilmente (per induzione) che $P_n(x)$ è un polinomio di grado n e con la stessa parità di n stesso, $\forall\ n\geq 0$. A titolo di esempio, riportiamo qui di seguito l'espressione di $P_n(x)$ per $2\leq n\leq 8$:

$$\begin{aligned}
P_2(x)&=\frac{1}{2}\left(3x^2-1\right)\ ,\\
P_3(x)&=\frac{1}{2}\left(5x^3-3x\right)\ ,\\
P_4(x)&=\frac{1}{8}\left(35x^4-30x^2+3\right)\ ,\\
P_5(x)&=\frac{1}{8}\left(63x^5-70x^3+15x\right)\ ,\\
P_6(x)&=\frac{1}{16}\left(231x^6-315x^4+105x^2-5\right)\ ,\\
P_7(x)&=\frac{1}{16}\left(429x^7-693x^5+315x^3-35x\right)\ ,\\
P_8(x)&=\frac{1}{128}\left(6435x^8-12012x^6+6930x^4-1260x^2+35\right)\ .
\end{aligned} \tag{4}$$

I polinomi di Legendre godono di varie proprietà, la più importante delle quali è, probabilmente, quella di costituire una base ortogonale nello spazio funzionale $L^2\big([-1,1]\big)$. In particolare, i polinomi di Legendre compaiono nella seguente ben nota espansione in serie di Taylor:

$$\frac{1}{\sqrt{1-2xt+t^2}}=\sum_{n=0}^{+\infty}P_n(x)t^n\ , \tag{5}$$

che sussiste per opportuni insiemi di valori di x e t. La formula precedente è molto usata in vari campi della fisica, perché essa consente di scrivere (senza avere funzioni a denominatore) l'espressione alla sinistra del simbolo "=", la quale compare nell'energia potenziale di tipo gravitazionale o elettrostatica.

Rappresentazione dei polinomi di Legendre
Quando si effettua il calcolo dei polinomi di Legendre tramite un programma in linguaggio **C**, bisogna per prima cosa adottare una buona tecnica per *rappresentare* in modo conveniente questi stessi polinomi sul computer. Ai fini della soluzione degli esercizi qui proposti, la seguente strategia è adeguata, senza essere troppo complicata.

Nel programma in linguaggio **C**, si identifichi ciascun polinomio di Legendre P_n con un vettore di tipo `double` di dimensione $N+1$; i primi $n+1$ elementi di tale vettore ospiteranno i valori $a_0 \ \ldots$, a_n dei coefficienti del polinomio $P_n(x) = a_0 + a_1 x + \ldots a_n x^n$. Ovviamente, tale rappresentazione è valida solo se $n \leq N$; al fine di garantire che questa condizione sia sempre verificata durante l'esecuzione del programma, dovranno essere presenti gli opportuni controlli laddove verrà inserito il valore di n.

È sicuramente opportuno stabilire il valore del parametro N per mezzo di una direttiva `#define`. Tale valore dovrà essere abbastanza grande, senza però essere enorme al punto di rallentare eccessivamente l'esecuzione del programma; per fissare le idee, $N = 200$ dovrebbe essere una scelta opportuna per garantire il buon funzionamento del programma in ognuna delle parti che verranno descritte in seguito.

Obiettivo (intermedio) 1:
si scriva un programma in linguaggio **C** che verifica *numericamente* l'equazione (3) *per una fissata coppia di valori* di n e x. Il programma deve contenere:

(A) una *function* che ha come argomenti un vettore $\underline{a}$ e un intero n, che contengono, rispettivamente, *(alla fine della chiamata)* i valori dei coefficienti $a_0 \ \ldots$, a_n di un polinomio $P(x) = a_0 + a_1 x + \ldots a_n x^n$ e *(già all'inizio della chiamata)* il grado n di quello stesso polinomio; tale *function* deve permettere di effettuare l'*input da tastiera* dei coefficienti del polinomio;

(B) una *function* che ha tre argomenti: un vettore $\underline{a}$, un intero n e un numero reale $\bar{x}$, tali che $\underline{a}$ e n sono, rispettivamente, i coefficienti e il grado di un polinomio $P(x) = a_0 + a_1 x + \ldots a_n x^n$; tale *function (alla fine della chiamata)* deve restituire il valore di $P(\bar{x})$;

