Test del generatore di numeri casuali

La funzione di libreria *rand()* genera un numero casuale compreso tra 0 e RAND_MAX.
Tale funzione permette di generare una sequenza di numeri pseudocasuali. Tali numeri si definiscono pseudocasuali perche', fissato il primo valore della sequenza (chiamato seme), e' fissata tutta la sequenza di valori che saranno generati nelle successive invocazioni della funzione *rand()*.
Infatti i numeri sono generati mediante una certa funzione *f(x)* che, dato l'ultimo valore *x* generato, genera il prossimo valore random.
Facciamo un esempio: supponiamo di aver dato come seme il valore 1. Allora il primo valore generato dalla funzione *rand()* sara' 41 (provare per credere). La prossima volta che verra' invocata la funzione *rand()* si otterra' il valore *f*(41), che e' pari a 18647. Alla successiva invocazione si otterra' il valore di *f*(18647), che e' pari a 6334, e cosi' via.
Pertanto, per ottenere sequenze diverse, bisogna cambiare il valore del seme. Per cambiare il valore del seme, si utilizza la funzione *srand(n)*, ove *n* e' il nuovo valore che si vuol dare al seme.
Per utilizzare le funzioni *rand()* ed *srand(n)*, bisogna includere il file di intestazione "stdlib.h".
Scrivere un programma per testare il generatore di numeri casuali. Il programma deve invocare la funzione *rand()* per un elevato numero di volte, quindi deve stampare i seguenti valori:

- Valore minimo generato
- Valore massimo generato
- Valore medio generato, pari alla somma di tutti i valori generati, diviso il numero di valori generati
- Scostamento tra il valore medio generato e quello atteso (qual e' il valor medio che ci attendiamo dalla generazione di numeri casuali tra 0 e RAND_MAX?)
- Scostamento percentuale tra il valore medio generato e quello atteso, numericamente uguale a: (scostamento/valore medio atteso)*100

La soluzione e' nel file test_rand.cpp.

Problemi e suggerimenti.

- La somma di un elevato numero di valori puo' dar luogo ad overflow. Quindi, sommare tutti i valori e dividere poi per il numero di valori generati portera', quasi sicuramente, ad un risultato errato. Provate a pensare ad una soluzione alternativa. Per farlo tenete conto che RAND_MAX e' un numero piccolo rispetto all'ampiezza dell'intervallo di rappresentazione dei numero interi. Pertanto la somma di due numeri generati dalla funzione *rand()* non da mai luogo ad overflow. Inoltre la media di due numeri e' sempre minore del numero piu' grande tra i due. Quindi, se si fa la media tra due numeri generati dalla funzione *rand()*, si ottiene un numero che puo' essere sommato senza problemi alla media calcolata su altri due numeri generati dalla funzione *rand()*. Se questo piccolo suggerimento vi basta e riuscite da soli a trovare una soluzione, e' meglio, altrimenti potete tenere conto dei prossimi suggerimenti. Il primo e' relativo ad una possibile implementazione ricorsiva, il secondo e' relativo ad una possibile implementazione iterativa.
- Possibile approccio ricorsivo (se conoscete la ricorsione). Sia val_i l'i-esimo valore generato dalla funzione rand(), allora, se il numero N di valori generati e' pari, vale la seguente proprieta':

$$Valore\_medio = \frac{\sum_{i=1}^{N} val_i}{N} = \frac{\sum_{i=1}^{N/2} val_i + \sum_{i=\frac{N}{2}+1}^{N} val_i}{2\frac{N}{2}} = \frac{\frac{\sum_{i=1}^{N/2} val_i + \sum_{i=\frac{N}{2}+1}^{N} val_i}{\frac{N}{2}}}{2} = \frac{\frac{\sum_{i=1}^{N/2} val_i}{\frac{N}{2}} + \frac{\sum_{i=\frac{N}{2}+1}^{N} val_i}{\frac{N}{2}}}{2}$$

Ossia, per calcolare la media della sequenza iniziale di N numeri, posso calcolare la media parziale della prima sotto-sequenza di N/2 numeri e la media parziale della successiva sotto-sequenza di N/2 numeri, quindi posso fare la media tra le due medie parziali. Se anche N/2 e' un numero pari, allora posso applicare ricorsivamente lo stesso ragionamento per calcolare ciascuna delle suddette medie parziali (ossia posso suddividere ciascuna delle due sotto-sequenze in due ulteriori sotto-sequenze) e cosi' via. Ovviamente la ricorsione si ferma ogni volta che arrivo ad una sequenza di soli due valori. Affinché ciascuna sotto-sequenza si possa ulteriormente dividere in due parti uguali, che caratteristica deve avere il valore iniziale di N? (la risposta è nell'ultimo suggerimento)

- Sfruttando la stessa proprietà vista nel precedente suggerimento, si può implementare una soluzione iterativa. In questo caso però, anziché suddividere inizialmente la sequenza in due parti, conviene dividere subito la sequenza in N/2 coppie consecutive e calcolare la media per ciascuna coppia. Quindi si divide la sequenza di valori medi risultanti in N/4 coppie consecutive e si calcola la media per ciascuna coppia. Si continua così fino a ridursi ad una sola coppia e, quindi, si calcola il valor medio finale come media tra i due valori di tale coppia. Che caratteristica deve avere N, affinché, ad ogni passo iterativo, si possa sempre generare un numero pari di coppie di valori (su cui lavorare per calcolare le nuove medie parziali al prossimo passo iterativo)?
- Il valore di N, sia nel caso di soluzione ricorsiva, che nel caso di soluzione iterativa, deve essere una potenza intera di 2

Estensione (se si conosce la ricorsione): All'interno del programma, calcolate il valor medio sia in modo ricorsivo, che in modo iterativo. Quindi confrontate i due risultati. Stabilite una soglia massima di tolleranza della differenza tra i due risultati. Se si supera la soglia, il programma comunica un errore, altrimenti stampa entrambi i valori medi (quello ottenuto col metodo ricorsivo e quello ottenuto col metodo iterativo), e lo scostamento tra i due valori (sia in termini assoluti che in percentuale rispetto ad uno dei due valori medi calcolati). Tale scostamento dovrebbe essere molto ridotto.