Calcolo Numerico — Informatica

Manolo Venturin A.A. 2010–2011

Guida all'esame

Testo aggiornato al **23 maggio 2011**.

L'esame consiste in una prova scritta della durata di **2 ore**. Tale prova è composta da **tre/quattro domande**.

A partire da pag. 2 sono riportate delle domande tipo con il relativo tempo di risposta suddiviso nel tempo da dedicare alla teoria e nel tempo da dedicare all'implementazione MatLab/Octave che viene fissato a 15 minuti per tutte le domande. Nel combinare le domande finali, la parte di implementazione di laboratorio non viene sempre chiesta (tipicamente solo su un esercizio è presente).

Le possibili combinazioni che portano ad un totale 120 minuti sono:

1. 60 min + 30 min + 30 min;
2. 45 min + 45 min + 30 min;
3. 30 min + 30 min + 30 min + 30 min.

Nell'ultimo caso ci sarà una domanda di implementazione di codice MatLab/Octave che verrà valutata 30 min invece di 15 min; questo per tener conto della presenza di quattro domande invece di tre.

Inoltre, le domande sono scelte su argomenti diversi, in particolare:

- Analisi degli errori e stabilità numerica (dalla Domanda # 1 alla Domanda # 7);
- Ricerca di zeri di funzioni non-lineari (dalla Domanda # 8 alla Domanda # 11);
- Risoluzione di sistemi lineari con metodi diretti o iterativi (dalla Domanda # 12 alla Domanda # 16);
- Fattorizzazione QR e SVD e minimi quadrati (dalla Domanda # 17 alla Domanda # 18);
- Interpolazione polinomiale (dalla Domanda # 19 alla Domanda # 20).

Se la parte di **implementazione** non risultasse sufficiente la prova non è considerata superata.

Domanda # 1 [30 minuti (teoria) + 15 minuti (Laboratorio)]

Studiare dal punto di vista numerico la valutazione della derivata di una funzione calcolata mediante rapporto incrementale al tendere a zero dell'incremento, cioè $f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$, illustrando l'andamento tipico dell'errore commesso (errori di approssimazione e di cancellazione numerica).

Scrivere un programma MatLab/Octave che permetta di evidenziare questo fenomeno nel caso della funzione $f(x) = x^3 + 1$.

Domanda # 2 [30 minuti (teoria) + 15 minuti (Laboratorio)]

Studiare dal punto di vista numerico la valutazione della seguente espressione $\frac{(1+\eta)^{-1} - 1}{\eta} \equiv 1$ evidenziando il fenomeno della cancellazione numerica.

Scrivere un programma MatLab/Octave che permetta di mostrare questo fenomeno.

Domanda # 3 [30 minuti (teoria) + 15 minuti (Laboratorio)]

Studiare dal punto di vista numerico la valutazione delle seguenti due espressioni: (i) $P_1(x) = (x - 1)^7$ e (ii) $P_2(x) = x^7 - 7x^6 + \dots - 1$ evidenziando, dove presente, il fenomeno della cancellazione numerica.

Scrivere un programma MatLab/Octave che permetta di mostrare questo fenomeno.

Domanda # 4 [30 minuti (teoria) + 15 minuti (Laboratorio)]

Studiare dal punto di vista numerico la valutazione delle seguenti due espressioni: (i) $P_1(x) = (x - 1)^7$ e (ii) $P_2(x) = x^7 - 7x^6 + \dots - 1$ evidenziando, dove presente, il fenomeno della cancellazione numerica.

Scrivere un programma MatLab/Octave che permetta di mostrare questo fenomeno.

Domanda # 5 [30 minuti (teoria) + 15 minuti (Laboratorio)]

Studiare dal punto di vista numerico la valutazione delle radici di un'equazione di II grado, cioè $ax^2 + bx + c = 0$, evidenziando, dove presente, il fenomeno della cancellazione numerica.

Scrivere un programma MatLab/Octave che permetta di mostrare questo fenomeno.

Domanda # 6 [30 minuti (teoria) + 15 minuti (Laboratorio)]

Per calcolare il seguente integrale $I_n = e^{-1} \int_0^1 x^n e^x dx$ si possono utilizzare le seguenti due formulazioni equivalenti (dall'integrazione per parti dell'integrale): (i) $I_{n+1} = 1 - (n + 1) I_n$ con $I_1 = e^{-1}$; (ii) $I_n = \frac{1 - I_{n+1}}{n+1}$ con $I_\infty = 0$. Analizzare dal punto di vista numerico queste due successioni.

Scrivere un programma MatLab/Octave che permetta di valutare le due successioni.