(C) l'*input da tastiera* dei valori di n e $x \in \mathbf{R}$; il valore intero di n deve essere sottoposto a un test in modo tale che, se esso *non è compreso* nell'intervallo $[2, N]$, deve essere *reinserito correttamente*;

(D) *tre chiamate della function descritta al punto (A)* con scelte opportune degli argomenti, in modo tale da effettuare l'*input da tastiera* dei coefficienti dei polinomi P_n, P_{n-1} e P_{n-2}, dove il valore di n è quello stabilito al punto (C);

(E) il calcolo della quantità $\sigma = (2n-1)xP_{n-1}(x) - (n-1)P_{n-2}(x) - nP_n(x)$; tale calcolo deve essere svolto effettuando opportunamente *tre chiamate della function descritta al punto (B)*;

(F) la stampa sul video *in formato esponenziale* del valore della quantità σ definita al punto (E); ovviamente, la verifica numerica dell'equazione (3) *è da considerarsi ben* *riuscita* se il valore assoluto di σ è al più dell'ordine di grandezza dell'*errore di macchina.*

Obiettivo (intermedio) 2:
si integri il programma richiesto dall'obiettivo 1, aggiungendo una verifica *numerica* della seguente equazione:

$$P_0(x) + P_1(x)t + P_2(x)t^2 - \frac{1}{\sqrt{1-2xt+t^2}} = \mathcal{O}(t^3) \ , \tag{6}$$

che segue dalla formula (5). A tal fine si proceda come segue:

(A) si sottopongano a un test preliminare i valori di n e x; qualora $n \neq 2$ oppure $|x| > 1$ si stampi sul video un messaggio (che afferma che non è possibile eseguire la verifica richiesta) e *non si eseguano* le istruzioni descritte ai seguenti punti (B)–(C3);

(B) si iterino le istruzioni descritte ai seguenti punti (C1)–(C3) per j (che è un *contatore* intero) che va da 0 fino a 5;

- (C1) si ponga $t = 0.9/10^j$;
- (C2) si calcoli il valore del primo membro dell'equazione (6), cioè $\delta = P_0(x) + P_1(x)t + P_2(x)t^2 - \frac{1}{\sqrt{1-2xt+t^2}}$; tale calcolo dovrà essere effettuato utilizzando ripetutamente la *function* descritta al punto (B) dell'obiettivo 1;
- (C3) si stampino ordinatamente sul video i valori *in formato esponenziale* di t e $|\delta|$.

Un consiglio
La tabella seguente rappresenta un esempio di come può essere impostato l'output richiesto al punto (C3). I valori numerici riportati sono quelli corrispondenti al caso $x = -0.7$; si osservi che la tabella seguente mostra molto chiaramente che $|\delta| = \mathcal{O}(t^3)$ *(vero?!)*.

t	|delta|
9.000000e-01	1.038015e − 02
9.000000e-02	1.153810e − 04
9.000000e-03	1.376505e − 07
9.000000e-04	1.400624e − 10
9.000000e-05	1.403164e − 13
9.000000e-06	1.592151e − 16

Obiettivo (intermedio) 3:
si integri il programma richiesto dall'obiettivo 2, in modo tale da effettuare, $\forall\ x \in \mathbf{R}$, la verifica numerica dell'equazione differenziale (1). A tal fine si proceda come segue:

(A) si scriva una *function* che ha tre argomenti: un vettore $\underline{a}$, un intero n e un secondo vettore $\underline{b}$, tali che $\underline{a}$ e n sono, rispettivamente, i coefficienti e il grado di un polinomio $P(x) = a_0 + a_1x + \ldots a_nx^n$; *(alla fine della chiamata)* nel vettore $\underline{b}$ dovranno essere riportati i coefficienti del polinomio $Q(x) = P(x)$ (proprio così, $Q(x)$ altro non è che una copia di $P(x)$) cioè $P(x) = Q(x) = b_0 + b_1x + \ldots b_nx^n$;

(B) si scriva una *function* che ha tre argomenti: un vettore $\underline{a}$, un intero n e un numero reale c, tali che $\underline{a}$ e n sono, rispettivamente, i coefficienti e il grado di un polinomio

$P(x) = a_0 + a_1x + \ldots a_nx^n$; *(alla fine della chiamata)* nel vettore $\underline{a}$ dovranno essere riportati i coefficienti del nuovo polinomio $cP(x)$, cioè $cP(x) = a_0 + a_1x + \ldots a_nx^n$;

