

Calcolo Numerico Con Laboratorio Di MATLAB

Daniel Xhakalliu

4 aprile 2022

Indice

1	Testi degli esercizi in MATLAB	3
1.1	Esercizio	3
1.2	Esercizio	3
1.3	Esercizio	3
2	Testi dei problemi in MATLAB	4
2.1	Problema	4
2.2	Problema	4
2.3	Problema	5
3	Soluzioni ai problemi	6
3.1	Soluzione al problema 2.1	6
3.2	Soluzione al problema 2.2	7
3.3	Soluzione al problema 2.3	8
4	Appendice	10
4.1	Funzioni MATLAB	10
4.1.1	Metodo Ruffini Horner	10
4.1.2	Formula dei Trapezi	11
4.1.3	Valore estrapolato	11
4.2	Script MATLAB	12
4.2.1	Script problema 2.1	12
4.2.2	Script problema 2.2	13
4.2.3	Script problema 2.3	15

1 Testi degli esercizi in MATLAB

In seguito verranno enunciati gli esercizi dai quali abbiamo implementato dei programmi in MATLAB necessari per risolvere i problemi.

1.1 Esercizio

Scrivere un programma MATLAB che implementa l'algoritmo di valutazione del polinomio d'interpolazione in uno o più punti. Il programma deve:

- prendere in input tre vettori (a componenti reali) $[x_0, x_1, \dots, x_n]$, $[y_0, y_1, \dots, y_n]$, $[t_1, \dots, t_m]$, con x_0, x_1, \dots, x_n tutti distinti;
- restituire in output il vettore $[p(t_1), \dots, p(t_m)]$ che contiene le valutazioni nei punti t_1, \dots, t_m del polinomio $p(x)$ interpolante i dati $(x_0, y_0), \dots, (x_n, y_n)$.

1.2 Esercizio

Scrivere un programma Matlab che implementa la formula dei trapezi. Il programma deve:

- prendere in input gli estremi a, b di un intervallo, una funzione $f(x)$ definita su $[a, b]$ e il numero $n \geq 1$ di sottointervalli in cui viene suddiviso $[a, b]$;
- restituire in output I_n , l'approssimazione di $\int_a^b f(x) dx$ data dalla formula dei trapezi di ordine n .

1.3 Esercizio

Usando i programmi creati per risolvere gli Esercizi 1.1 e 1.2, scrivere un programma Matlab che implementa il metodo di estrapolazione. Il programma deve:

- prendere in input gli estremi a, b di un intervallo, una funzione $f(x)$ definita su $[a, b]$ e un vettore $[n_0, n_1, \dots, n_m]$ di numeri $n_0, n_1, \dots, n_m \geq 1$ tutti distinti;
- restituire in output il valore estrapolato $p(0)$, dove $p(x)$ è il polinomio d'interpolazione dei dati $(h_0^2, I_{n_0}), (h_1^2, I_{n_1}), \dots, (h_m^2, I_{n_m})$ e h_0, h_1, \dots, h_m sono i passi di discretizzazione delle formule dei trapezi $I_{n_0}, I_{n_1}, \dots, I_{n_m}$ per approssimare $\int_a^b f(x) dx$.

2 Testi dei problemi in MATLAB

2.1 Problema

Si consideri la funzione \sqrt{x} .

- (a) Sia $p(x)$ il polinomio d'interpolazione di \sqrt{x} sui nodi

$$x_0 = 0, x_1 = \frac{1}{64}, x_2 = \frac{4}{64}, x_3 = \frac{9}{64}, x_4 = \frac{16}{64}, x_5 = \frac{25}{64}, x_6 = \frac{36}{64}, x_7 = \frac{49}{64}, x_8 = 1.$$

Calcolare il vettore colonna

$$[p(\zeta_1) - \sqrt{\zeta_1}, p(\zeta_2) - \sqrt{\zeta_2}, \dots, p(\zeta_{21}) - \sqrt{\zeta_{21}}]^T$$

dove $\zeta_i = \frac{i-1}{20}$ per $i = 1, \dots, 21$ e osservare in che modo varia la differenza $p(\zeta_i) - \sqrt{\zeta_i}$ al variare di i da 1 a 21.

- (b) Tracciare il grafico di \sqrt{x} e di $p(x)$ sull'intervallo $[0, 1]$, ponendo i due grafici su un'unica figura e inserendo una legenda che ci dica qual è la funzione \sqrt{x} e qual è il polinomio $p(x)$.

