(Fac-Simile di) Esame di Calcolo Numerico:

Durata esame: 2h30

1 Ricerca di zeri di funzione

Teoria
Si dia la definizione di ordine di convergenza di un metodo di iterazione e una condizione sufficiente per cui un metodo con funzione di iterazione g derivabile p volte ha ordine di convergenza p.

Esercizio
Data la funzione $f(x) = e^{-x} - 2\cos(x)$ si determini una approssimazione del minimo di $f(x)$ in $[0, 1]$ attraverso il metodo di Newton. Come approssimazione si calcolino solo le prime tre iterate.

2 Algebra Lineare Numerica

Teoria
Dato il sistema lineare $Ax = b$, si dia una stima all'errore relativo alla soluzione del sistema lineare nel caso di perturbazione sul termine noto, ovvero nel caso in cui il termine noto sia $\tilde{b} = b + \delta b$.

Esercizio
Dato il sistema lineare $Ax = b$ dove

$$A = \begin{pmatrix} -2 & -5 & 6 \\ 0 & -1 & -2 \\ 1 & 3 & -1 \end{pmatrix}, \qquad b = \begin{pmatrix} 11 \\ -3 \\ -2 \end{pmatrix}.$$

- Decomporre A attraverso la decomposizione LU.
- Risolvere il sistema $Ax = b$ attraverso l'uso della LU

3 Interpolazione e approssimazione di funzioni

Teoria
Si descrivano le proprietà dell'operatore di interpolazione di Lagrange.

Esercizio
Avendo le seguenti quattro coppie di dati $(0,4)$, $(1,2)$, $(2,2)$, $(3,16)$

- Si faccia la tabella delle differenze divise,
- Si indichi il polinomio di grado non superiore a 3 nella forma di Newton interpolante i dati.

4 Quadratura

Teoria
Si dia la definizione di esattezza di una formula di quadratura.

Esercizio
Sia da calcolare il seguente integrale

$$I = \int_1^3 (2x + \cos(x\pi))\mathrm{d}x$$

- Si approssimi I con la formula di Simpson composta utilizzando 2 suddivisioni in parti uguali dell'intervallo di integrazione.
- Calcolando analiticamente l'integrale, si calcoli l'errore assoluto tra l'approssimazione e il valore esatto.

5 Matlab

Lo script consegnato deve essere eseguibile.
Non verranno valutati script con errori di esecuzione.

Data una funzione $f : [a,b] \to \mathbb{R}$ contenente un unico zero in $I = [a,b]$, è possibile approssimare la posizione di tale zero attraverso i metodi di bisezione o di Newton visti già a lezione.
Il metodo di Halley, parte da un punto iniziale x_0 e, data una funzione f sufficientemente regolare, genera la successione

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{2(f'(x_n))^2 - f(x_n)f''(x_n)},$$

convergente allo zero di f, purché il denominatore rimanga diverso da zero.
A partire dalla function **newton.m**, si crei la function `halley.m`, dove implementare il metodo omonimo. Tra gli input si aggiunga la derivata seconda in quanto necessaria per il metodo. Si usi come criterio di arresto o quando $|x_{n+1} - x_n| < \texttt{tol}$ o quando vengono raggiunte le iterate massime preassegnate. In aggiunta, si modifichi l'errore di controllo prima di applicare il metodo in modo tale che esca nel caso in cui il denominatore sia uguale a zero e, similmente, si aggiunga una modifica interna al ciclo per lo stesso motivo.

In seguito, si scriva uno script denominato **CognomeNome_matricola.m** dove utilizzare i vari metodi per la ricerca dello zero (bisezione, Newton e Halley).
Si definisca la funzione

$$f(x) = \cos(2x)^2 - x^2.$$

Le derivate (prima e seconda) sono

$$f'(x) = -2\sin(4x) - 2x \qquad f''(x) = -8\cos(4x) - 2,$$

e tali funzioni, nello script, dovranno essere definite come anonymous functions associate alle variabili `Df` e `D2f`.
In seguito si approssimi lo zero della funzione usando i metodi di bisezione, di Newton e di Halley, utilizzando le function create e assegnate. Per il metodo di bisezione si consideri l'intervallo utilizzato per disegnare il grafico, ovvero `a=0`, `b=1`. Per tutti i metodi si parta dal valore `x0 = 0.5`, e si utilizzino come parametri di tolleranza `tol = 1e-8` e di numero massimo di iterate `nMax = 1000`.
Si stampi a schermo il numero di iterate impiegate da tutti i metodi aggiungendo al valore anche una frase indicante a cosa corrisponda il valore stampato a schermo.
Si generino i vettori `errB`, `errN`, `errH` dove immagazzinare i valori degli errori assoluti tra la soluzione *esatta*, il cui valore è ottenuto attraverso il comando `fzero(f,0.5)` (dove `f` è la variabile associata alla funzione f), e tutte le iterate dei quattro metodi (ciascun metodo nel vettore corrispondente alla iniziale in maiuscolo).
Si creino inoltre due ulteriori grafici in due ulteriori finestre separate, entrambi in scala semilogaritmica. Il primo contenente gli errori dei metodi della corda e di bisezione in funzione dell'iterata e il secondo con gli errori del metodo di Newton e del metodo di Halley, anch'essi in funzione dell'iterata. Entrambi i grafici siano dotati di legenda e vengano costruiti utilizzando come stile grafico un cerchietto unito da una linea in colori diversi.
Infine, si stampi a schermo la seguente frase completando i puntini con il metodo che ha impiegato meno iterate e aggiungendo dopo i due punti il valore della soluzione ottenuta con tale metodo, il valore deve essere espresso in formato decimale con una cifra prima della virgola e sei dopo

```
Soluzione ottenuta con il metodo ...  :
```

Attenzione: Lo script consegnato deve essere eseguibile. Non verranno valutati script con errori di esecuzione.