FONDAMENTI DI INFORMATICA

ELABORATO

"UPWORDS!"

DI RICCARDO CAMPI

INTRODUZIONE

L'Elaborato ha come scopo quello di realizzare in UPWORDS!, una versione digitale e in italiano del gioco da tavolo anglosassone Up Words, una variante tridimensionale del classico Scarabeo.

Al fine di rendere l'Elaborato prodotto più in linea con le specifiche fornite si è scelto di apportare alcune modifiche al gioco, come ad esempio l'adozione delle tessere dello Scarabeo italiano o l'eliminazione di alcune regole adatte solo alla versione anglosassone.

ANALISI DEI REQUISITI

Le regole del gioco restano per la maggioranza invariate e sono state tradotte ed inserite nel programma sotto il nome di "istruzioni". Si può giocare quindi da 2 a 4 giocatori e si possono formare parole come nella versione da tavolo. A cambiare sono invece le tessere (ora riprese dallo Scarabeo) e alcune regole, inconciliabili con la digitalizzazione o l'italianizzazione.

Il programma riconosce tutte le parole di 3 o più lettere, controllandone la validità in caso di verbi declinati o estesi con suffissi e verificandone l'esistenza in un vocabolario digitale, espanso per motivi di giocabilità.

A differenza della versione anglosassone, UPWORDS! permette la modifica di una parola esistente anche cambiandone solo la pluralità o il genere, vista la maggiore complessità dell'italiano rispetto alla lingua originale.

UPWORDS! mette inoltre a disposizione del giocatore di turno uno strumento inimmaginabile per la versione da tavolo: il suggerimento. Il programma vaglia centinaia di migliaia di combinazioni ed incrocia i risultati con il vocabolario per poi fornire la lista delle lettere effettivamente posizionabili sulla scacchiera, ordinata in base al punteggio e aggiornata in tempo reale.

La grafica tridimensionale e la GUI semplificata inoltre migliorano giocabilità e impatto del gioco sui giocatori.

NOTE SULLA REALIZZAZIONE

Il programma si avvale delle funzionalità grafiche fornite dalla libreria open-source OpenGl, la cui gestione è stata resa meno difficoltosa per mezzo di una delle numerose versioni della libreria Gl Utility Toolkit, in questo caso la FreeGlut.

Ci si è inoltre avvalso, previa comparazione con altre analoghe, delle strutture dati messe a disposizione dalla STL, scongiurando così il rischio di fare uso di strutture deprecate.

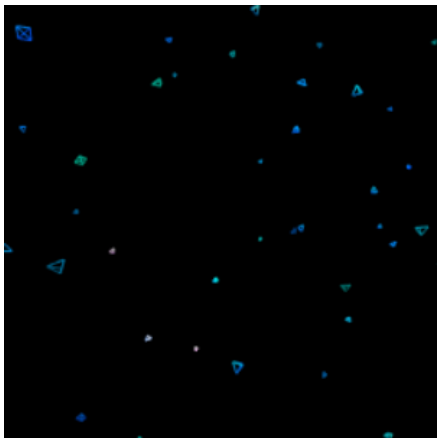
Il paradigma grazie al quale è stato possibile progettare soluzioni semplici al fine di risolvere problemi complessi è esprimibile con il concetto di oggetto; Quasi tutte le entità virtuali necessarie al funzionamento del gioco o della grafica sono quindi pensate come corpi aventi specifiche informazioni atte a descriverne lo stato. Durante lo sviluppo si è cercato quindi di dotare le entità sopraccitate di alcune proprietà generalizzate in categorie, che hanno permesso di gestirle in maniera del tutto accomunabile alla gestione degli oggetti nella realtà. A conferma di questa tesi potrebbe essere portato come esempio il sistema che permette di coordinare la scacchiera e le tessere durante il gioco, realizzato grazie a vettori di strutture facilmente modificabili da funzioni apposite.