2.2 Problema

Si consideri la funzione

$$f(x) = e^x.$$

Per ogni intero $n \geq 1$ indichiamo con I_n la formula dei trapezi di ordine n per approssimare

$$I = \int_0^1 f(x) dx = 1.7182818284590 \dots$$

- (a) Per ogni fissato $\epsilon > 0$ determinare un $n = n(\epsilon)$ tale che $|I - I_n| \leq \epsilon$.
- (b) Costruire una tabella che riporti vicino ad ogni $\epsilon \in \{10^{-1}, 10^{-2}, \dots, 10^{-10}\}$:
- il numero $n(\epsilon)$;
 - il valore I_n per $n = n(\epsilon)$;
 - il valore esatto I (in modo da confrontarlo con I_n);
 - l'errore $|I - I_n|$ (che deve essere $\leq \epsilon$).
- (c) Calcolare le approssimazioni di I ottenute con le formule dei trapezi I_2, I_4, I_8, I_{16} e confrontarle con il valore esatto I .

- (d) Sia $p(x)$ il polinomio d'interpolazione dei valori I_2, I_4, I_8, I_{16} sui nodi $h_2^2, h_4^2, h_8^2, h_{16}^2$, dove $h_2 = \frac{1}{2}, h_4 = \frac{1}{4}, h_8 = \frac{1}{8}, h_{16} = \frac{1}{16}$ sono i passi di discretizzazione relativi alle formule dei trapezi I_2, I_4, I_8, I_{16} rispettivamente. Calcolare $p(0)$ e confrontare $I_2, I_4, I_8, I_{16}, p(0)$ con il valore esatto I . Che cosa si nota?

2.3 Problema

Consideriamo la funzione $f(x) = x^2 e^{-x}$ e indichiamo con I_n la formula dei trapezi di ordine n per approssimare $I = \int_0^1 f(x) dx$.

- (a) Calcolare I prima manualmente e poi con la funzione simbolica **int** di MATLAB;
- (b) Calcolare $I_5, I_{10}, I_{20}, I_{40}$.
- (c) Calcolare $p(0)$, dove $p(x)$ è il polinomio d'interpolazione dei dati $(h_0^2, I_5), (h_1^2, I_{10}), (h_2^2, I_{20}), (h_3^2, I_{40})$ e h_0, h_1, h_2, h_3 sono i passi di discretizzazione delle formule dei trapezi $I_5, I_{10}, I_{20}, I_{40}$.
- (d) Riportare in una tabella:
- i valori $I_5, I_{10}, I_{20}, I_{40}, p(0)$;
 - gli errori $|I_5 - I|, |I_{10} - I|, |I_{20} - I|, |I_{40} - I|, |p(0) - I|$.
- (e) Posto $\epsilon = |p(0) - I|$, determinare un n in modo tale che la formula dei trapezi I_n fornisca un'approssimazione di I con errore $|I_n - I| \leq \epsilon$. Calcolare successivamente I_n e verificare che effettivamente $|I_n - I| \leq \epsilon$.

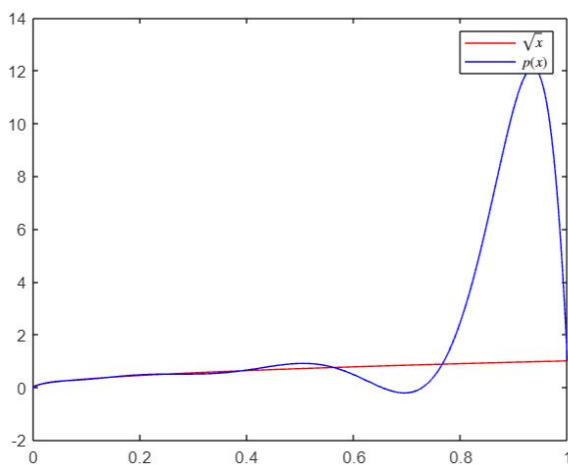
3 Soluzioni ai problemi

3.1 Soluzione al problema 2.1

(a) Il vettore colonna è dato da:

$$\begin{pmatrix} 0 \\ 0.009373456935820 \\ -0.016624898598359 \\ 0.006265159516694 \\ 0.026059100541981 \\ -0.000000000000000 \\ -0.046798842893449 \\ -0.052843679514482 \\ 0.019043791981465 \\ 0.136657922266043 \\ 0.195969221000565 \\ 0.070222900207983 \\ -0.298665479678432 \\ -0.793827451939212 \\ -1.047857448417112 \\ -0.461689802877413 \\ 1.600121563949947 \\ 5.337600132745780 \\ 9.648720381277604 \\ 10.731478361986419 \\ -0.000000000000199 \end{pmatrix}$$

