

Laboratorio di Calcolo per Fisici, Quarta esercitazione

Canale D-K, Docente: Livia Soffi,
Esercitori: Prof. S. Rahatlou e Prof. R. Faccini

Lo scopo della quarta esercitazione di laboratorio è di simulare il gioco della roulette sfruttando la **generazione di numeri pseudocasuali**.

Nel gioco della roulette semplificato, la ruota contiene **36 caselle numerate da 1 a 36**. Il croupier lancia una piccola palla bianca, e la fa ruotare nella roulette, finché essa non si ferma su uno dei numeri. Le puntate sono piazzate sul tavolo, secondo i possibili esiti del lancio della palla. Un giocatore può scommettere che la pallina finisca su:

- Un **numero preciso** compreso tra 1 e 36
- Un numero **dispari o pari**
- Un **numero basso** [1-18] **o alto** [19-39]



► Prima parte (obbligatoria)

Scrivere un programma chiamato ***roulette.c*** che simuli N lanci della pallina.

1. Stampare un messaggio di istruzioni per l'utente che chieda di **inserire** N .
2. Acquisire il numero di lanci N , assicurandosi che N sia minore di 10000.
3. Per ogni lancio:
 - (a) generare un numero **intero random tra 1 e 36**
 - (b) Riconoscere se il numero generato è pari (P) o dispari (D), alto (A) o basso (B).
 - (c) Stampare su schermo l'esito del lancio nel formato dell'esempio:
"Hai generato il numero 7. Il numero è dispari e basso."
 - (d) Contare quante volte si verificano ciascuno dei casi precedenti: $P/D/A/B$
4. Alla fine degli N lanci stampare **la frequenza con cui si è verificato ciascun risultato** $P/D/A/B$.
5. Eseguire il programma *roulette.c* per $N = 10, 100, 1000, 10000$.
6. Scrivere su un file *esiti.dat* il valore di N e delle frequenze con cui si sono verificati i risultati $P/D/A/B$. Come cambiano le frequenze in funzione di N ? Il risultato è coerente con le vostre aspettative?

NB. Si ricorda che per compilare ed eseguire il programma in C si deve digitare sul terminale:
`gcc roulette.c -o roulette.x -lm -Wall`
`./roulette.x`

► **Seconda parte (facoltativa)**

Scrivere il programma *gioco.c* per simulare una vera partita di roulette.

1. All'inizio del gioco il giocatore ha un **credito di 100 euro**
2. Chiedere all'utente **che somma vuole puntare**. La cifra deve essere compresa tra 2 e 10 euro e comunque sempre inferiore o uguale al credito.
3. Chiedere all'utente **su che numero scommette**. Controllare che il numero sia tra 1 e 36.
4. Generare il numero random tra 1 e 36 corrispondente all'esito del lancio della pallina
5. Se la pallina è finita sullo stesso numero scelto dall'utente, questi **vince due volte la sua posta, altrimenti perde la posta**.
6. Stampare su schermo il numero uscito e un messaggio che dica all'utente **se ha vinto o se ha perso e a quanto ammonta il suo credito**.
7. Implementare l'opportuno costrutto iterativo che permetta all'utente di continuare a giocare **finchè il credito e' almeno 2 euro e comunque per non più di 20 mani**.
8. Scrivere un messaggio che informi l'utente alla fine del gioco. Specificare quante mani sono state giocate e se il suo credito finale è nullo, altrimenti stamparne il valore.

Suggerimenti:

- La funzione **rand()** definita nella libreria **stdlib.h** permette di generare numeri **interi** tra 0 e **RAND_MAX**.
- L'algoritmo di generazione deve essere inizializzato con un seme **una sola volta**:
int seed;
seed = time(0);
srand(seed);
Si ricorda che la funzione **time(0)** e' definita nella libreria **time.h**
- Per generare un numero **intero** casuale compreso **tra 1 e nmax** si usa:
i=rand() % nmax + 1;
- Nel calcolo della frequenza, prima di fare il rapporto di due interi bisogna fare il casting del numeratore a double come segue:
double freq;
freq = (double)Num/Den;