Laboratorio Calcolo Numerico

Esercizio 1 Si vuole determinare un polinomio interpolatore della funzione

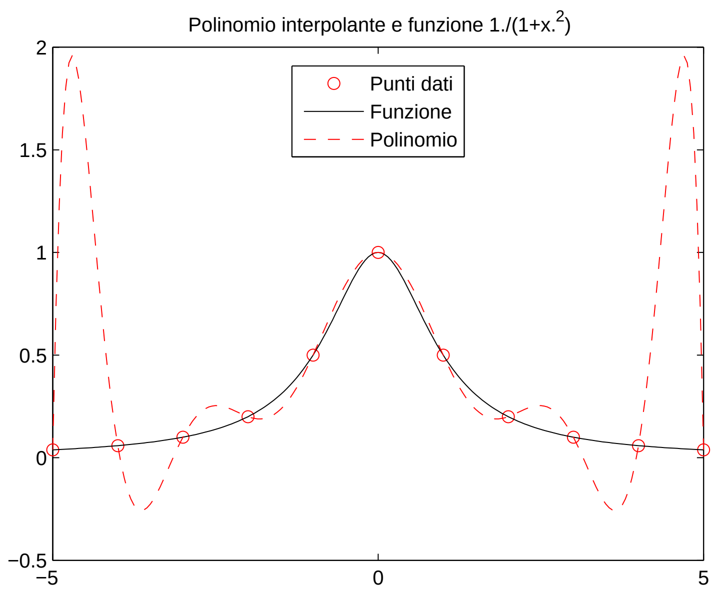
$$f(x) = \frac{1}{1+x^2}$$

nell'intervallo $I = [-5, 5]$. Considerata la forma di Newton del polinomio interpolante, si costruisca una tabulazione della funzione data costituita da 11 nodi equidistanti nell'intervallo I, e si memorizzino i nodi in una vettore `xn` ed i corrispondenti valori della funzione nel vettore `fxn`.

Si scriva una funzione di nome `polnewtondf` che abbia la seguente intestazione e che calcoli i coefficienti della base di Newton con le **differenze finite in avanti**.

```
function c = polnewtondf (x,y)
% POLNEWTONDF Calcola i coefficienti del polinomio interpolatore
%             utilizzando la forma di Newton con le differenze
%             finite
%
% Uso:
%   c = polnewtondf (x,y)
%
% Dati di ingresso:
%   x     vettore dei nodi
%   y     vettore dei valori della funzione da interpolare nei nodi
%
% Dati di uscita:
%   c     vettore colonna dei coefficienti ordinati per indici
%         crescenti (c_0, c_1, ... )
```

Ovviamente i coefficienti restituiti dovranno coincidere con quelli ottenuti con la function `polnewton` sviluppata nel laboratorio precedente, ed anche la figura definita nello script (usare anche la function `horner` per il calcolo del valore del polinomio nei 201 valori di x) risulterà essere la stessa:



Il passo `h` dovrà essere calcolato all'interno della function utilizazando i nodi memorizzati nel vettore `x`.

All'interno della function, dovranno essere effettuati preventivamente i seguenti controlli:

- la lunghezza dei vettori `x` ed `y` deve essere la stessa;
- i nodi devono essere memorizzati in ordine crescente nel vettore `x` (suggerimento, funzione Matlab `sort` da usare per il controllo);
- i nodi devono essere equispaziati.

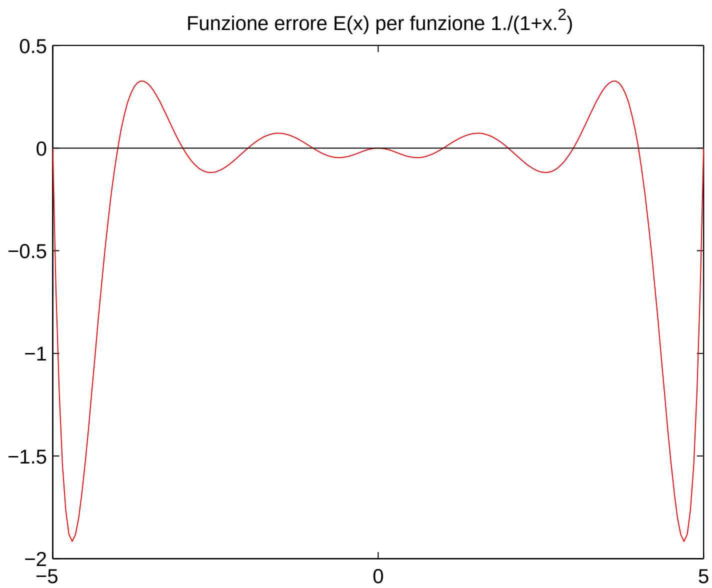
In caso di violazione, si esca con un `error`.

Esercizio 2

Si produca nello script anche una figura che rappresenti nello stesso intervallo l'asse x e la funzione errore definita come

$$E(x) = f(x) - P(x).$$

ottenendo la seguente figura



Esercizio 3

Risultati analoghi a quelli degli esercizi precedenti si possono ottenere con due funzioni Matlab:

```
POLYFIT Fit polynomial to data.
   P = POLYFIT(X,Y,N) finds the coefficients of a polynomial P(X) of
   degree N that fits the data Y best in a least-squares sense.
   P is a row vector of length N+1 containing the polynomial coefficients in
   descending powers, P(1)*X^N + P(2)*X^(N-1) +...+ P(N)*X + P(N+1).
```

che permette di determinare un vettore che contiene i coefficienti del polinomio interpolante rispetto alla base canonica;

```
POLYVAL Evaluate polynomial.
   Y = POLYVAL(P,X) returns the value of a polynomial P evaluated at X. P
   is a vector of length N+1 whose elements are the coefficients of the
   polynomial in descending powers.
       Y = P(1)*X^N + P(2)*X^(N-1) + ... + P(N)*X + P(N+1)
```

che valuta un polinomio di cui si conoscono i coefficienti rispetto alla base canonica su tutti i valori precedentemente memorizzati su di un vettore.

Si scriva un altro script che produca le stesse figure precedentemente richieste (che risulteranno praticamente uguali) e poi si cerchi di strutturarle con il comando `subplot` come segue