(b) La figura seguente mostra il grafico di \sqrt{x} e di $p(x)$ sull'intervallo $[0, 1]$:



3.2 Soluzione al problema 2.2

(a) Dato un qualsiasi $\epsilon > 0$, la formula $n = \frac{0.4759448347}{\sqrt{\epsilon}}$ determina un n tale che $|I - I_n| \leq \epsilon$.

(b) La tabella richiesta:

ϵ	n	I_n	I	$ I - I_n $
10^{-1}	2	1.75393109246483	1.71828182845905	$3.56 \cdot 10^{-2}$
10^{-2}	5	1.72400561978279	1.71828182845905	$5.72 \cdot 10^{-3}$
10^{-3}	16	1.71884112857999	1.71828182845905	$5.59 \cdot 10^{-4}$
10^{-4}	48	1.71834397651311	1.71828182845905	$6.21 \cdot 10^{-5}$
10^{-5}	151	1.71828810844886	1.71828182845905	$6.27 \cdot 10^{-6}$
10^{-6}	476	1.71828246043305	1.71828182845905	$6.31 \cdot 10^{-7}$
10^{-7}	1506	1.71828189159303	1.71828182845905	$6.31 \cdot 10^{-8}$
10^{-8}	4760	1.71828183477879	1.71828182845905	$6.31 \cdot 10^{-9}$
10^{-9}	15051	1.71828182909114	1.71828182845905	$6.32 \cdot 10^{-10}$
10^{-10}	47595	1.71828182852224	1.71828182845905	$6.31 \cdot 10^{-11}$

(c) In seguito vengono mostrate le approssimazioni di I con le formule dei trapezi I_2, I_4, I_8, I_{16} e gli errori commessi da essi.

n	I_n	$ I - I_n $
2	1.75393109246483	$3.56 \cdot 10^{-2}$
4	1.72722190455752	$8.94 \cdot 10^{-3}$
8	1.7205185921643	$2.23 \cdot 10^{-3}$
16	1.71884112857999	$5.59 \cdot 10^{-4}$

(d) Osservando la tabella successiva possiamo notare che al crescere di n le approssimazioni usando la formula dei trapezi sono più accurate, ma il valore estrapolato $p(0)$ approssima commettendo l'errore più piccolo.

n	I_n	I	$ I - I_n $
2	1.75393109246483	1.71828182845905	$3.56 \cdot 10^{-2}$
4	1.72722190455752	1.71828182845905	$8.94 \cdot 10^{-3}$
8	1.7205185921643	1.71828182845905	$2.23 \cdot 10^{-3}$
16	1.71884112857999	1.71828182845905	$5.59 \cdot 10^{-4}$
	$p(0)$	I	$ I - p(0) $
-	1.71828182846039	1.71828182845905	$1.34 \cdot 10^{-12}$

3.3 Soluzione al problema 2.3

(a) Iniziamo calcolando manualmente l'integrale.

$$\int_0^1 x^2 e^{-x} dx.$$

Integriamo per parti $e^{-x}x^2$,

$$\int f dg = fg - \int g df$$

dove $f = x^2$, $dg = e^{-x} dx$, $df = 2x dx$, $g = -e^{-x}$:

$$\int_0^1 x^2 e^{-x} dx = (-e^{-x}x^2)\Big|_0^1 + 2 \int_0^1 e^{-x}x dx = -\frac{1}{e} + 2 \int_0^1 e^{-x}x dx.$$

Integriamo per parti $e^{-x}x$,

$$\int f dg = fg - \int g df$$

dove $f = x$, $dg = e^{-x} dx$, $df = dx$, $g = -e^{-x}$:

$$-\frac{1}{e} + 2 \int_0^1 e^{-x}x dx = -\frac{1}{e} + (-2e^{-x}x)\Big|_0^1 + 2 \int_0^1 e^{-x} dx = -\frac{3}{e} + 2 \int_0^1 e^{-x} dx.$$

Utilizzando il teorema fondamentale del calcolo integrale:

$$-\frac{3}{e} + 2 \int_0^1 e^{-x} dx = -\frac{3}{e} + (-2e^{-x})\Big|_0^1 = 2 - \frac{5}{e} = 0.160602794142788$$

Utilizzando la funzione **int** di MATLAB abbiamo che $I = 2 - \frac{5}{e}$, che è uguale alla nostra risposta.

