

Calcolo della radice quadrata

Ci sono diversi algoritmi iterativi per il calcolo della radice quadrata di un numero positivo **a**. In questa esercitazione implementeremo due di questi metodi.

Parte I: Metodo Babilonese

La radice quadrata \sqrt{a} di un numero positivo **a** si può calcolare con un metodo iterativo come il limite della successione

$$r_{n+1} = \frac{1}{2} \left(r_n + \frac{a}{r_n} \right)$$

Il valore del primo termine r_0 è influente e può essere scelto a piacere. Infatti potete verificare che cambiando tale valore iniziale il risultato non cambia! Il ciclo termina quando la differenza $|r_{n+1} - r_n|$ tra due termini successivi è minore della precisione ϵ desiderata.

Scrivere un programma **esercitazione04.c** per eseguire i seguenti passi

1. Scrivere sullo schermo un messaggio informativo riguardo il programma
2. Acquisire dall'utente il valore della precisione **epsilon** compresa tra **[1.E-6, 0.1]** e verificarne la correttezza. In caso di errore ripetere l'operazione.
3. Acquisire dall'utente il valore del termine **r0**
4. Acquisire dall'utente il valore di **a** e verificare che sia positivo. In caso di errore ripetere l'operazione.
5. Implementare il metodo di babilonese utilizzando un opportuno ciclo.
 1. in ciascuna iterazione scrivere sullo schermo il numero di iterazioni effettuate e la stima attuale del risultato con il formato **%4.20lf**
6. Al termine del ciclo, scrivere sullo schermo il numero di iterazioni **Ntot** eseguite
7. Scrivere sullo schermo la differenza tra il valore ottenuto da voi e **sqrt(a)** utilizzando il formato **%2.2E** per scrivere il risultato con la notazione scientifica per apprezzare anche le differenze molto piccole
8. Potete eseguire il programma per lo stesso valore di **a** ma variando la precisione **epsilon** per vedere come cambia **Ntot** al variare di **epsilon**.

Parte II: Metodo Alternativo

L'inverso $1/\sqrt{a}$ della radice quadrata di un numero positivo a si ottiene come il limite della successione

$$y_{n+1} = \frac{3}{2}y_n - \frac{a}{2}y_n^3$$

Anche se la scelta del termine y_0 è arbitraria, tuttavia il metodo è più sensibile al valore scelto e può facilmente divergere (provare per credere!). Occorre quindi avere una stima iniziale di

$y_0 \sim 1/\sqrt{a}$ con il metodo babilonese. Scrivere un programma

esercitazione04bis.c :

1. Acquisire dall'utente il valore della precisione **epsilon** compresa tra **[1.E-6, 0.1]** e verificarne la correttezza. In caso di errore ripetere l'operazione.
2. Acquisire dall'utente il valore di **a** e verificare che sia positivo. In caso di errore ripetere l'operazione.
3. Utilizzando un valore a piacere di **r0**, fare 2 iterazioni con il metodo babilonese per calcolare **r2**
4. Implementare il nuovo metodo iterativo con la successione **yn** con un opportuno ciclo che utilizzi come valore iniziale **y0 = 1/r2**
5. Scrivere sullo schermo il numero di iterazioni **Ntot** eseguite prima di interrompere il ciclo quando **| 1/yn+1 - 1/yn | < epsilon**
6. Scrivere sullo schermo la differenza tra il valore di **1/y** e **sqrt(a)** utilizzando la notazione scientifica.

Suggerimenti

- Utilizzare valori noti di **a** per verificare la correttezza dell'algoritmo
- Commentare brevemente i passi più importanti nel codice utilizzando i commenti del linguaggio C con la sintassi **/* commento */**
- Includere gli opportuni header file necessari per la compilazione e correggere i **warning**
- Per compilare e creare l'eseguibile dovete fare

```
gcc -Wall -o app.exe programma.c -lm
```