(C) si scriva una *function* che ha due argomenti: un vettore $\underline{a}$ e un intero n, tali che essi sono, rispettivamente, i coefficienti e il grado di un polinomio $P(x) = a_0 + a_1x + \ldots a_nx^n$; *(alla fine della chiamata)* nel vettore $\underline{a}$ dovranno essere riportati i coefficienti del prodotto polinomiale $xP(x)$, cioè $xP(x) = a_0 + a_1x + \ldots a_{n+1}x^{n+1}$;

(D) si scriva una *function* che ha tre argomenti: un vettore $\underline{a}$, un intero n e un secondo vettore $\underline{b}$, tali che $\underline{a}$ e n sono, rispettivamente, i coefficienti e il grado di un polinomio $P(x) = a_0 + a_1x + \ldots a_nx^n$; *(alla fine della chiamata)* nel vettore $\underline{b}$ dovranno essere riportati i coefficienti della derivata di $P(x)$, cioè $P'(x) = b_0 + b_1x + \ldots b_{n-1}x^{n-1}$;

(E) si scriva una *function* che ha cinque argomenti: un vettore $\underline{a}$, un intero n, un secondo vettore $\underline{b}$, un altro intero m e un terzo vettore $\underline{c}$; tali che $\underline{a}$ e n sono, rispettivamente, i coefficienti e il grado di un polinomio $P(x) = a_0 + a_1x + \ldots a_nx^n$, mentre $\underline{b}$ e m sono, rispettivamente, i coefficienti e il grado di un polinomio $Q(x) = b_0 + b_1x + \ldots b_mx^m$; a tale *function* è richiesto di funzionare correttamente solo se $n \geq m$, altrimenti l'esecuzione dell'intero programma può essere arrestata, dopo aver stampato su video un messaggio di errore; *(alla fine della chiamata)* nel vettore $\underline{c}$ dovranno essere riportati i coefficienti della somma polinomiale $P(x) + Q(x)$, cioè $P(x) + Q(x) = c_0 + c_1x + \ldots c_nx^n$;

(F) si scriva una *function* che ha due argomenti: un vettore $\underline{a}$ e un intero n, tali che essi sono, rispettivamente, i coefficienti e il grado di un polinomio $P(x) = a_0 + a_1x + \ldots a_nx^n$; tale *function* non deve fare altro che stampare ordinatamente sul video i coefficienti del polinomio $P(x)$;

(G) all'interno della *main function* si effettuino le opportune *chiamate delle function descritte ai punti (A)–(E)*, in modo tale da calcolare il membro di destra dell'equazione (1), cioè $R(x) = \frac{\mathrm{d}}{\mathrm{d}x}\left[(1-x^2)\frac{\mathrm{d}}{\mathrm{d}x}P_n(x)\right] + n(n+1)P_n(x)$; infine, si utilizzi la *function descritta al punto (F)* per visualizzare il polinomio risultato $R(x)$. Ovviamente, la verifica numerica dell'equazione differenziale (1) *è da considerarsi ben riuscita* se il valore assoluto di ciascun coefficiente di $R(x)$ è al più dell'ordine di grandezza dell'*errore di macchina.*

Obiettivo (intermedio) 4:
si integri il programma richiesto dall'obiettivo 3, in modo tale da effettuare il calcolo *iterativo* dei polinomi di Legendre utilizzando le formule (2)–(3). A tal fine, si proceda come segue:

(A) all'interno della *main function*, si ripeta una seconda volta l'*input da tastiera* del valore di n che deve essere sottoposto a un test in modo tale che, se esso *non è compreso* nell'intervallo $[2, N]$, deve essere *reinserito correttamente*;

(B) si scriva una *function* che ha due argomenti: un intero n e un vettore $\underline{a}$; tale *function* effettua il calcolo del polinomio di Legendre P_n, quindi *(alla fine della chiamata)* nel vettore $\underline{a}$ dovranno essere riportati i coefficienti di $P_n(x)$, cioè $P_n(x) = a_0 + a_1x + \ldots a_nx^n$; questa *function* può essere realizzata in modo da tradurre in linguaggio di programmazione la procedura descritta ai seguenti punti (c)–(f);

