

# Corso di Laurea in Fisica

## Esame di Laboratorio II – I Modulo

23 settembre 2021

1

### **Abstract**

Si risolva il seguente esercizio, scrivendo un programma in C++. Ai fini della valutazione, il primo criterio che deve essere soddisfatto è che il codice compili senza errori ed esegua realizzando le funzionalità richieste dal testo. Per la valutazione sarà inoltre tenuto in considerazione il fatto che i codici siano scritti con ordine, utilizzando opportunamente l'**indentazione** e i **commenti**. Si richiede infine di iniziare i codici con una riga di commento contenente il comando necessario per creare l'eseguibile.

## Il teorema centrale del limite

1. si scriva una libreria (definita ed implementata nei file `random.h` e `random.cc` rispettivamente) che generi numeri pseudo-casuali secondo una distribuzione di densità di probabilità Gaussiana utilizzando il teorema centrale del limite, partendo dalla generazione di numeri pseudo-casuali secondo una distribuzione uniforme. Si implementi il seguente prototipo:  
`double rand_TCL_gaus (double mean, double sigma, int N),`  
dove `mean` è il valor vero della media del campione, `sigma` il valor vero della sua deviazione standard e `N` il numero di numeri pseudo-casuali uniformi sommati;
2. si scriva un programma `test.cpp` che verifichi il comportamento della funzione `rand_TCL_gaus` generando con essa 1000 numeri pseudo-casuali;
3. si aggiunga a `test.cpp` il calcolo non distorto di media e varianza della distribuzione ottenuta, verificando che corrispondano ai valori veri utilizzati come parametri di `rand_TCL_gaus`;
4. si aggiunga a `test.cpp` il riempimento di un `TH1F` con il campione di 1000 eventi;
5. si esegua un fit gaussiano della distribuzione dell'istogramma e si confrontino i valori della media e sigma risultanti dal fit ai valori veri utilizzati come parametri di `rand_TCL_gaus`;
6. si scriva un programma `main.cpp` che generi, in un ciclo, diversi campioni di 1000 numeri pseudo-casuali con la funzione `rand_TCL_gaus` al crescere del parametro `N` (in scala logaritmica);
7. all'interno del ciclo, si raccolga ogni campione generato in un `std::vector`;
8. si scriva una funzione che calcoli la curtosi di ciascun campione implementando il seguente prototipo:  
`double calcola_curtosi (vector<double> & campione)`  
e si calcoli la curtosi per ciascun campione con questa funzione;
9. si disegni in un `TGraph` l'andamento della curtosi in funzione del parametro `N`.

Gli studenti affetti da disturbi specifici dell'apprendimento (DSA) potranno tralasciare i punti numero 5 e 9. Questi ultimi dovranno anche consegnare, oltre allo svolgimento del tema, una copia del proprio Progetto Universitario Individualizzato (P.Uo.I).