

Corso di Laboratorio di Calcolo - Prova pratica finale

Prova Pratica 2019.1 *Il paradosso di Monty Hall*



Il *paradosso di Monty Hall* è un sorprendente quesito di teoria della probabilità. Si tratta di un paradosso perché la sua soluzione è inequivocabile, ma l'intuizione quasi sempre porta a risultati completamente sbagliati. Il problema si può formulare nei termini seguenti.

Supponete di dover scegliere una di tre carte coperte. Le tre carte sono un Asso, un Cavallo e un Re. Se pescate un Asso vincete un premio, altrimenti dovete pagare voi. Una volta fatta la scelta e prima di scoprire la carta, il mazziere, che può vedere le carte, scopre una delle due rimaste facendo attenzione a lasciare coperto, qualora non sia la carta che avete scelto voi, l'asso. A questo punto avete la possibilità di scegliere se cambiare carta o di tenere quella che avete scelto inizialmente. Cosa conviene fare?

La maggior parte delle persone pensa che cambiare carta non sia né conveniente né sconsigliato, poiché ritiene che, a questo punto, ci sia il 50 % di probabilità di aver fatto la scelta giusta. In realtà, conviene cambiare carta, perché la probabilità di aver scelto la carta giusta è solo del 33 % circa ($1/3$), mentre cambiare carta implica una probabilità di vincita pari a $2/3$.

In questo programma simuliamo il gioco e verifichiamo sperimentalmente quanto sopra.

1. Il programma chiede all'utente quante volte vuole simulare la giocata. Il numero N di giocate può essere la massimo 10 000.
2. Qualora l'utente immetta un numero non ammesso, il programma deve reiterare la richiesta fino a quando l'utente non rispetta le prescrizioni.
3. Il programma chiede all'utente se deve simulare un giocatore furbo o meno.
4. Attraverso una funzione `assign()` che riceve in ingresso un *array* di tre componenti, il programma distribuisce le carte in modo casuale. Ogni componente dell'*array* rappresenta una carta coperta. Il suo contenuto rappresenta la carta. Le carte con indice 0, 1 e 2, per esempio, possono essere un Asso, un Cavallo e un Re oppure un Re, un Asso e un Cavallo, etc..
5. Il programma, quindi, sceglie casualmente una delle tre carte.
6. Una funzione `change()` che riceve in ingresso la scelta fatta dal giocatore (il programma) e la sequenza delle carte, suggerisce la carta che si può scegliere in alternativa. In sostanza, `change()` rimuove dalle possibili scelte una delle due carte non scelte, accertandosi che quella vincente (l'asso) resti tra le alternative. Suggerisce quindi di cambiare carta con quella rimasta.
7. A questo punto, se il giocatore è furbo si scopre la carta scelta come suggerito da `change()`, altrimenti si scopre quella scelta inizialmente. Se la carta è un asso, il giocatore guadagna un euro, altrimenti ne perde uno.
8. I punti da 3 a 6 si ripetono fino a quando il numero di giocate raggiunge il numero N scelto dall'utente oppure se il giocatore ha perso tutto il suo budget. Inizialmente il giocatore possiede un *budget* di 100 euro. Al termine della simulazione il programma scrive su schermo la somma rimasta nelle tasche del giocatore.

Per controllare il funzionamento del programma si suggerisce di stampare, a ogni iterazione, la sequenza delle carte distribuite, la scelta iniziale del giocatore (indice della carta e suo valore), la proposta fatta dal mazziere (sempre con indice e valore), la scelta definitiva e la somma rimasta.

Accedete al computer con username `studente` e password `informatica`. Scrivete il programma in un unico file di nome `<cognome>.<nome>.c` nella home directory. Dal nome del *file* eliminate tutti i caratteri speciali. Ad esempio, lo studente Marco D'Alò scriverà il programma in un *file* di nome `dalo_marco.c`. È consentito l'uso di libri e appunti, ma non di mezzi idonei al collegamento a Internet.
