Laurea Triennnale in Matematica
MATLAB. S1, A.A. 2018-19

Scheda 3

1. Si scrivano i comandi MATLAB necessari per produrre la matrice

$$\mathtt{A} = \begin{bmatrix} 5 & 10 & 15 & \dots & 5000 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix},$$

evitando di fare un ciclo `for` o `while`.

2. Si scriva il vettore riga

$$\mathtt{v} = \begin{bmatrix} 2^1 & 2^2 & 2^3 & \dots & 2^{50} \end{bmatrix},$$

evitando di fare un ciclo `for` o `while`.

3. (a) Scriva un programma per valutare un polinomio scritto nella forma $p(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_0$ in un punto $x_0 \in \mathbb{R}$ utilizzando l'algoritmo di Horner. I dati in ingresso devono essere: un vettore coi coefficienti a_j, $j = 0, \ldots, n$ del polinomio ed x_0.

(b) Modificare il programa precedente in modo che il dato in ingresso x_0 possa essere un vettore e il risultato sia un vettore $p(x_0)$ della stessa dimensione di x_0 coi valori del polinomio in ogni componente di x_0.

(c) Confronti la efficienza dell'algoritmo di Horner con metodi più *naive* per valutare polinomi. Includa *contatori* nel codice MATLAB in grado di monitorare il numero totale di operazioni, distinguendo fra moltiplicazioni/divisioni e somme/sottrazioni.

(d) Confronti la efficienza dei diversi metodi di valutazione rappresentando graficamente il numero totale di operazioni rispetto al grado n del polinomio, per diversi valori di n.

4. Scriva un programma che prenda come dati in ingresso $x_0 \in \mathbb{R}$ e un vettore coi coefficienti di un polinomio $p(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_0$. Utilizzi l'algoritmo di Horner per calcolare i coefficienti $b_0, \ldots, b_n$ tali che

$$\begin{aligned} p(x) =& b_0 + b_1(x - x_0) + \ldots + b_n(x - x_0)^n \\ =& b_0 + (x - x_0)\Big[b_1 + (x - x_0)\big[\cdots[b_{n-1} + b_n(x - x_0)]\cdots\big]\Big]. \end{aligned}$$

Indicazione: Il valore di p in x_0 è il resto della divisione $\frac{p(x)}{x - x_0}$. I coefficienti $b_0, \ldots, b_n \in \mathbb{R}$ si possono ottenere realizzando n divisioni successive per $x - x_0$. Per fare ogni divisione applicare il metodo di Ruffini-Horner.

5. (a) Scriva una funzione MATLAB che calcoli il fattoriale di un numero intero dato $k \in \mathbb{N}_0$. Se il numero non è un intero maggiore o uguale a zero la funzione dovrebbe riportare un messaggio d'errore.

(b) Scriva una funzione MATLAB che per x, n come dati in ingresso, approssimi la funzione esponenziale e^x attraverso la valutazione della somma parziale

$$T_n(x) = \sum_{k=0}^{n} \frac{x^k}{k!}.$$

Scrivere il programma in modo che per x un vettore il risultato $T_n(x)$ sia un vettore della stessa dimensione di x.

(c) Per $x_0 \in \{-10, -5, 5, 10\}$, si rappresenti graficamente l'errore relativo con rispetto ad n, definito da

$$E_n(x_0) := \frac{|T_n(x_0) - \mathrm{e}^{x_0}|}{\mathrm{e}^{x_0}}, \qquad \text{for } 0 \leq n \leq 20.$$

Si salvi la figura in formato PDF col nome `errore_exp.pdf` e i dati x_0, $E_n(x_0)$ in un file di dati MATLAB di nome `dati_error_exp.mat`.

(d) Modifichi la funzione MATLAB scritta precedentemente in modo che prenda come dati in ingresso x e una toleranza *tol* (a.e. $10^{-3}, 10^{-6}$) e approssimi la funzione esponenziale e^x attraverso la valutazione della somma parziale

$$T_n(x) = \sum_{k=0}^{n} \frac{x^k}{k!},$$

per il primo valore di n tale che per $x = (x_j)_{j=1}^{d}$ sia

$$\max_{1 \leq j \leq d} \left| \frac{T_n(x_j) - \mathrm{e}^{x_j}}{\mathrm{e}^{x_j}} \right| \leq tol.$$

Si testi il programma con $x_0 \in \{-10, -5, 5, 10\}$.

6. Si scriva una funzione di MATLAB che prenda come dati di ingresso due vettori e restituisca il prodotto scalare fra di loro. La funzione deve essere chiamata con la sintassi

$$\texttt{p} = \texttt{prodotto}(\texttt{x}, \texttt{y})$$

con

- `x`: vettore in ingresso
- `y`: vettore in ingresso
- `p`: risultato del prodotto scalare di `x` per `y`.

Il programma deve controllare la dimensione dei vettori in ingresso `x,y`, che possono essere righe o colonne, e mostrare un messaggio di errore se non si può calcolare il loro prodotto scalare. Per valutare il prodotto non si deve fare un ciclo `for` ne `while`.

7. Si scriva una funzione MATLAB chiamata `cappello` che valuti la funzione matematica:

$$f(x,h) = \begin{cases} \dfrac{h+x}{h}, & \text{se} \quad -h \le x \le 0, \\ \dfrac{h-x}{h}, & \text{se} \quad 0 \le x \le h, \\ 0, & \text{altrimenti.} \end{cases} \tag{1}$$

dove $h > 0$ è un parametro scalare e x è un numero reale. La funzione MATLAB deve potere prendere come dati d'ingresso un vettore riga o colonna `x`, uno scalare `h>0` e restituire un vettore della stessa dimensione di `x` coi valori corrispondenti di f. Il programma deve dare errore se `h` ≤ 0 e si deve evitare di fare un ciclo `for` o `while` per valutare la funzione nel vettore `x`. Scrivere inoltre il programma in modo che l'istruzione `help` fornisca una descrizione della funzione.

Esempi sulla finestra di comandi:

```
>> x=[-1:.2:1];
>> h=.5;
>> cappello(x,h)

ans =

        0    0    0    0.2000    0.6000    1.0000    0.6000    0.2000    0    0
0
>> help cappello
Funzione che valuta una funzione lineare a tratti.
```

8. **Interpolazione lineare a tratti**. Si osservi che fissato un intervallo $[a,b]$, $N \ge 1$ nodi equispaziati

$$x_j = a + jh, \quad h = \frac{b-a}{N}, \quad j = 0, \ldots, N, \tag{2}$$

e valori $y_j \in \mathbb{R}$, $j = 0, \ldots, N$, associati alle ascisse $x_j \in \mathbb{R}$, $j = 0, \ldots, N$, la funzione matematica

$$p(x) = \sum_{j=0}^{N} y_j f(x - x_j, h), \tag{3}$$

con f in (1) definisce un interpolante lineare a tratti per i dati $(x_j, y_j)_{j=0}^N$.

Si scriva una funzione MATLAB chiamata `interpola` che prenda come dati in ingresso `a`, `b`, `N` e una function-handle `fdata` e restituisca i valori della funzione $p(y)$ con p definita in (3), x_j in (2) e $y_j = fdata(x_j)$, $j = 1, \ldots, N$.

Si testi il programma con la funzione matematica

$$g(x) = \frac{1}{1 + 25x^2}, \qquad x \in [-5, 5].$$

Si produca inoltre un grafico di g e di `interpola` che permetta di visualizzare come si comporta l'interpolante per diversi valori di N.

9* Si scriva un programma MATLAB di nome `settemezzo` che permetta al utente di gioccare al sette e mezzo con la macchina.