Calcolo Numerico
TEST del 11 SETTEMBRE 2017

Cognome e nome ____ **Matricola** ____________
Informatica

Postazione ____________________

FIRMA PER CONSEGNARE ____

FIRMA PER RITIRARSI ____

Si raccomanda agli studenti di **commentare adeguatamente** script e function Matlab.

- Creare una function di nome `trapezi_composta.m` che implementi l'algoritmo relativo alla formula composta dei trapezi. La function deve avere come parametri in ingresso la funzione integranda `f`, gli estremi dell'intervallo di integrazione $[a, b]$ ed il numero m di suddivisioni dell'intervallo di integrazione. I parametri in uscita devono essere l'approssimazione dell'integrale ottenuta con il metodo ed il passo h di integrazione. La function avrà quindi la seguente intestazione:

```
function [int,h] = trapezi_composta (f,a,b,m);
%TRAPEZI Formula dei trapezi composta
%
% [int,h] = trapezi_composta (f,a,b,m);
%
% Dati di ingresso:
% f: funzione integranda
% a: estremo sinistro dell'intervallo di integrazione
% b: estremo destro dell'intervallo di integrazione
% m: numero di sottointervalli
% Dati di uscita:
%  int:    approssimazione dell'integrale definito
%  h:      passo di integrazione
```

- Si implementi uno script `trapezi_adattativa` in cui si assegnino come input
  - una funzione `f`,
  - gli estremi dell'intervallo di integrazione `a`, `b`,
  - la tolleranza `toll`,

  e come output
  - il vettore `I` composto da approssimazioni successive dell'integrale richiesto,
  - valore di riferimento Q della soluzione esatta tramite la funzione Matlab `quadl`, con una tolleranza `TOL`$=10^{(-15)}$.
  - il valore binario `flag`.

  Le approssimazioni successive $I(n)$ dell'integrale $\int_a^b f(x)dx$ per $n = 1, 2, ...$, saranno ottenute utilizzando `trapezi_composta`, raddoppiando il numero di sottointervalli precedente. Piu' precisamente, $I(1)$ sará ottenuto per $m = 1$, $I(2)$ per $m = 2$ e in generale $I(n)$ per $m = 2^{n-1}$.

  Si imponga `nmax=100` come massimo valore accettabile di n, ovvero l'ultimo valore che il codice puó eventualmente calcolare é $I(\text{nmax})$. Le iterazioni dovranno essere arrestate quando per $n = n^*$ la quantitá $E_n = |I_{n+1} - I_n| <$ `toll`.

  Si ponga `flag`=1 se il codice ha calcolato l'integrale richiesto con $E_n <$ `toll` in al piú `nmax` iterazioni, `flag`=0 altrimenti.

- Si implementi uno script `esempio` in cui si utilizzi `trapezi_adattativa` assegnando come input
  - la funzione $f(x) = x^{11/2}$,
  - gli estremi dell'intervallo di integrazione $a = 0$, $b = 1$,
  - `toll`$=10^{-8}$ quale tolleranza.

  e come output
  - il vettore `I` composto da approssimazioni successive dell'integrale richiesto,
  - il valore di riferimento Q della soluzione esatta tramite la funzione Matlab `quadl`, con una tolleranza `TOL`$=10^{(-15)}$.

  Quando i risultati ottenuti sono ritenuti corretti, `esempio` produca una figura che contenga in scala semi-logaritmica le coppie (n, E_n^*) per $n = 1, \ldots, n^*$, essendo $E_n^* = |I_n - Q|/|Q|$, $n = 1, \ldots, n^*$ la successione degli errori relativi ottenuta considerando le approssimazioni successive I_n ed il valore di riferimento Q calcolato tramite `trapezi_adattativa`.

  Si salvi la figura ottenuta in `myplot.jpg`.

  Infine, lo script `esempio` scriva in una tabella `tabella.txt` le coppie $(n, I(n))$ per $n = 1, \ldots, n^*$. I valori di `I(n)` siano descritti in notazione esponenziale, con 1 cifra *prima* della virgola e 15 *dopo* la virgola.