

Esercitazione 24/11/2016

Argomenti:

- Array;
- Funzione "rand()";
- Funzioni;

Esercizio 1:

Scrivere il programma che gestisce un tabellone di punteggi (leaderboard) che permetta di inserire il punteggio di 10 giocatori (Giocatore 0, Giocatore 1, ... Giocatore 9). Una volta che i dati sono stati immessi, stampare a video la lista dei giocatori e dei punteggi inseriti in ordine inverso a cui sono stati inseriti e la media dei punteggi.

Esercizio 2:

Scrivere il programma di un gioco simile a "Battaglia navale":

- In una griglia 10 X 10, il programma deve posizionare casualmente una piccola nave nemica (1 X 1).
- Ad ogni turno, il giocatore dichiara la casella da colpire, inserendo X e Y della cella bersaglio (se il valore non è valido, segnalare e richiedere);
- Se il giocatore ha colpito il bersaglio, segnalare la vittoria e chiedere se si vuole ricominciare o meno. Altrimenti dichiarare quanto si è andati vicini al bersaglio stampando "Fuoco" se il colpo è arrivato entro un quadrato 3 X 3 intorno al bersaglio, "Fuochino" entro un quadrato 5 X 5, "Acqua" altrimenti.

Esercizio 3:

Scrivere la versione deterministica del gioco dell'esercizio 2 se si è scelto per una versione non deterministica. Scrivere la versione non deterministica se si è optato per una versione deterministica.

Esercizio 4:

Scrivere il programma di un gioco che gestisca una navicella spaziale. La navicella avrà una serie di parametri e disporrà di una serie di missioni che influenzeranno tali parametri.

Parametri:

- Carburante (100 unità di partenza);
- Munizioni (50 unità di partenza);
- Scorte di sopravvivenza (100 unità di partenza)

Missioni:

- Esplorazione (-20 carburante, -20 scorte) - id: "esplorazione";
- Difesa della Terra (-5 carburante, -5 scorte, -20 munizioni) - id: "difesa";
- Scorta ricercatori (-10 carburante, -10 scorte, -10 munizioni) - id: "scorta";

Presentare al giocatore le missioni disponibili e chiedergli di sceglierne una digitando la stringa identificativa (id). Gestire il caso della missione non valida, avvertendo il giocatore.

Una volta che il giocatore ha scelto la missione, applicare ai parametri i modificatori indicati, stampare il nuovo assetto dei parametri e controllare: se un parametro è giunto a zero (o meno), avvertire il giocatore che la navicella ha terminato la quantità X e terminare l'applicazione. In caso contrario riproporre le missioni.

Ogni missione dovrà essere implementata mediante una corrispondente funzione.

Compito per casa:

TRIS AVANZATO

Realizzare il gioco del tris.

Sequenza di gioco

- Introduzione;
- Round di gioco 1;
- Round di gioco 2;
- ...
- Esito (Vittoria / Sconfitta)

Introduzione

Segnalare che il giocatore avrà le X e l'avversario computerizzato (che chiameremo CPU) avrà gli 0 (carattere zero).

Round di gioco

Ogni round avrà questa sequenza:

- Presentare al giocatore la tavola di gioco
es.:

		0		X
—		—		—
		X		
—		—		—
		0		

- Chiedere al giocatore di inserire la colonna in cui posizionare la X;
- Chiedere al giocatore di inserire la riga in cui posizionare la X;
- Validare la mossa del giocatore (verificare che le coordinate date corrispondano ad una cella esistente e vuota. Si può assumere che l'input sia un intero). Se le coordinate sono errate, richiederle.
- Presentare la tavola di gioco aggiornata (usare una funzione dedicata per generare la tavola);
- Scegliere una posizione libera casuale come mossa di CPU;
- Presentare la tavola di gioco aggiornata (usare una funzione dedicata per generare la tavola);
- Calcolare l'esito. In caso di vittoria o di sconfitta, procedere all'esito;
- Altrimenti ripetere il round di gioco;

Esito

Se il giocatore ha perso, chiedere al giocatore se vuole rigiocare e, in tal caso, far ripartire il gioco, altrimenti terminare il programma.

Se il giocatore ha vinto, chiedere al giocatore se vuole rigiocare in modalità difficile e, in tal caso, far ripartire il gioco con la seguente differenza: se il giocatore ha messo la X in due celle allineate e la terza cella è libera, CPU la dovrà mettere lo 0 in tale cella.

Esempio:

Turno del giocatore appena terminato

		0		X
—		—		—
		X		
—		—		—

Mossa di CPU

		0		X
—		—		—
		X		
—		—		—
0				

Ovviamente, se il giocatore rifiuta di fare una nuova partita, il gioco termina.