(c) all'interno della *function* deve essere allocata la memoria per almeno tre vettori che rappresentano i polinomi $Q(x)$, $R(x)$ e $S(x)$ (inoltre, saranno necessari alcuni polinomi che ospiteranno i risultati di qualche operazione intermedia);

(d) si definiscano opportunamente i valori dei coefficienti dei vettori, rispettivamente, corrispondenti a $Q(x)$ e $R(x)$ in modo tale che $Q(x) = 1$ e $R(x) = x$ (cioè $Q(x)$ e $R(x)$ vengono *inizialmente posti uguali* a $P_0(x)$ e $P_1(x)$);

(e) si iterino le istruzioni descritte ai seguenti punti (e1)–(e3) in modo tale da costituire un ciclo, che viene eseguito per i (che è un *contatore* intero) che va da 2 fino a n ;

  (e1) utilizzando ripetutamente *le function descritte ai punti (A)–(E) dell'obiettivo 3*, si effettui il calcolo del polinomio $S(x) = [(2i-1)xR(x) - (i-1)Q(x)]/i$, cioè $R(x)$ viene posto uguale a $P_i(x)$;

  (e2) con un'opportuna *chiamata della function relativa al punto (A) dell'obiettivo 3*, si ridefinisca $Q(x)$ in modo tale che $Q(x) = R(x)$, cioè $Q(x)$ viene ora posto uguale a $P_{i-1}(x)$;

  (e3) con un'opportuna *chiamata della function relativa al punto (A) dell'obiettivo 3*, si ridefinisca $R(x)$ in modo tale che $R(x) = S(x)$, cioè $R(x)$ viene ora posto uguale a $P_i(x)$;

(f) si effettuino alcuni test alla fine della *function*, i quali devono essere strutturati come segue: se $n = 0$, allora si ridefinisca $S(x)$ in modo tale che $S(x) = Q(x)$, cioè $S(x)$ viene ora posto uguale a $P_0(x)$; altrimenti, se $n = 1$, allora si ridefinisca $S(x)$ in modo tale che $S(x) = R(x)$, cioè $S(x)$ viene ora posto uguale a $P_1(x)$; si osservi che, dopo l'esecuzione dei suddetti test e del ciclo descritto ai punti (e)–(e3), $S(x)$ è sicuramente uguale al polinomio di Legendre richiesto, cioè $P_n(x)$;

(G) all'interno della *main function* si effettui la *chiamata della function descritta al punto (F) dell'obiettivo 3* per visualizzare l'espansione calcolata del polinomio $P_n(x)$. Ovviamente, si consiglia di confrontare i risultati ottenuti con qualcuna delle espressioni di $P_2(x)$, ... , $P_8(x)$, riportate in formula (4).

Obiettivo (finale) 5:

al programma richiesto dagli obiettivi precedenti si aggiunga anche la verifica *numerica* dell'equazione (5). A tal fine, si proceda come segue:

(A) all'interno della *main function*, si effettui l'*input da tastiera* altre due volte, in modo da definire, prima, un nuovo valore di x che deve essere sottoposto a un test in modo tale che, se esso *non è compreso* nell'intervallo $[-1, 1]$, deve essere *reinserito correttamente*; inoltre, si definisca, poi, un valore di t che deve essere sottoposto a un test in modo tale che, se esso *non è compreso* nell'intervallo $[0, 1/2]$, deve essere *reinserito correttamente*;

(B) si effettui il calcolo della quantità $\varepsilon = \sum_{n=0}^{N} P_n(x)t^n - \frac{1}{\sqrt{1-2xt+t^2}}$; tale calcolo deve essere svolto effettuando opportunamente la *chiamata della function descritta al precedente obiettivo 4*;

(C) si stampi ordinatamente sul video il valore *in formato esponenziale* di $|\varepsilon|$. Ovviamente, la verifica numerica dell'equazione (5) *è da considerarsi ben riuscita* [$\star$] se il valore assoluto di ε è al più dell'ordine di grandezza dell'*errore di macchina, quando N è abbastanza grande.*

[$\star$] Si osservi che al punto (A) dell'obiettivo 5 imponiamo che $t \in [0, 1/2]$ (quando $x \in [-1, 1]$). In verità, questa condizione è stata posta solo per far sì che la convergenza delle somme parziali al risultato sia abbastanza veloce, anche se non è la più ampia possibile.