Software system Architecture

Documentazione del codice

**Introduzione**

Questo documento spiega come è sviluppato il codice che permette di creare il gioco del dinosauro di Google tramite programmazione in C

**Struttura del progetto**

Il progetto e strutturato da 4 librerie e 4 file C

Le librerie sono:

* appearance.h
* game.h
* menu.h
* minunit.h

I file C sono

* appearance.c
* game.c
* menu.c
* main.c

**Requisiti minimi**

Per poter compilare i file e creare il file eseguibile e necessario avere un pc con le seguenti specifiche e librerie installate

* Ubuntu 14.04.6 o superiore
* 2 GB di RAM
* 2 GHz
* Ncurses (Libreria)

Infine per compilare inserire il comando:

gcc main.c game.c appearance.c menu.c -lncurses -o dino.exe

Per lanciare il gioco il comando è:

./dino.exe

***appearance.c***

usa delle funzioni contenute nel file libreria appearance.h per muovere il cursore e stampare a schermo la sua posizione

* void dinosaur1(int y, int x)  **stampa il modello del dinosauro rivolto a dx**
* void dinosaur2(int y, int x) **stampa il modello del dinosauro rivolto a sx**
* void cactus1(int y, int x) **stampa il modello di un cactus**
* void cactus2(int y, int x) **stampa il modello di un cactus diverso da cactus1**
* void sun(int y, int x) **stampa il modello del sole**
* void moon(int y, int x) **stampa il modello della luna**
* void showLoss(int y, int x) **visualizza a schermo le opzioni in caso di sconfitta**
* void showTrex(int x, int y)
* void clearCactus1(int y, int x) **elimina i cactus stampati sullo schermo**
* void cleardinosaurDown(int diY, int diX) **elimina la parte sotto del dinosauro**
* void cleardinosaurUp(int diY, int diX) **elimina la parte superiore del dinosauro**

***appearance.h***

è un file libreria e contiene le funzioni per creare determinati elementi a schermo

*#ifndef APPEARANCE\_H*

*#define APPEARANCE\_H*

* *void dinosaur1(int y, int x);* **contiene la funzione per la stampa del dinosauro rivolto a dx**
* *void dinosaur2(int y, int x);* **contiene la funzione per la stampa del dinosauro rivolto a sx**
* *void cactus1(int y, int x);* **contiene la funzione per la stampa di un modello di un cactus**
* *void cactus2(int y, int x);* **contiene la funzione per la stampa di un modello di un cactuS**  **diverso dal cactus 1**
* *void sun(int y, int x);* **contiene la funzione per stampare a schermo il sole**
* *void moon(int y, int x);* **contiene la funzione per stampare a schermo la luna**
* *void showLoss(int y, int x);* **contiene la funzione per visualizzare a schermo le opzioni in**  **caso di sconfitta**
* *void showTrex(int x, int y);*
* *void clearCactus1(int y, int x);* **contiene la funzione per eliminare i cactus**
* *void cleardinosaurDown(int diY, int diX);* **contiene la funzione per eliminare la parte**  **inferiore del dinosauro**
* *void cleardinosaurUp(int diY, int diX);***contiene la funzione per eliminare la parte**  **superiore del dinosauro**

*#endif*

***game.c***

Compila il gioco facendolo funzionare e rende disponibili varie funzioni per l'utente che lo esegue

* int checkGame(int y, int x, int diY, int diX) **verifica se il gioco corrente puo essere**  **terminato nel loop attuale**
* int computeTime(int delayTime) **aumenta la velocita di gioco**
* void showdinosaur(int diY, int diX) **crea il modello del dinosauro selezionato**
* int computePrize(int score, int usedPrize) **da al giocatore il premio**
* void startEngine(int highScore, struct user firstUser) **avvia il motore di gioco principale**
* void all\_tests **inizializza i test**
* int run\_all\_tests **avvia i test**

***game.h***

avvia il motore di gioco e "crea" il dinosauro protagonista del gioco

#ifndef GAME\_H

#define GAME\_H

* void startEngine(int highScore, struct user firstUser); **avvia il motore di gioco e stampa le**  **opzioni del menu**
* void showdinosaur(int diY, int diX); **stampa il modello del dinosqauro selezionato nel**  **menu**

#endif

***highscore.txt***

mantiene il punteggio piu alto realizzato

***LICENSE***

é la licenta di utilizzo del software

***main.c***

è la funzione (main) principale del gioco senza la quale non partirebbe

* int main(int argc, char \*\*argv) **avvia i test e il menu iniziale**

***menu.c***

richiude le funzioni di salvataggio del punteggio nel caso sia piu alto dell'highscore e eventualmente lo salva nel file highscore.txt

* void startMenu **visualizza il menu iniziale prima dell'avvio effettivo del gioco**
* void endGame **visualizza il menu per ricominciare o uscire nel caso si perdesse**

***menu.h***

contiene la funzione per la creazione del menu e contiene la struct per contenere i dati dell'utente

* struct user **contiene i dati dll'utente come età,nome e cognome**
* void startMenu(); **richiama la funzione per il menu iniziale**
* void endGame **richiama la funzione per visualizzare determinati elementi dell'endgame**

***minuti.h***

richiama la funzione per avvio dei test

* int run\_all\_tests(int argc, char \*\*argv) **richiama l'esecuzione dei test**

***README.md***

contiene le informazioni su chi ha creato i file,i vari comandi per controllare il dinosauro e come compilare il gioco

Esecuzione del codice lato utente



* Spiegazione del codice scritto
* File 1 (appearance.c): nelle funzioni scritte, viene scritto come deve essere, prima e dopo l’animazione, il dinosauro, il cactus quando il dinosauro sta correndo, cosa succede dopo che il dinosauro ha saltato il cactus, cosa succede al cactus dopo che il dinosauro gli ha sparato il suo “prize”, cosa succede se il dinosauro va contro a un cactus e che cosa si vede dopo la morte del dinosauro, il sole, la luna e che cosa si vede prima di iniziare a correre col dinosauro
* File 2 (game.c): nelle funzioni scritte, viene spiegato al terminale che cosa far vedere su schermo nel mentre che hai le dita sui tasti “spazio” e “k” (lo “spazio” fa saltare il dinosauro mentre “k” fa lanciare il “prize” → un missile)
* File 3 (main.c): Nel codice di questo file vengono solamente “inclusi” i file in cui è scritto totalmente il codice
* File 4 (menu.c): Nel codice di questo file viene descritto come deve essere il menu di inizio gioco ed in che ordine devono essere mostrati gli input ed anche cosa deve essere mostrato quando il giocatore perde