Correzione Esame di Calcolo Numerico

31/01/2020

1. 1a) $1 + \texttt{eps} = 1.000000000000000$.
$\texttt{eps} = 2.220446049250313e - 16$ per cui la variazione delle cifre decimali rispetto al numero 1 non è visibile nella rappresentazione `format long`.

1b) $1 + \texttt{eps} > 1$ infatti `eps` è definito come il più piccolo numero macchina positivo tale che $1 + \texttt{eps} > 1$.

2. Integrazione numerica.
2a) La *formula del trapezio semplice* approssima l'integrale

$$\int_a^b f(x)dx$$

sostituendo la funzione integranda $f(x)$ con il polinomio di primo di grado passante per i punti $(a, f(a))$ e $(b, f(b))$, ossia

$$\int_a^b f(x)dx \approx \frac{b-a}{2}\left[f(a) + f(b)\right]$$

La formula ha grado di esattezza 1 perché integra esattamente tutti polinomi di grado ≤ 1 ed esiste almeno un polinomio di grado 2 che non viene integrato esattamente.

2b) La *formula del trapezio composita* si ottiene suddividendo l'intervallo $[a, b]$ in m sottointervalli di ampiezza $H = (b - a)/m$ e applicando ad ogni sottointervallo la formula del trapezio semplice:

$$\int_a^b f(x)dx \approx \frac{H}{2}\left(f(x_0) + 2\sum_{i=1}^{m-1} f(x_i) + f(x_m)\right)$$

con $x_i = a + iH$, $i = 0, \ldots, m$.

2c) Applicando la formula del trapezio composita si ottiene

$$\begin{aligned}
\int_{10}^{20}(x-\sqrt{2})dx &= \int_{10}^{15}(x-\sqrt{2})dx + \int_{15}^{20}(x-\sqrt{2})dx \\
&\approx \tfrac{15-10}{2}[(10-\sqrt{2}) + (15-\sqrt{2})] + \tfrac{20-15}{2}[(15-\sqrt{2}) + (20-\sqrt{2})] \\
&= \tfrac{5}{2}(25-2\sqrt{2}) + \tfrac{5}{2}(35-2\sqrt{2}) = 150 - 10\sqrt{2}\,.
\end{aligned}$$

e coincide con il risultato esatto:

$$\int_{10}^{20}(x-\sqrt{2})dx = \frac{x^2}{2} - \sqrt{2}x\Bigg|_{x=10}^{x=20} = \frac{400}{2} - \frac{100}{2} - 20\sqrt{2} + 10\sqrt{2} = 150 - 10\sqrt{2}$$

come ci si aspetta poichè la formula ha grado di esattezza 1 e la funzione integranda è un polinomio di primo grado.

3. I comandi Matlab necessari per lo svolgimento dei punti 3c), 3d), 3e), 3f) e 3g) sono implementati nel file **Es3.m** che utilizza la function **Backward.m**.

3a) Data una matrice triangolare superiore $U \in \mathbb{R}^{n\times n}$, l'*algoritmo di sostituzione all'indietro* per la risoluzione del sistema lineare $Ux = b$

$$\begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ 0 & u_{22} & u_{23} & \cdots & u_{2n} \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \cdots & 0 & u_{n-1\,n-1} & u_{n-1\,n} \\ 0 & \cdots & 0 & 0 & u_{n\,n} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix}$$

è:

$$\begin{cases} x_n = \dfrac{b_n}{u_{n\,n}} \\ x_i = \dfrac{\left(b_i - \sum_{j=i+1}^{n} u_{i\,j} x_j\right)}{u_{ii}} \qquad\qquad i = n-1, \ldots, 1 \end{cases}$$

3b) Il costo computazione dell'algoritmo di sostituzione all'indietro è pari ad $O(n^2)$ operazioni macchina infatti:

- ad ogni iterazione è prevista una divisione, per un totale di n divisioni
- alla iterata i-esima, per $i = n-1, \ldots, 1$, sono previste $n-i$ moltiplicazioni, per un totale di $\frac{(n-1)n}{2}$ moltiplicazioni
- alla iterata i-esima, per $i = n-1, \ldots, 1$, sono previste $n-i$ somme algebriche, per un totale di $\frac{(n-1)n}{2}$ somme algebriche.

3f) L'operatore "\" di Matlab implementa, per matrici triangolari superiori, il metodo di sostituzione all'indietro e quindi gli errori ottenuti ai punti 3e) e 3f) sono dello stesso ordine di grandezza.

3g) Un esempio di tabella che si può ottenere è:

n	condiz	err "\"	cputime "\"	err Backward	cputime Backward
1	3.9200e+03	1.5543e-15	7.7700e-05	2.2204e-15	7.0360e-04
2	1.4426e+05	7.3308e-13	5.6800e-05	1.3896e-12	2.0160e-04
3	6.8750e+07	8.5503e-10	2.0230e-04	7.5528e-10	2.5720e-04
4	1.8891e+08	5.9886e-10	4.9700e-05	1.8968e-09	2.8280e-04
5	1.2025e+12	6.5189e-06	2.8830e-04	9.0461e-06	7.0300e-04
6	4.0383e+14	3.8319e-04	1.7900e-05	4.7590e-04	2.9920e-04
7	6.7102e+17	1.5413e-02	2.1000e-05	1.4893e-02	3.4060e-04
8	9.6013e+11	1.3831e-05	3.2700e-05	1.7441e-05	5.9260e-04
9	3.8503e+12	4.7471e-06	2.6700e-05	1.3923e-05	4.3430e-04
10	5.8568e+17	1.0121e+00	3.5700e-05	5.1152e-01	4.8680e-04

L'aumento del condizionamento comporta un aumento (anche notevole) dell'errore sul risultato.
L'operatore "\" di Matlab implementa, per matrici triangolari superiori, il metodo di sostituzione all'indietro e quindi anche i tempi computazionali ottenuti ai punti 3e) e 3f) sono molto simili ma all'aumentare di n si risente di questa ottimizzazione.