RELAZIONE 2D-RACE

**StartGame**

StartGame è la funzione richiamata direttamente dal menù (iniziale o di Game Over), è il loop che racchiude tutte le altre funzioni. StartGame prende come parametro un intero ‘D’, ovvero la difficoltà (1, 2 o 3) scelta dal giocatore nel menù, e restituisce un intero, ovvero il massimo punteggio totalizzato durante la partita, in modo da poterlo visualizzare nella schermata di Game Over.

Prima dell’inizio del ciclo “infinito” vengono dichiarate delle costanti per la gestione degli ostacoli e per la diminuzione della benzina, per poi costruire i vari oggetti (strada, fuel, macchina, liste di ostacoli e score) e inizializzare le variabili di livello, coordinate della macchina e “Flashes” (numero di “flash” della macchina quando un ostacolo viene colpito). Il ciclo del gioco è un loop che termina solo in caso di esaurimento del carburante, di raggiungimento dello score 0 o nel caso in cui dal menù di pausa si scelga di uscire. Tutte le linee di codice all’interno del loop sono invece funzioni implementate nelle diverse classi richiamate, permettendo la creazione, l’eliminazione e la gestione degli ostacoli, il movimento dell’auto e la gestione di carburante e punteggio.

**Liste**

Liste è una libreria che contiene la definizione delle liste di oggetti implementati e inoltre definisce le funzioni di base per la gestione delle liste.

-Add<T>\{Gas} permette di aggiungere un nuovo oggetto in coda alla lista apposita in seguito a un random presente in Funzioni che decide quale item inserire e quando.

-Del<T>\{Gas} permette di eliminare l’elemento in testa a una determinata lista di oggetti casualmente.

-Man<T> permette di controllare se l’oggetto ha colliso con la macchina, quindi diminuisce/aumenta lo score e il fuel, inoltre fa apparire, stampare e muovere gli oggetti richiamando le funzioni definite nelle classi apposite.

**DoStrada**

DoStrada è la classe che contiene la matrice “scheletro” del gioco e le sue funzioni permettono di stampare tutti gli oggetti i cui caratteri vengono passati alla classe stessa, valutando anche il colore da impostare per ogni carattere.

**Fuel**

Fuel è la classe che gestisce il carburante e contiene un intero (il livello effettivo di carburante) e un array di caratteri ‘ \* ‘, ognuno di questi equivale a 30 punti di carburante. Le funzioni “decr” e “incr” permettono di aumentare e diminuire il carburante modificando il numero di asterischi, mentre “drawast” passa quest’ ultimi alla matrice strada.

**Title**

È la classe che stampa il titolo del gioco con una ascii art e visualizza il prompt da premere per inizializzare il menù.

**Menù**

La classe Menù si occupa di gestire l'avvio e l'uscita dal gioco, andrà a fornire allo start game il valore di difficoltà da impostare (il menù è navigabile con "W" e "S" rispettivamente sopra e sotto, per selezionare premere invio quando l'opzione desiderata è colorata di verde).

**Macchina**

La classe Macchina si occupa della gestione degli input forniti dall'utente ("W" "A" "S" "D" sopra, sinistra, giù, destra) per far muovere il veicolo in gioco e definisce le coordinate della macchina da stampare nella matrice generata da DoStrada, inoltre controlla che il veicolo non superi i limiti della mappa e in caso di tentativo di superarli andrà a penalizzare il punteggio. Nel caso la macchina vada "Fuori strada" o impatti un ostacolo essa lampeggerà.

**Pausa**

La classe pausa va a definire un Menù richiamabile dall'utente con "P" nel corso della partita, l'utente avrà a disposizione 2 scelte: continuare la partita corrente o terminarla tornando al menù iniziale (i comandi di interazione sono gli stessi di menù).

**Ostacoli**

Le classi riguardanti gli ostacoli sono molto simili tra loro, le uniche differenze si possono trovare nel range/frequenza di spawn, nei danni applicati da ogni ostacolo alla macchina del giocatore e nel simbolo che li identifica sulla mappa; inoltre un particolare oggetto, “Ostacolo3”, è anche dotato di un movimento randomico. Ogni ostacolo/bonus ha, oltre il simbolo che lo identifica, delle coordinate (xPos, yPos) e un value, utile soprattutto nelle modifiche di punteggio derivanti dalle collisioni.

* Costruttore

Il costruttore imposta le coordinate xPos e yPos a dei valori arbitrari totalmente ininfluenti nell’esecuzione, utili solo ad allocare l’oggetto.

* Appear()

La funzione appear() ha come input un numero randomico e il livello attuale. È la funzione che permette la creazione dell’oggetto, attribuendogli una xPos randomica, (il range di spawn è unico per ogni oggetto, viste le diverse dimensioni), e la yPos viene settata a 0. Prima di un certo livello la funzione decide sia se l’oggetto spawna, settando la yPos da 29 a 0 e quindi riallocando l’oggetto in modo che sia visibile e sensibile alle collisioni solo in determinati casi dettati da un numero randomico, sia dove spawna, cambiando la xPos in funzione del numero randomico. Dopo il livello, la funzione decide esclusivamente la nuova xPos dell’oggetto.

* Print()

La funzione print() ha come input un’istanza della classe DoStrada e si occupa di mostrare l’oggetto sulla mappa. I punti che compongono ogni oggetto vengono disegnati da un’altra funzione, la funzione print() richiama questa funzione singolarmente per ogni pixel dell’ostacolo.

* Disegna()

E’ la funzione che si occupa di modificare uno alla volta i punti/pixel relativi all’oggetto, modificando l’istanza della classe DoStrada oppure cancellando l’oggetto se le caselle necessarie non sono tutte libere.

**Score**

La classe riguardante il punteggio, “Score”, si occupa di aggiornare il punteggio e il livello, stamparli, aumentarli o diminuirli in funzione del tempo e delle collisioni. In particolare la classe “Score” comprende delle funzioni che modificano il punteggio guadagnato al secondo (modificando l’attributo “aumento”), a seguito di collisioni sia positive che negative e gli attributi tengono anche traccia del punteggio massimo. Il resto della classe è necessario per stampare il livello e il punteggio, carattere per carattere, sulla matrice principale.

* Stampa() Sia nel caso dello score che del livello, la stampa è fatta attraverso una struttura, allocata dinamicamente in base al numero necessario di cifre. La struttura utilizzata è una semplice lista di interi, chiamata Cifre, che contiene un intero e un puntatore al next.