

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 11 luglio 2006

Creare un programma che

1. Chieda all'utente l'input di un k reale.

2. Tabuli la funzione $f(x) = e^{\left(\frac{x^2 + x}{x^2 + k^2}\right) - \frac{1}{2}}$ nell'intervallo $[-2, 2]$ con passo 0.05.

Ovvero crei due arrays X e Y , una con i valori delle x una coi valori delle y della funzione.

3. Cerchi il massimo e il minimo della funzione nell'intervallo e scriva sul display i valori del massimo e del minimo e i punti di massimo e di minimo.
4. Dica se la funzione ha sicuramente uno zero nell'intervallo, usando il teorema degli zeri ed esaminando i valori del massimo e del minimo.

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 14 luglio 2006

La serie di Fibonacci

Creare un programma che

1. Chieda all'utente l'input di un numero n intero e positivo (eventuale verifica).
2. Costruisca un array F il cui i -mo elemento sia l' i -esimo elemento della successione di Fibonacci che è così definita:

$$F(1) = \mathbf{1} \quad F(2) = \mathbf{1} \quad F(3) = 1 + 1 = \mathbf{2} \quad F(4) = 1 + 2 = \mathbf{3}$$

$$F(5) = 2 + 3 = \mathbf{5} \quad \dots \quad F(i) = F(i-2) + F(i-1)$$

La serie abbia termine con l'ultimo numero di Fibonacci minore o uguale a n .

3. Costruisca un secondo array R il cui i -mo elemento sia il rapporto tra l' i -esimo elemento della successione di Fibonacci e l' $i+1$ -mo e quindi consenta di esaminare il limite di questi rapporti
4. Chieda all'utente l'input di un numero p intero, positivo e minore o uguale a n (eventuale verifica) e verifichi se p fa parte o meno della serie di Fibonacci costruita.

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 12 settembre 2006

Interpolazione spline lineare

Costruire un file **dati.txt** in cui siano riportati (separati da [return]) gli 11 numeri a lato che sono i valori della funzione $f(x)$ nei punti $x_0 = -5, \dots, x_{10} = 5$:

Creare un programma che sostituisca alla funzione $f(x)$ la sua spline lineare nell'intervallo $[-5, 5]$, ovvero:

1. Legga dal file i dati creando un array.
2. Chieda all'utente un numero reale x
 - (a) Se il numero è esterno all'intervallo $[-5, 5]$ scriva un messaggio di errore e chieda un'altro numero.
 - (b) Se il numero è 100, termini l'esecuzione.
 - (c) Se il numero è interno all'intervallo $[-5, 5]$ calcoli il valore della spline lineare in x e chieda un'altro numero.

Si tenga presente che il programma dovrà determinare in quale intervallo ricade il numero x , calcolare la retta che fa parte della spline lineare in quell'intervallo e calcolare il valore della retta in x .

-2
0
1
1
2
0
0
-1
0
-2
1

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 10 gennaio 2007

Creare un programma che

1. Chieda all'utente l'input di due numeri reali a e p , (entrambi positivi e $p < a$ con eventuale controllo).
2. Tabuli la funzione $f(x) = e^{-x^2}$ nell'intervallo $[-a, a]$ con passo p , ovvero crei due array, una con le ascisse e una con le ordinate della funzione (è praticamente impossibile che l'array delle ascisse termini esattamente con a . Concluderlo con un numero minore di a).
3. Calcoli numericamente l'integrale $\int_{-a}^a f(x) dx$ mediante il metodo di Bézout dei trapezi sull'intervallo $[-a, a]$
4. Sarebbe bene che, dopo aver calcolato l'integrale, il programma non terminasse, ma ripartisse da capo chiedendo un altro a e un altro p , fintantoché l'utente non ponga $a = 0$, nel qual caso il programma terminerà salvando in un file i vari tentativi con diversi a , diversi passi e il valore dell'integrale ottenuto.

