Corso di Laurea in Informatica – Università di Ferrara
Calcolo Numerico
Esercizi dalla prova scritta e pratica del 26/01/2018

Esercizio 1

1) (T) (2 punti) Determinare la rappresentazione del numero $y = (-74301.92)_{10}$ secondo il formato ANSI standard IEEE in semplice precisione.

2) (M) (2 punti) Realizzare un M-script file Matlab che, usando la function `ruffiniHorner` in allegato, calcoli e stampi il valore del polinomio $p(x) = 3(x-2)^2(x+1)x^3 - (x+\pi)x^2 - 2$ e delle sue derivate prima $p'(x)$ e seconda $p''(x)$ in un prefissato punto x_0 accettato in input. A scopo esemplificativo, l'output delle stampe a monitor deve essere del tipo:

```
Esecizio 1

Inserire il punto (numero reale) nel quale valutare il polinomio: x0 = ...

Valore del polinomio in x0: p(x0) = ...
Valore della derivata prima in x0: p'(x0) = ...
Valore della derivata seconda in x0: p''(x0) = ...
```

Esercizio 2

1) (T) (2 punti) Valutare l'errore inerente e l'errore algoritmico nel calcolo dell'espressione:

$$\varphi(x) = \frac{1}{\sqrt{x} - \sqrt{\pi}}$$

Esistono valori di x per i quali il problema è mal condizionato? Esiste una diversa formulazione di tale espressione? In caso affermativo, dire se è numericamente più vantaggiosa, motivando la risposta.

2) (M) (3 punti) Realizzare un M-function file per la valutazione della seguente funzione:

$$f(\alpha) = \sum_{\substack{k=0 \\ k \text{ pari}}}^{n} \sin(k\alpha) \quad + \sum_{\substack{k=0 \\ k \text{ dispari}}}^{n} \cos(k\alpha) \qquad \alpha \in \mathbb{R}, \quad n \in \mathbb{N}.$$

Per un fissato valore di α, la function deve eseguire il calcolo di $f(\alpha)$ in due modi:

(1) mediante un (unico) ciclo `for`;

(2) in un unico comando, mediante sintassi vettoriale.

Temporizzare l'esecuzione nei due casi mediante le funzioni `tic` e `toc`. La function restituisca in output i valori di $f(\alpha)$ e i tempi rilevati. Scrivere un M-script file di prova per questa function, usando i valori $\alpha = 0.15$ e $n = 5000$. Visualizzare a video i due risultati ottenuti per $f(\alpha)$ in formato esponenziale e i rispettivi tempi di calcolo, mediante un'istruzione di stampa formattata.

Appendice: codici forniti

```
function [r, q] = ruffiniHorner(p, a)
% RUFFINIHORNER - Schema di Ruffini-Horner
%  Calcola il valore di un polinomio p(x) nel punto x = a e i coefficienti
%  del polinomio quoziente q(x) = p(x) / (x - a)
% SYNOPSIS
```

```
%   [r, q] = ruffiniHorner(p, a)
% INPUT
%   p (double array) - Vettore dei coefficienti del polinomio, ordinati
%                      da quello di grado piu' alto a quello di grado zero
%   a (double)       - Punto (numero reale) in cui calcolare il polinomio
% OUTPUT
%   r (double)       - Valore del polinomio nel punto x = a
%   q (double array) - Vettore dei coefficienti del polinomio quoziente
%                      q(x) = p(x) / (x - a)
%
  r = [];
  if ( isempty(p) )
    q = [];
    warning('Il vettore p dei coefficienti e'' vuoto');
    return
  elseif ( isempty(a) )
    q = p;
    warning('Il punto ''a'' in cui valutare il polinomio e'' vuoto');
    return
  else
    n = numel(p) - 1;  % grado del polinomio
    q = zeros(n, 1);
    q(1) = p(1);
    for i = 2 : n+1
      q(i) = q(i-1)*a + p(i);
    end
    r = q(n+1);
    q = q(1:n);
  end
end % fine della function ruffiniHorner
```

Soluzioni e commenti

Nel seguito sono riportate le soluzioni degli esercizi del compito, includendo in generale solo quei commenti che risultino fondamentali per la comprensione della soluzione. Qualora uno stesso esercizio possa eventualmente essere risolto in più modi, in alcuni casi si riportano le descrizioni anche di qualche modo alternativo. Si invitano gli studenti a contattare il docente per eventuali dubbi o chiarimenti.

Soluzione esercizio 1

Teoria

74301	/2
37150	1
18575	0
9287	1
4643	1
2321	1
1160	1
580	0
290	0
145	0
72	1
36	0
18	0
9	0
4	1
2	0
1	0
0	1

0.92	×2
0.84	1
0.68	1
0.36	1
0.72	0
0.44	1
0.88	0
0.76	1
0.52	1
⋮	⋮

$$
\begin{aligned}
(74301)_{10} &= (10010001000111101)_2 \\
(0.92)_{10} &= (0.11101011\ldots)_2 \\
y &= (-74301.92)_{10} \\
&= (-10010001000111101.11101011\ldots)_2 \\
&= (-1.001000100011110111101011\ldots)_2 \cdot 2^{(+10000)_2}
\end{aligned}
$$

$$
\begin{aligned}
y < 0 &\Rightarrow s = 1 \\
p &= (+10000)_2 = (+16)_{10} \\
&\Rightarrow \widetilde{p} = p + \mathit{bias} = 16 + 127 = 143 = 128 + 8 + 4 + 2 + 1 \\
&\qquad = (10001111)_2 \\
m &= (1.00100010001111011110101)_2 \\
&\Rightarrow \widetilde{m} = 00100010001111011110101
\end{aligned}
$$

(attenzione: troncamento della mantissa alla 24-esima cifra binaria)

$$
\mathrm{IEEE}_{32}(y) = 1\,\underbrace{10001111}_{\widetilde{p}}\,\underbrace{00100010001111011110101}_{\widetilde{m}}
$$

Matlab

Trasformiamo il polinomio in forma standard eseguendo i calcoli:

$$
\begin{aligned}
p(x) &= 3(x-2)^2(x+1)x^3 - (x+\pi)x^2 - 2 = 3(x^2-4x+4)(x+1)x^3 - x^3 - \pi x^2 - 2 \\
&= 3(x^3 - 4x^2 \cancel{+4x} + x^2 \cancel{-4x} + 4)x^3 - x^3 - \pi x^2 - 2 = 3x^6 - 9x^5 + 11x^3 - \pi x^2 - 2
\end{aligned}
$$

Contenuto dell'M-script file `esercizio1.m`:

```
% Cognome Nome
% Matricola
close all
clear all
%----------------------------
%  esercizio 1 - 26/01/2018
%----------------------------
disp('Esecizio 1');
p = [3 -9 0 11 -pi 0 -2]';
x0 = input('Inserire il punto (numero reale) nel quale valutare il polinomio: x0 = ');
[r1, q1] = ruffiniHorner( p,  x0 );
[r2, q2] = ruffiniHorner( q1, x0 );
[r3, q3] = ruffiniHorner( q2, x0 );
fprintf('\nValore del polinomio in x0: p(x0) = %g', r1);
fprintf('\nValore della derivata prima in x0: p''(x0) = %g', r2);
fprintf('\nValore della derivata seconda in x0: p''''(x0) = %g\n\n', 2*r3);
```