

Calcolo Numerico (Ingegneria Energia/Meccanica, Canale B)
Prova di Laboratorio IV, del 24 Gennaio 2024

Cognome/nome _____ Matricola _____ MEC - ENE

Postazione _____

FIRMA PER CONSEGNARE _____

FIRMA PER RITIRARSI _____

1. SI RACCOMANDA AGLI STUDENTI DI **commentare adeguatamente** SCRIPT E FUNCTION MATLAB.
2. OGNI PORZIONE DI CODICE **deve avere** QUALE PRIMA RIGA UN COMMENTO MEDIANTE % CON NOME, COGNOME, NUMERO DI MATRICOLA E POSTAZIONE.

1. Si implementi mediante la routine `divisore` che calcoli tutti i divisori di un numero naturale non nullo.

A tal proposito,

- Si crei una function di nome `divisore` che implementi tale formula, avente come input:
 - (a) il numero `n`;
- la stessa function deve dare in output
 - (a) un vettore `divs` dei suoi divisori;
 - (b) una variabile `flag` con valori 1 se il processo é terminato correttamente, 0 altrimenti;
- la function deve avere la seguente intestazione:
`function [divs, flag] = divisore (n)`

Relativamente alla routine:

- si assegni a `flag` il valore 1;
- se `n` ún numero negativo o nullo, si scriva su monitor
`il numero scelto non e' positivo`
si assegni a `divs` il vettore privo di componenti a `flag` il valore 0 e si esca immediatamente dalla routine;
- se `n-floor(n)` non é nullo, e quindi il numero reale positivo non é intero, si scriva su monitor
`il numero scelto non e' naturale`
si assegni a `divs` il vettore privo di componenti a `flag` il valore 0 e si esca immediatamente dalla routine;
- si assegni a `divs` il vettore con unica componente uguale a 1;
- si scriva un ciclo for con indici `k` da 2 alla parte intera verso il basso di $n/2$ (ricordando che il numero é positivo si usa `floor` o `ceil`?) al cui interno:
 - (a) se `rem(n,k)` é nullo allora si ponga `divs(end+1)` uguale a `k`;
- eseguito il ciclo-for, quindi all'esterno di esso, se $n > 1$, si ponga `divs(end+1)` uguale a `n`;

2. Si scriva una function `demo_divisore` in cui

- si assegni a `v` il vettore con componenti equispaziate da 1 a 11 con spaziatura uguale a 2;
- si ponga `w` il vettore ottenuto eseguendo componente per componente la potenza quarta di `v` (ovvero se $v = (v_1, v_2, \dots, v_m)$ allora $w = (v_1^4, v_2^4, \dots, v_m^4)$);
- si ponga `L` pari alla lunghezza del vettore `w`;

(continua →)

- definisca un ciclo-for in cui l'indice di iterazione k varia da 1 a L e alla k -sima iterazione di tale ciclo,
 - (a) si ponga n uguale a alla k -sima componente di w
 - (b) si applichi la routine `divisore` con input n e output `divs` e `flag`;
 - (c) si ponga `numero_divisori` pari alla lunghezza di `divs`;
 - (d) si stampino su monitor i valori assunti da n e `numero_divisori`, con 5 cifre prima della virgola, nessuna dopo la virgola, in formato decimale;
 - (e) si vada a capo di una riga su monitor (mediante un certo utilizzo di `fprintf`);
 - (f) si scrivano su monitor i divisori calcolati mediante il comando `disp(divs')` (attenzione al simbolo di trasposizione!);
- nella tabella che segue si scrivano i valori stampati su monitor dell'ultimo valore di n , del corrispettivo `numero_divisori` e dei valori di quest'ultimi

<code>n</code>	
<code>numero_divisori</code>	
<code>divisori</code>	