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 25 gennaio 2007

Creare un programma che

1. Legga un file di testo **poly.txt** creato precedentemente contenente una successione numerica

$$a_0, a_1, \dots, a_n$$

2. Calcoli quanti numeri ha letto e chiami n questo numero *diminuito di 1*.
3. Tabuli la funzione polinomiale $p(x) = a_0 + a_1x + \dots + a_nx^n$ di grado n nell'intervallo $[-2, 2]$ con passo 0.1, ovvero crei due array, una con le ascisse e una con le ordinate della funzione. (Tenere presente che probabilmente sarà meglio ribattezzare i coefficienti b_1, \dots, b_n per evitare i problemi dell'indice nullo delle array)

4. Il calcolo va fatto mediante l'algoritmo di Hörner.

$$p(x) = a_0 + x \left(a_1 + x \left(a_2 + x \left(a_3 + x \left(a_4 + \dots \right) \right) \right) \right)$$

5. Dopo aver calcolato l'integrale, il programma dovrebbe salvare in un file ben strutturato la tabulazione.

Ben strutturato vuol dire di questo tipo:

```
-2.0  3.0
-1.9  3.1
...  ...
2.0  5.6
```

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 9 febbraio 2007

La funzione di Collatz

L'algoritmo di Collatz è così fatto:

Si parte con un numero intero $n > 1$.

Se n è pari al passo successivo si prende il numero $n/2$.

Se invece n è dispari al passo successivo si prende $3n + 1$.

L'algoritmo termina sempre con il numero 1, anche se non è facile prevedere dopo quanti passi. È un fatto noto anche se non ancora dimostrato rigorosamente.

Creare un programma che

- Chieda all'utente un numero intero $n > 1$ (e controlli che sia intero e maggiore di 1.)
- Esegua l'algoritmo di Collatz, visualizzando i passi e il numero di passi, per esempio così:


```
1  20
2  10
3   5
4  16
5   8
6   4
7   2
8   1
```
- Salvi in un file di testo tutti i passi dell'algoritmo e il numero di passi necessari. Il nome del file dovrebbe comprendere il numero da cui si è partiti (per esempio "Coll135.txt")

Note:

1. Dato che FORTRAN arrotonda e non sempre $n/2$ è intero anche se n è perfettamente pari, per verificare se un numero n è intero si può per esempio calcolare la parte interna $\text{int}(n)$ e guardare se $n - \text{int}(n)$ è molto piccolo (per esempio minore di 10^{-5}).
2. Analogamente per vedere quando si arriva a 1, invece di controllare se il numero è proprio 1, è meglio verificare se la sua differenza con 1 è molto piccola.

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 16 marzo 2007

L'algoritmo di punto fisso

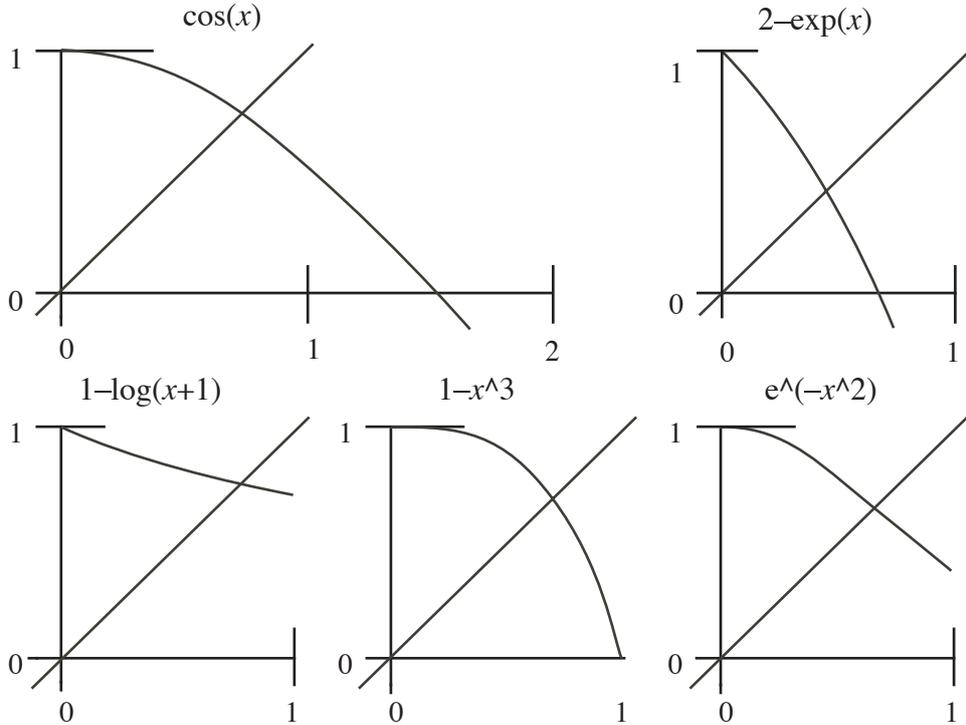
Sia $f(x)$ definita in $[0, 1]$.

