Calcolo Numerico, IV Esercitazione

CdL Informatica

Esercizi svolti a lezione

1) Scrivere una function Matlab che calcola un'approssimazione della radice quadrata di un numero non negativo, usando una procedura iterativa basata sul metodo di bisezione. La function deve:

- prendere in ingresso un numero non negativo N, due numeri non negativi m, n tali che $m^2 < N < n^2$ ed un numero intero positivo $N_{\max}$;
- implementare il metodo di bisezione a partire dall'intervallo $[m, n]$ ed arrestarlo quando è stato raggiunto il numero massimo di iterazioni $N_{\max}$;
- restituire in uscita l'approssimazione calcolata ed il numero di iterazioni impiegate dal metodo di bisezione.

Eseguire la function dal prompt dei comandi per calcolare la radice quadrata di 6.
Confrontare l'approssimazione ottenuta con quella fornita dal comando `sqrt(6)`.

2) Scrivere una function Matlab che calcola un'approssimazione della radice quadrata di un numero non negativo, usando una procedura iterativa basata sul metodo di Newton. La function deve:

- prendere in ingresso un numero non negativo N, un punto iniziale positivo x_0, un numero $0 < \tau < 1$ ed un numero intero positivo $N_{\max}$;
- implementare il metodo di Newton a partire dal punto iniziale x_0 ed arrestarlo quando un criterio d'arresto basato su τ è soddisfatto, oppure quando è stato raggiunto il numero massimo di iterazioni $N_{\max}$;
- restituire in uscita l'approssimazione calcolata ed il numero di iterazioni impiegate dal metodo di Newton.

Eseguire la function dal prompt dei comandi per calcolare la radice quadrata di 6. Confrontare i risultati ottenuti con quelli della function dell'esercizio precedente.

3) Scrivere una function Matlab che realizza un algoritmo basato sul metodo di bisezione per risolvere il problema $f(x) = 0$. La function deve:

- prendere in ingresso due numeri a, b con $a < b$, un riferimento ad una funzione per il calcolo di f ed un numero intero positivo $N_{\max}$;
- implementare il metodo di bisezione a partire dall'intervallo $[a, b]$ ed arrestarlo quando è stato raggiunto il numero massimo di iterazioni $N_{\max}$;
- restituire in uscita l'approssimazione calcolata ed il numero di iterazioni impiegate dal metodo di bisezione.

Scrivere una function Matlab che realizza un algoritmo basato sul metodo di Newton per risolvere il problema $f(x) = 0$. La function deve:

- prendere in ingresso un punto iniziale x_0, un numero $0 < \tau < 1$, un riferimento ad una funzione per il calcolo di f, un riferimento ad una funzione per il calcolo di f', ed un numero intero positivo $N_{\max}$;
- implementare il metodo di Newton a partire dal punto iniziale x_0 ed arrestarlo quando un criterio d'arresto basato su τ è soddisfatto, oppure quando è stato raggiunto il numero massimo di iterazioni $N_{\max}$;
- restituire in uscita l'approssimazione calcolata ed il numero di iterazioni impiegate dal metodo di Newton.

Scrivere uno script che mette a confronto le due funzioni realizzate per la risoluzione dell'equazione non lineare

$$\arctan(x) = 0.$$

Per il metodo di Newton, scegliere $x_0 = 1$ e $x_0 = 2$. Che cosa si nota al variare dell'iterata iniziale?

Esercizi suggeriti per casa

1) Scrivere una function Matlab che calcola un'approssimazione della radice cubica di un numero, usando una procedura iterativa basata sul metodo di Newton. La function deve:

- prendere in ingresso un numero N, un punto iniziale x_0, un numero positivo ε con $\varepsilon_m < \varepsilon < 1$ ed un numero intero positivo $MAXIT$;
- implementare il metodo di Newton a partire dal punto iniziale x_0 ed arrestarlo quando il criterio d'arresto $|x_k^3 - N| \leq \varepsilon$ è soddisfatto, oppure quando è stato raggiunto il numero massimo di iterazioni $MAXIT$;
- restituire in uscita l'approssimazione calcolata ed il numero di iterazioni impiegate dal metodo di Newton.

Ripetere l'esercizio usando come criterio d'arresto la condizione $|x_{k+1} - x_k| \leq \varepsilon_1 |x_{k+1}| + \varepsilon_2$ dove $\varepsilon_m < \varepsilon_1 < 1$, $\varepsilon_m < \varepsilon_2 < 1$.