(b) Nella tabella seguente sono mostrati i valori di $I_5, I_{10}, I_{20}, I_{40}$:

n	I_n
5	0.161816576820683
10	0.160908578632096
20	0.160679386811339
40	0.160621951474857

- (c) Sia $p(x)$ il polinomio d'interpolazione dei dati $(h_0^2, I_5), (h_1^2, I_{10}), (h_2^2, I_{20}), (h_3^2, I_{40})$ dove h_0, h_1, h_2, h_3 sono i passi di discretizzazione delle formule dei trapezi $I_5, I_{10}, I_{20}, I_{40}$. Il valore estrapolato $p(0)$ è uguale a 0.160602794142805.
- (d) La tabella richiesta:

n	I_n	$ I_n - I $
5	0.161816576820683	$1.21 \cdot 10^{-3}$
10	0.160908578632096	$3.05 \cdot 10^{-4}$
20	0.160679386811339	$7.65 \cdot 10^{-5}$
40	0.160621951474857	$1.91 \cdot 10^{-5}$
	$p(0)$	$ p(0) - I $
-	0.160602794142805	$1.62 \cdot 10^{-14}$

- (e) Sia $\epsilon = |p(0) - I| = 1.62 \cdot 10^{-14}$. Utilizzando la formula $n = \frac{0.4082482905}{\sqrt{\epsilon}}$, troviamo un $n = 3209335$ tale che:

$$2.77 \cdot 10^{-17} = |I_n - I| \leq \epsilon = 1.62 \cdot 10^{-14}$$

Calcolando $I_{3209335}$ otteniamo:

$$I_{3209335} = 0.160602794142788$$

4 Appendice

In questa sezione sono presenti tutti i programmi MATLAB utilizzati per svolgere gli esercizi e i problemi presenti in questo documento.

4.1 Funzioni MATLAB

4.1.1 Metodo Ruffini Horner

INPUT

x : un vettore $[x_0, x_1, \dots, x_n]$ dove x_0, x_1, \dots, x_n sono tutti distinti.

y : un vettore $[y_0, y_1, \dots, y_n]$.

t : un vettore $[t_0, t_1, \dots, t_m]$.

OUTPUT

PdiT: il vettore $[p(t_0), p(t_1), \dots, p(t_m)]$ che contiene la valutazione nei punti t_0, t_1, \dots, t_m del polinomio $p(x)$ interpolante i dati $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$.

```
1 function PdiT = RuffiniHorner(x, y, t)
2 a = length(x);
3 %Calcolo la tabella delle differenze divise.
4 M = zeros(a, a);
5
6 %La prima colonna e' uguale a Y
7 M(:,1) = y(:);
8
9 %Calcolo il resto della tabella
10 for col = 2:a
11     for row = col:a
12         M(row, col) = (M(row, col - 1) - M(col - 1, col - 1) )
13         / (x(row) - x(col - 1));
14     end
15 end
16 %Calcoliamo P(t)
17 PdiT = zeros(1, length(t));
18 for i=1:length(t)
19     h = 0;
20     for j=a:-1:1
21         h = M(j, j) + (t(i) - x(j))*h;
22     end
23     PdiT(i) = h;
24 end
25 end
```

4.1.2 Formula dei Trapezi

INPUT

a : estremo inferiore dell'intervallo.

b : estremo superiore dell'intervallo.

$funx$: una funzione $f(x)$ definita su $[a, b]$.

n : il numero di sottointervalli in cui viene suddiviso $[a, b]$.