Creare un programma che

- Chieda all'utente un numero reale $0 < t < 1$ (e controlli che sia nell'intervallo.)
- Chieda all'utente un numero reale $\varepsilon > 0$ (e controlli che sia positivo)
- Esegua l'algoritmo di punto fisso sulla funzione $f(x)$ partendo da t e arretandolo quando $\frac{|t(i) - t(i+1)|}{|t(i)|} < \varepsilon$, oppure quando rilevi che l'algoritmo non converge.
- Salvi in un file di testo tutti i passi dell'algoritmo e il numero di passi necessari.

Sperimentare il programma con le seguenti funzioni:

$$f(x) = \cos(x) \quad f(x) = 2 - e^x \quad f(x) = 1 - \ln(x+1) \quad f(x) = 1 - x^3 \quad f(x) = e^{-x^2}$$



Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 11 giugno 2007

Il metodo di Eulero

Creare un programma che

1. Chieda all'utente l'input di un numero reale y_0 compreso tra -1 e 2 e di un numero reale $h > 0$ (se lo si ritiene opportuno, si può inserire una routine di controllo).
2. Crei col metodo di Eulero la tabulazione della soluzione del problema di Cauchy

$$\begin{cases} y' = \frac{y^2 - t^2}{t^2 + 1} \\ y(0) = y_0 \end{cases}$$

nell'intervallo $[0, 4]$ con passo h ovvero crei due array x e f

$$x(1) = 0 \quad x(2) = h \quad x(3) = 2h \quad x(4) = 3h \quad \dots \quad x(??) = 4$$

$$f(1) = y_0 \quad f(2) = y(h) \quad (\text{dove } y \text{ è la soluzione approssimata}) \quad \dots \quad f(??) = ??$$

3. Salvi la tabulazione in un file di testo così strutturato:

$$\begin{array}{c|c} 0 & y(0) \\ h & y(h) \\ 2h & y(2h) \\ \dots & \dots \end{array}$$

Suggerimento 1: Forse l'array x non termina esattamente con 4.....

Suggerimento 2: Prestare attenzione agli indici che descrivono le due array e alla lunghezza delle due array.

Suggerimento 3: Non confondete l'indice delle array (che è un *numero intero*) con la variabile t che deve essere *reale* e non confondere l'array f con la funzione y soluzione approssimata del problema.

Suggerimento 4: Usare anche carta e penna prima di scrivere il listato sul computer.....

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 27 giugno 2007

Le Bézier quadratiche

Inizialmente creare un file di testo contenente i seguenti sei numeri

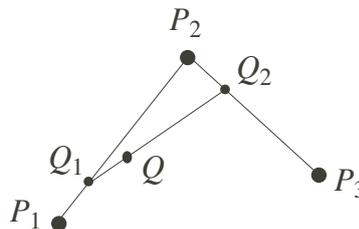
$$-1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0$$

Indi creare un programma che

1. Legga dal file i sei numeri e li interpreti come i tre punti $P_1(-1, 0)$ $P_2(0, 1)$ $P_3(1, 0)$
2. Chieda all'utente se vuole cambiare i tre punti. In tal caso li richieda e li salvi nel file di testo (Opzionale: chiedere se si vuole cambiare uno solo dei tre e quale).
3. Tabuli la parabola di Bézier avente P_1, P_2, P_3 come poligono di controllo, ovvero scriva un po' di punti della parabola.

Usare l'algoritmo di deCasteljau determinando quindi per ogni $t \in [0, 1]$ un punto Q_1 su P_1P_2 e uno Q_2 su P_2P_3 e quindi il punto finale Q su Q_1Q_2 .

Occorre scegliere un passo (eventualmente con richiesta all'utente); p.es. con passo 0.1 si devono ottenere 11 punti.



4. Salvi la tabulazione in un secondo file di testo così strutturato:

$$\begin{array}{c|c} x_1 & y_1 \\ x_2 & y_2 \\ \cdots & \cdots \\ x_n & y_n \end{array}$$

Ovviamente (x_1, y_1) sarà P_1 e (x_n, y_n) sarà P_3 .

