

Laboratorio di Calcolo per Fisici, Esercitazione valutata del 21/12/2023, A.A. 2023/2024

Nome _____ Cognome _____

Matricola _____ Ritirato/a

Lo Scopo di questa prova d'esame è di scrivere un programma in C e uno script in python seguendo la traccia riportata di seguito. Si tenga presente che:

1. Per svolgere il compito si hanno a disposizione 3 ore.
2. Si possono usare libri di testo, prontuari e gli appunti ma non è ammesso parlare con nessuno né utilizzare cellulari, tablet o laptop, pena l'annullamento del compito.
3. Il programma va scritto e salvato esclusivamente sul computer del laboratorio, a cui si deve accedere utilizzando come username **studente** e come password **informatica**
4. **Tutti i file vanno salvati in una cartella chiamata ELCDIC_NOME_COGNOME nella home directory**, dove NOME e COGNOME indicano rispettivamente il tuo nome e cognome. Ad esempio lo studente *Marco Rossi* deve creare una cartella chiamata ELCDIC_MARCO_ROSSI contenente tutti i file specificati nel testo. **Tutto ciò che non si trova all'interno della cartella suddetta non verrà valutato.** In tutti i programmi e script inserisci all'inizio un commento con il tuo nome, cognome e numero di matricola.
5. **Dovete consegnare il presente testo indicando nome, cognome e numero di matricola** (vedi sopra), barrando la casella "Ritirato/a" se ci si vuole ritirare, ovvero se non si vuole che la presente prova venga valutata.

Gino e Mario decidono di fare il seguente gioco: Gino sceglie un numero N compreso tra 10 e 20 e Mario estrae casualmente dei numeri compresi tra 1 e N . Mario comincia la partita con $P = N$ punti, e dopo ogni estrazione controlla se k , il numero appena estratto, sia già uscito: se sì perde k punti, altrimenti li guadagna. Mario continua a estrarre numeri finché $P \leq 0$ o $P \geq 4N$. Nel primo caso ha vinto Gino, nel secondo ha vinto Mario.

► **Prima parte:** Si scriva un programma chiamato `nome_cognome.c` (tutto minuscolo, senza eventuali spazi, accenti o apostrofi) che simuli un numero di giocate compreso fra $PMIN = 1$ e $PMAX = 10000$ tra Gino e Mario. In particolare il programma dovrà:

1. Chiedere inizialmente il numero di giocate da effettuare, reiterando la richiesta se il numero immesso è minore di $PMIN$ o maggiore di $PMAX$
2. Simulare le partite, estraendo casualmente, per ogni partita, un numero intero $N \in [10, 20]$ e poi, finché la partita non termina, estrarre numeri casuali interi k_i .
3. Stampare su schermo, alla fine di ogni partita, il vincitore della partita. Quando si è certi del funzionamento del programma, commentare questa istruzione.
4. Riempire un array `passi[100]` con le occorrenze dei passi necessari per terminare la partita. In altre parole, al termine di ogni partita se Mario ha dovuto estrarre n numeri per vincere o perdere, l'elemento `passi[n]` deve essere incrementato di 1.
5. Stampare su schermo un resoconto delle partite: numero di partite, numero di vittorie di Gino, numero di vittorie di Mario.
6. Salvare su un file chiamato `passi.dat` le frequenze relative ai passi necessari per terminare la partita, eseguendo il programma con il numero massimo di partite. La frequenza relativa al numero di passi è pari al numero di occorrenze, cioè il numero di partite terminate dopo tale numero di passi, diviso il numero di partite totali. Ogni riga del file dovrà contenere due numeri: il numero di passi n e la corrispondente frequenza, espressa con 4 cifre decimali.

Nello scrivere il programma si richiede che vengano implementate almeno le seguenti funzioni:

- `inserimento(...)` che chieda all'utente il numero di partite da giocare, che deve venire restituito tramite un puntatore a intero passato come argomento.
- `gioca(...)` che prenda in input `passi` e N e giochi una partita. Alla fine della partita la funzione aggiorna `passi` e restituisce 0 se ha vinto Mario, 1 se ha vinto Gino.
- `salva_freq(...)` che scriva su file i risultati.

Suggerimento: in `gioca(...)` bisognerà tener traccia di quali numeri sono usciti e quali no: questo si può fare, utilizzando un po' di memoria in più del necessario, con un array di 21 caselle.

► **Seconda parte:** Utilizzando il file `passi.dat` creare con python il relativo grafico. Infine, salvare un'immagine di tale grafico in un file chiamato "passi.png". Lo script python si dovrà chiamare `nome_cognome.py`. **Nota Bene:** il grafico dovrebbe avere un massimo per $n = 6$.