Domanda # 7 [30 minuti (teoria) + 15 minuti (Laboratorio)]

Per calcolare il valore di pi-greco si può utilizzare la seguente successione

$$\begin{cases} z_1 = 1, & z_2 = 2 \\ z_{n+1} = 2^{n-\frac{1}{2}} \sqrt{1 - \sqrt{1 - 4^{1-n} \cdot z_n^2}} \end{cases}$$

Analizzare tale successione dal punto di vista numerico e proporre una successione numericamente stabile.

Scrivere un programma MatLab/Octave che permetta di valutare le due successioni.

Domanda # 8 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare il metodo di Bisezione riportando la convergenza del metodo e i relativi test di arresto. Fornire, a scelta, un esempio di applicazione riportando le prime due iterate del metodo. Scrivere in linguaggio MatLab/Octave il ciclo principale di convergenza del metodo, riportare inoltre la dichiarazione della funzione MatLab/Octave con i relativi parametri di ingresso ed uscita.

Domanda # 9 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare il metodo di Newton o delle tangenti riportando la convergenza del metodo e i relativi test di arresto. Fornire, a scelta, un esempio di applicazione riportando le prime due iterate del metodo. Scrivere in linguaggio MatLab/Octave il ciclo principale di convergenza del metodo, riportare inoltre la dichiarazione della funzione MatLab/Octave con i relativi parametri di ingresso ed uscita.

Domanda # 10 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare il metodo delle secanti riportando la convergenza del metodo e i relativi test di arresto. Fornire, a scelta, un esempio di applicazione riportando le prime due iterate del metodo. Scrivere in linguaggio MatLab/Octave il ciclo principale di convergenza del metodo, riportare inoltre la dichiarazione della funzione MatLab/Octave con i relativi parametri di ingresso ed uscita.

Domanda # 11 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare il metodo di punto fisso riportando la convergenza del metodo e i relativi test di arresto. Fornire, a scelta, un esempio di applicazione riportando le prime due iterate del metodo. Scrivere in linguaggio MatLab/Octave il ciclo principale di convergenza del metodo, riportare inoltre la dichiarazione della funzione MatLab/Octave con i relativi parametri di ingresso ed uscita.

Domanda # 12 [30 minuti (teoria)]

Illustrare il condizionamento di un sistema lineare e sue implicazioni nella risoluzione dei sistemi lineari. Fornire un esempio, a scelta, di calcolo del numero di condizionamento per una matrice di dimensioni 2×2 utilizzando la norma infinito.

Domanda # 13 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare i metodi di eliminazione gaussiana per la risoluzione di sistemi lineari motivando la necessità dell'introduzione del pivoting. Illustrare inoltre la fattorizzazione LU di una matrice e risoluzione dei sistemi lineari che si ottengono.

Scrivere in linguaggio MatLab/Octave una funzione per la risoluzione dei sistemi lineari nel caso di una matrice triangolare superiore.

Domanda # 14 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare il metodo iterativo di Jacobi per la risoluzione dei sistemi lineari.

Scrivere in linguaggio MatLab/Octave una funzione per la risoluzione dei sistemi lineari mediante il metodo di Jacobi.

Domanda # 15 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare il metodo iterativo di Gauss-Seidel per la risoluzione dei sistemi lineari.

Scrivere in linguaggio MatLab/Octave una funzione per la risoluzione dei sistemi lineari mediante il metodo di Gauss-Seidel.

Domanda # 16 [30 minuti (teoria)]

Pregi e difetti dei metodi diretti ed iterativi per la risoluzione dei sistemi lineari.

Domanda # 17 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare l'algoritmo che realizza la fattorizzazione QR di una matrice mediante le trasformazioni di Householder.

Scrivere il codice MatLab dell'algoritmo che realizza la fattorizzazione QR appena descritta.

Domanda # 18 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare l'approssimazione polinomiale (minimi quadrati) e la relativa risoluzione mediante: (i) fattorizzazione di Cholesky, (ii) fattorizzazione QR e (iii) fattorizzazione SVD.

Scrivere il codice MatLab che calcola tale minimi utilizzando la fattorizzazione QR e SVD.

Domanda # 19 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare l'interpolazione polinomiale e il fenomeno di Runge.

Istruzione MatLab/Octave che realizzano l'interpolazione polinomiale.

Domanda # 20 [45 minuti (teoria) + 15 minuti (Laboratorio)]

Illustrare il fenomeno di Runge e l'interpolazione polinomiale a tratti.

Istruzione MatLab/Octave che realizzano l'interpolazione polinomiale a tratti (spline lineare e cubiche).