OUTPUT

I_n : l'approssimazione $\int_a^b f(x) dx$ data dalla formula dei trapezi di ordine n .

```
1 function In = FormulaDeiTrapezi(a, b, funx, n)
2
3 %Trovo il passo di discretizzazione (l'ampiezza)
4 h = (b - a) / n;
5
6 %Trovo il valore dentro le parentesi della formula
7 s = (funx(a) + funx(b)) / 2;
8
9 for j = 1:n-1
10     s = s + funx(a + j * h);
11 end
12
13 %Formula dei trapezi
14 In = h * s;
15 end
```

4.1.3 Valore estrapolato

INPUT

a : estremo inferiore dell'intervallo.

b : estremo superiore dell'intervallo.

f : una funzione $f(x)$ definita su $[a, b]$.

n : un vettore $[n_0, n_1, \dots, n_m]$ di numeri distinti.

OUTPUT

p_0 : il valore estrapolato $p(0)$, dove $p(x)$ è il polinomio d'interpolazione dei dati $(h_0^2, I_{n_0}), (h_1^2, I_{n_1}), \dots, (h_m^2, I_{n_m})$ e h_0, h_1, \dots, h_m sono i passi di discretizzazione delle formule dei trapezi $I_{n_0}, I_{n_1}, \dots, I_{n_m}$ per approssimare $\int_a^b f(x) dx$.

```

1 function p0 = Estrapolazione(a, b, f, n)
2
3 %Calcolo le formule dei trapezi
4 In = zeros(length(n), 1);
5 h = zeros(length(n), 1);
6
7 for i = 1:length(In)
8     In(i) = FormulaDeiTrapezi(a, b, f, n(i));
9     h(i) = (b - a) / n(i);
10 end
11
12 %Calcolo p(0) usando RuffiniHorner
13 p0 = RuffiniHorner(h.^2, In, 0);
14 end

```

4.2 Script MATLAB

4.2.1 Script problema 2.1

(a)

```

1 funx = @(x) sqrt(x);
2
3 x = [0; 1/64; 4/64; 9/64; 16/64; 25/64; 36/64; 49/64; 1];
4
5 %Calcolo il vettore y
6 y = funx(x);
7
8 %Calcolo il vettore colonna richiesto dal problema.
9 pK = zeros(21, 1);
10 for i=1:21
11     pK(i) = RuffiniHorner(x, y, (i-1)/20) - funx((i-1)/20);
12 end
13 disp(pK);

```

(b)

```

1 funx = @(x) sqrt(x);
2
3 x = [0; 1/64; 4/64; 9/64; 16/64; 25/64; 36/64; 49/64; 1];
4 punti = linspace(0, 1, 1000);

```

```

5
6 % Grafico di sqrt(X)
7 plot(punti, funx(punti), 'red');
8 hold on
9
10 % Grafico di p(x)
11 plot(punti, RuffiniHorner(x, funx(x), punti, 'blue'));
12 hold off
13
14 legend('$\sqrt{x}$' , '$p(x)$', 'Interpreter', 'latex');

```

4.2.2 Script problema 2.2

(a)

```

1 syms x;
2 f = exp(x);
3 a = 0;
4 b = 1;
5
6 f_2 = matlabFunction(diff(f,2));
7 K = abs(max(f_2(linspace(a,b))));
8
9 n = @(e) sqrt(((b-a)^3*K)/(12*e));
10
11 disp(n);

```

(b)

```

1 syms x;
2 f = exp(x);
3 a = 0;
4 b = 1;
5
6 %Funzione che calcola n(e)
7 f_2 = matlabFunction(diff(f,2));
8 K = abs(max(f_2(linspace(a,b))));
9
10 ne = @(e) sqrt(((b-a)^3*K)/(12*e));
11
12 f = matlabFunction(f);
13
14 %Vettore colonna degli errori
15 eV = [10^(-1); 10^(-2); 10^(-3); 10^(-4); 10^(-5);
        10^(-6); 10^(-7); 10^(-8); 10^(-9); 10^(-10)];
16
17 %Il valore esatto del integrale della funzione
18 I = integral(f, 0, 1);
19

```

```

20 %Creo i vettori che saranno le colonne della nostra
    tabella
21 I = ones(10, 1).*I;
22 In = zeros(10, 1);
23 n = zeros(10, 1);
24 e = zeros(10, 1);
25
26 %Aggiorno i valori delle righe della tabella
27 for i=1:10
28     n(i) = ceil(ne(eV(i)));
29     In(i) = FormulaDeiTrapezi(0, 1, f, n(i));
30     e(i) = abs(I(i) - In(i));
31 end
32
33 %Creo la tabella
34 data = table(eV, n, In, I, e);
35
36 data.Properties.VariableNames = {'Epsilon' 'n' 'In' 'I' ,
    |I - In|'};
37
38 disp(data);

