

Esercizi di programmazione

I due esercizi seguenti sono presi da esami passati (prof. Albarelli) e riportano esercizi simili a quelli che potrebbero essere proposti nei prossimi esami. Sottolineo però che, per tutti gli esami svolti in modalità online, il testo dell'esercizio sarà necessariamente più circostanziato (e di conseguenza più lungo) in quanto è necessario restringere l'interfaccia delle classi implementate all'interno dell'esercizio per poter eseguire poi i test automatizzati su HackerRank.

Altri esercizi simili di preparazione all'esame sono quello pubblicato su HackerRank e sul Moodle del corso come simulazione d'esame, e le quattro esercitazioni date durante lo svolgimento del corso.

Esercizio 1

Progettate e implementate tutte le classi che ritenete necessarie per realizzare un simulatore del popolare gioco d'azzardo solitamente chiamato "video poker". Il mazzo di carte ne contiene 52, 13 per ciascun seme, e viene mescolato all'inizio del gioco: dovete individuare una modalità di mescolamento che sia equa, anche se non è necessario che sia efficiente. Successivamente vengono mostrate le prime cinque carte del mazzo al giocatore, che ne può rifiutare alcune, anche tutte, o nessuna. Le carte rifiutate vengono sostituite con altre, prelevate ordinatamente dal mazzo. A questo punto, sulla base delle cinque carte che il giocatore ha in mano, il programma comunica il punteggio ottenuto, che deve essere il maggiore tra i seguenti, elencati in ordine crescente:

- No pair ("Niente"). La configurazione peggiore, che contiene cinque carte spaiate che non compongono alcuna delle configurazioni elencate nel seguito.
- One pair ("Coppia"). Due carte dello stesso valore, ad esempio due regine.
- Two pairs ("Doppia coppia"). Due coppie, ad esempio due regine e due cinque.
- Three of a kind ("Tris"). Tre carte dello stesso valore, ad esempio tre regine.
- Full House ("Full"). Un tris e una coppia, ad esempio tre regine e due 5.
- Four of a kind ("Poker"). Quattro carte con lo stesso valore, ad esempio quattro regine medesimo seme.

Non è necessario implementare alcuna interfaccia utente o permettere effettivamente di giocare. Semplicemente scrivere un main che svolga le seguenti operazioni tramite gli opportuni metodi:

- Mescola il mazzo;
- Preleva una mano per il giocatore;
- Se non ha almeno un full cambia 3 carte (se ha coppia), 2 carte (se ha tris) o una carta (se ha doppia coppia);
- Preleva una mano per il computer e stampa chi ha vinto.

Esercizio 2

Simulate un sistema di car sharing usato da lavoratori pendolari per caricare e scaricare passeggeri in apposite stazioni. Immaginate che ci siano 30 stazioni, a un miglio di distanza una dall'altra lungo un determinato percorso: per ogni stazione generate un numero casuale di automobili, ciascuna con una destinazione determinata casualmente, e un numero casuale di passeggeri, ciascuno con una destinazione ancora determinata casualmente. Ogni guidatore carica a bordo tre passeggeri a caso, scelti tra quelli la cui destinazione precede quella del guidatore, scaricandoli nella stazione giusta e caricando, se possibile, altri passeggeri. La tariffa che viene pagata al guidatore è un importo costante per ciascun miglio di distanza percorso da ciascun passeggero. Eseguite la simulazione mille

volte e visualizzate l'incasso medio per miglio percorso. Progettate le classi Car (automobile), Passenger (passeggero), Station (stazione) e Simulation (simulazione).