Suggerimento 1: Usare un array per le ascisse e uno per le ordinate.

Suggerimento 2: Prestare attenzione agli indici che descrivono le due array e alla lunghezza delle due array.

Suggerimento 3: Non confondete l'indice delle array (che è un *numero intero*) con la variabile t che deve essere *reale*.

Suggerimento 4: Usare anche carta e penna prima di scrivere il listato sul computer.....

Esame di Metodi numerici per l'ingegneria navale

Prova al calcolatore 9 luglio 2007

Ricerca zeri mediante algoritmo di bisezione

Scegliere una funzione di variabile reale $f(x)$ e un intervallo $[a, b]$ in cui sia definita.

Creare un programma che

1. Tabuli la funzione con un passo largo (diciamo 0.1)
2. Esamini i valori della funzione per vedere se ci sono cambiamenti di segno tra due passi successivi e scriva sul display una frase del tipo "La funzione ha ... zeri nell'intervallo ..." (Opzionalmente elenchi la collocazione approssimativa degli zeri), oppure "La funzione non ha zeri ..."
3. Se esiste almeno uno zero chiedi all'utente un numero piccolo ε .
4. Per ognuno dei cambiamenti di segno determini lo zero della funzione ivi compreso mediante l'algoritmo di bisezione con un errore inferiore a ε .
5. Salvi la tabulazione e gli zeri in un file di testo così strutturato:

$$\begin{array}{c} \text{La funzione è } x^2 + \dots \\ \begin{array}{c|c} a & f(a) \\ a + 0.1 & f(a + 0.1) \\ \cdots & \cdots \\ b & f(b) \end{array} \\ \text{Gli zeri della funzione sono} \\ x_1 = \dots \\ x_2 = \dots \\ \dots \end{array}$$

Suggerimento 1: Usare un array per le ascisse e uno per le ordinate.

Suggerimento 2: Forse l'array x non termina esattamente con b

Suggerimento 3: Prestare attenzione agli indici che descrivono le due array e alla lunghezza delle due array.

Suggerimento 4: Non confondete l'indice delle array (che è un *numero intero*) con la variabile x che deve essere *reale*.

Suggerimento 5: È bene che l'algoritmo di bisezione sia una subroutine, dato che è possibile che venga usato parecchie volte.

Suggerimento 6: Usare anche carta e penna prima di scrivere il listato sul computer.....

Sperimentare il programma con le seguenti funzioni nell'intervallo $[-3, 3]$ o con altre funzioni e altri intervalli che si ritengono interessanti:

$$\arctan(x + 1) - x \quad x^2 + 3x^2 - 1 \quad x^4 - 3x^2 + x + 1 \quad x^4 + x + 2$$

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 20 luglio 2007

Algoritmo elementare di sorting

- Preparare alcuni files (almeno un paio) che elenchino almeno 10 numeri reali casuali tipo

12
1.34
-56
...

- Creare un programma che
 1. Chieda all'utente il nome del file
 2. Legga i numeri dal file
 3. Ponga i numeri in ordine di grandezza mediante un algoritmo di sorting, ma memorizzi l'ordine originale.
 4. Salvi i numeri ordinati in un file strutturato all'incirca in questo modo:

1^ numero:	-56	era al posto 3
2^ numero:	-2	era al posto 9
3^ numero:	1.34	era al posto 2
	...	

5. Opzionalmente il programma potrebbe contare gli scambi effettuati.
6. Opzionalmente il programma potrebbe consentire di effettuare sorting di svariati files senza dover rilanciare il programma e dare al file di output un nome attinente al nome del file su cui si è effettuato il sorting (Per esempio se il file era: "Elenco.txt" il file di output potrebbe chiamarsi "Elenco_ord.txt")

Suggerimento 1: Usare un array per i numeri e uno per l'ordine originale.

Suggerimento 2: Nella cartella X:/pub on aula/materiale dei corsi/odetti ci sono già quattro file pronti di numeri casuali da copiare nella vostra cartella

Suggerimento 3: L'algoritmo di sorting più elementare (anche se sicuramente il meno efficiente) esamina tutte le coppie di numeri consecutivi e scambia la coppia qualora sia necessario. L'esame prosegue fintantoché non si arrivi a un esame senza scambi. Per dare un termine all'algoritmo occorre una variabile tipo flag che viene azzerata prima dell'esame di tutti i numeri e viene settata qualora ci sia uno scambio.