```

(c)

```

1 f = @(x) exp(x);
2
3 %Il valore esatto del integrale della funzione
4 I = integral( f, 0, 1);
5
6 %Vettore dei n
7 nV = [2; 4; 8; 16];
8
9 %Calcoliamo il vettore dei In
10 In = zeros(4, 1);
11 for i=1:4
12     In(i) = FormulaDeiTrapezi(0, 1, f, nV(i));
13 end
14
15 %Vettore dei errori I-In per ogni n
16 err = abs(ones(4,1)*I - In);
17
18 data = table(nV, In, err);
19 data.Properties.VariableNames = {'n' 'In' '|I - In|'};
20
21 disp(data);

```

(d)

```

1 f = @(x) exp(x);

```

```

2
3 %Il valore esatto del integrale della funzione
4 I = integral(f, 0, 1);
5
6 %Vettore dei n
7 nV = [2; 4; 8; 16];
8
9 %Valore p(0)
10 p0 = Estrapolazione(0, 1, f, nV);
11
12 %Calcoliamo il vettore dei In
13 In = zeros(4, 1);
14 for i=1:4
15     In(i) = FormulaDeiTrapezi(0, 1, f, nV(i));
16 end
17
18 %Confrontiamo InV, p(0) con I
19 err = abs([In; p0] - ones(5,1)*I);
20
21 data = table([In; p0], ones(5,1)*I, err);
22 data.Properties.VariableNames = {'In', p(0), 'I', '|I - In|', 'err'};
23
24 disp(data);

```

4.2.3 Script problema 2.3

(a)

```

1 syms x
2 f = x.^2 .* exp(-x);
3 I = int(f, 0, 1);
4
5 disp(I);

```

(b)

```

1 f = @(x)(x.^2) .* exp(-x);
2
3 n = [5; 10; 20; 40];
4
5 %Calcolo la formula dei trapezi per ogni n
6 In = zeros(4, 1);
7 for i=1:4
8     In(i) = FormulaDeiTrapezi(0, 1, f, n(i));
9 end
10
11 data = table(n, In);
12 disp(data);

```

(c)

```
1 f = @(x)(x.^2) .* exp(-x);
2
3 n = [5; 10; 20; 40];
4
5 %Calcolo il valore estrapolato p(0)
6 p0 = Estrapolazione(0, 1, f, n);
7
8 disp(p0);
```

(d)

```
1 f = @(x)(x.^2) .* exp(-x);
2
3 n = [5; 10; 20; 40];
4
5 %Integrale di f
6 I = integral(f, 0, 1);
7
8 %Calcolo la formula dei trapezi per ogni n
9 In = zeros(4, 1);
10 for i=1:4
11     In(i) = FormulaDeiTrapezi(0, 1, f, n(i));
12 end
13
14 %Calcolo il valore estrapolato p(0)
15 p0 = Estrapolazione(0, 1, f, n);
16
17 %Mostro la tabella
18 data = table(In, abs(In - I));
19 data.Properties.VariableNames = {'In' '|In - I|'};
20
21 data1 = table(p0, abs(p0 - I));
22 data1.Properties.VariableNames = {'p(0)' '|p(0) - I|'};
23
24 disp(data);
25 disp(data1);
```

(e)

```
1 syms x;
2 funx = (x.^2) .* exp(-x);
3
4 n = [5; 10; 20; 40];
5
6 %Integrale di funx
7 f = matlabFunction(funx);
8
9 I = integral(f, 0, 1);
```



```

10
11 %Calcolo il valore estrapolato p(0)
12 p0 = Estrapolazione(0, 1, f, n);
13
14 err = abs(I - p0);
15
16 %Troviamo un n che soddisfa la regola
17 f_2 = matlabFunction(diff(funx,2));
18 K = abs(max(f_2(linspace(0,1)))));
19
20 n = @(e) sqrt(((1-0)^3*K)/(12*e));
21 x = ceil(n(err));
22
23 In = FormulaDeiTrapezi(0, 1, f, x);
24
25 %Costruisco la tabella
26 data = table(x, In, abs(In - I), err);
27 data.Properties.VariableNames = {'n' 'In' '|In - I|' 'epsilon'};
28 disp(data);

```