

Algoritmi da codificare per la prova di laboratorio di Analisi Numerica in linguaggio Matlab o Octave (A.A. 2013-2014, M. Annunziato)

In tutti gli algoritmi prevedere un contatore per contare il numero di operazioni macchina che potrà essere dato in uscita. I requisiti qui richiesti sono minimi, lo studente può aggiungere anche altri dati in ingresso o uscita. Documentare opportunamente tutti i codici. Di propria iniziativa si possono creare algoritmi relativi ad altri argomenti del corso.

[F= facoltativo]

1. Determinazione della precisione macchina
2. Determinazione del più piccolo numero rappresentabile
3. Algoritmo di sostituzione all'indietro. (dati: Matrice triangolare superiore, vettore termini noti. uscita: soluzione del sistema di equazione triangolare)
4. Algoritmo di sostituzione in avanti. (dati: Matrice triangolare inferiore, vettore termini noti. uscita: soluzione del sistema di equazione triangolare) [F]
5. Metodo di eliminazione di Gauss senza pivoting. (dati: Matrice quadrata, vettore termini noti. uscita: matrice triangolare superiore trasformata dal metodo di eliminazione, vettore termini noti trasformato).
6. Soluzione sistema di equazione con eliminazione di Gauss, pivoting parziale, e fattorizzazione LU. (dati Matrice quadrata, vettore termini noti. uscita: vettore soluzione del sistema, matrici L, U, P della fattorizzazione $PA=LU$).
7. Come 6. con il controllo della quasi-singularità della matrice dei coefficienti [F]
8. Algoritmo di Thomas. (dati Matrice tri-diagonale, vettore termini noti. uscita: vettore soluzione del sistema di equazione). [F]
9. Metodo di Jacobi con formula di iterazione esplicita (senza l'uso della matrice di iterazione). (dati: Matrice quadrata, vettore termini noti, accuratezza richiesta, numero massimo di iterazioni. uscita: soluzione del sistema di equazione, stima dell'errore, numero effettivo di iterazioni).
10. Metodo di Gauss-Seidel con formula di iterazione esplicita (senza l'uso della matrice di iterazione). (dati: Matrice quadrata, vettore termini noti, accuratezza richiesta, numero massimo di iterazioni. uscita: soluzione del sistema di equazione, stima dell'errore, numero effettivo di iterazioni).
11. Metodo SOR con formula di iterazione esplicita (senza l'uso della matrice di iterazione). (dati: Matrice quadrata, vettore termini noti, parametro di accelerazione, accuratezza richiesta, numero massimo di iterazioni. uscita: soluzione del sistema di equazione, stima dell'errore, numero effettivo di iterazioni). [F]
12. Metodo delle potenze base (dati: Matrice quadrata, vettore iniziale, numero massimo di iterazioni, accuratezza richiesta. uscita: autovalore di modulo max, stima errore, numero di iterazioni).
13. Metodo delle potenze inverse (dati: Matrice quadrata, approssimazione iniziale di un autovalore, vettore iniziale, numero massimo di iterazioni, accuratezza richiesta. uscita: autovalore, stima errore, numero di iterazioni). [F]
14. Calcolo autovalori con algoritmo QR [F] (comunque essere in grado di usare l'algoritmo direttamente dalla riga dei comandi).
15. Quadratura trapezi base (dati: funzione, estremi di integrazione, numero nodi. uscita: valore integrale definito approssimato)
16. Quadratura trapezi con stima errore (dati: funzione, estremi di integrazione, accuratezza richiesta, numero max di livelli. uscita: valore integrale definito approssimato, stima errore)

17. Quadratura Cavalieri-Simpson adattiva ricorsiva (dati: funzione, estremi di integrazione, accuratezza richiesta, numero max di livelli. uscita: valore integrale definito approssimato, stima errore)
18. Quadratura Cavalieri-Simpson a schema fisso [F]
19. Quadratura trapezi adattiva ricorsiva [F]