Suggerimento 4: Un algoritmo di sorting più efficiente, ma più complesso a programrarsi funziona così:

Sia L la lunghezza dell'array.

Si esaminano non le coppie di numeri consecutivi, ma le coppie di numeri che distino $L/2$ e si scambiano le coppie qualora sia necessario (ci sono $L/2$ test da effettuare). L'esame prosegue esaminando le coppie di numeri che distino $L/4$ fintantoché non si arrivi all'esame di coppie consecutive. Il numero di test da effettuare è molto inferiore a quello del sorting elementare.

Suggerimento 5: Usare anche carta e penna prima di scrivere il listato sul computer.....

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 19 settembre 2007

Algoritmo elementare di statistica

Nella cartella X:/pub on aula/materiale dei corsi/odetti ci sono quattro file di nome "Voti1.txt", "Voti2.txt", "Voti3.txt", "Voti4.txt" che contengono numeri da 0 a 30 e sono l'elenco dei voti di alcuni compiti di un corso degli anni passati.

- Creare un programma che
 1. Chieda all'utente quale file deve leggere.
 2. Legga i numeri dal file.
 3. Esegua la statistica dei voti.
 4. Scriva la statistica su schermo e la salvi in un file.

La statistica deve essere strutturata all'incirca in questo modo:

Compiti consegnati: 82

Voto 0	ottenuto da:	2	studenti
		...	
Voto 17	ottenuto da:	7	studenti
Voto 18	ottenuto da:	12	studenti
Voto 19	ottenuto da:	9	studenti
		...	
Voto 30	ottenuto da:	0	studenti

Opzionalmente il programma potrebbe rilevare qual è il voto (i voti) ottenuto da un numero maggiore di studenti.

Opzionalmente il programma potrebbe effettuare una sorta di grafica rudimentale così:

		...	
Voto 17	ottenuto da:	7	studenti ooooooo
Voto 18	ottenuto da:	12	studenti ooooooooooooo
Voto 19	ottenuto da:	9	studenti ooooooooooo
Voto 20	ottenuto da:	8	studenti oooooooooo
		...	

Opzionalmente il programma potrebbe consentire di effettuare statistiche di diversi files senza dover rilanciare il programma e dare al file di output un nome attinente al nome del file su cui si è effettuato il sorting (Per esempio se il file era: "Voti1.txt" il file di output potrebbe chiamarsi "StatisticaVoti1.txt")

Esame di Metodi numerici per l'ingegneria navale
Prova al calcolatore 16 novembre 2007

Ricerca di crescita, decrescenza e convessità

Nella cartella X:/pub on aula/materiale dei corsi/odetti/MNIN/ ci sono quattro file di nome "funzione1.txt", "funzione2.txt", "funzione3.txt", "funzione4.txt" che contengono un elenco di numeri reali. Altri files simili possono essere creati.

I files vanno interpretati come la tabulazione di una funzione a partire da 0 con passo 0.1

- Creare un programma che
 1. Chieda all'utente quale file deve leggere.
 2. Legga i numeri dal file e determini il dominio della funzione.

3. Dica in quali intervalli del dominio la funzione è crescente o decrescente.
4. Scriva l'analisi della crescita su schermo e la salvi in un file.

L'analisi della crescita deve essere strutturata all'incirca in questo modo:

Intervallo: [0, 5.1]	
$f(x)$ crescente in	[0 , 1.2]
$f(x)$ decrescente in	[1.2, 1.6]
$f(x)$ crescente in	[1.6, 2.8]
	...
$f(x)$ crescente in	[4.6, 5.1]

Il programma può poi calcolare la derivata discreta in ciascun intervallo di ampiezza 0.1 con la formula:

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{0.1}$$

e quindi verificare convessità e concavità di $f(x)$ usando la stessa routine.

Opzionalmente il programma potrebbe consentire di effettuare analisi di diverse funzioni senza dover rilanciare il programma e dare al file di output un nome attinente al nome della funzione (Per esempio se il file era: "funzione1.txt" il file di output potrebbe chiamarsi "Analisi_funzione1.txt")