Un caso estremo di applicazione dei concetti appena espressi è invece quello attuato per le finestre utilizzate per la GUI dell'elaborato. Si è così voluto rendere l'organizzazione delle stesse più simile possibile a quella delle finestre delle Win32 API. Le informazioni di ogni finestra sono infatti immagazzinate in un vettore di strutture, tra le quali vi è un vettore la cui funzione è a sua volta quella di organizzare gli oggetti presenti all'interno di ciascuna finestra. Tutti gli oggetti componenti la GUI sono inoltre dotati di handle volti all'identificazione, di callbacks in caso di evento e di costanti che ne permettono la generalizzazione, ulteriore analogia con la libreria Windows. Alcune funzioni sviluppate appositamente permettono poi di accedere o modificare ogni singola informazione in modo molto

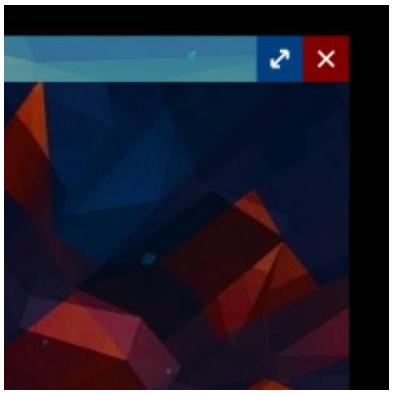
semplice e sono utilizzate, tra l'altro, per rendere la lettura o modifica del codice più naturale.

Ci si avvale poi di thread che permettono l'esecuzione di codice molto pesante (quale l'effettuazione di verifiche sulla correttezza o la funzionalità dei suggerimenti) o di codice contenente funzioni bloccanti (quale il caricamento iniziale) senza penalizzare la qualità dell'intero programma, evitando quindi latenze e attese.

PROVE DI TEST

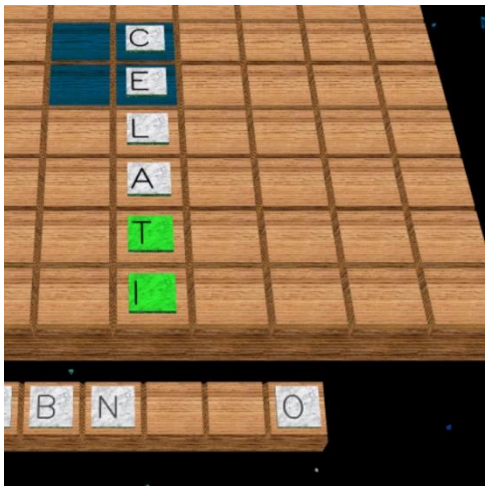


Particolare della grafica di sfondo: ogni prisma contiene informazioni sulla posizione, direzione, rotazione, velocità e colore, randomizzate durante l'inizializzazione. Una funzione goniometrica ne governa poi il moto in funzione del tempo, al fine di evitare la dispersione degli stessi.

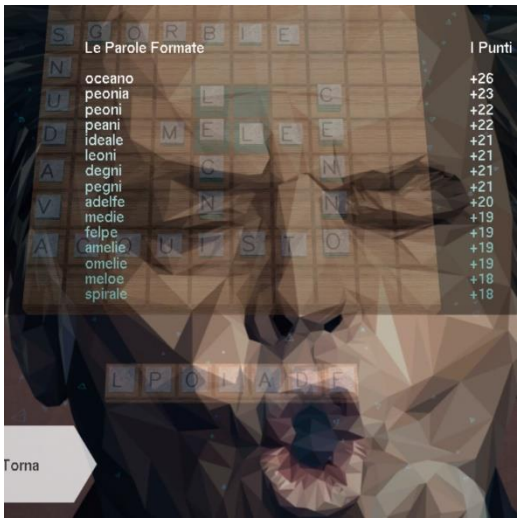


Ogni finestra ha dei bottoni predefiniti, in similitudine a quelle Microsoft. Se accade un evento, le finestre possono attuare meccanismi a callback, agendo così in risposta all'evento stesso.

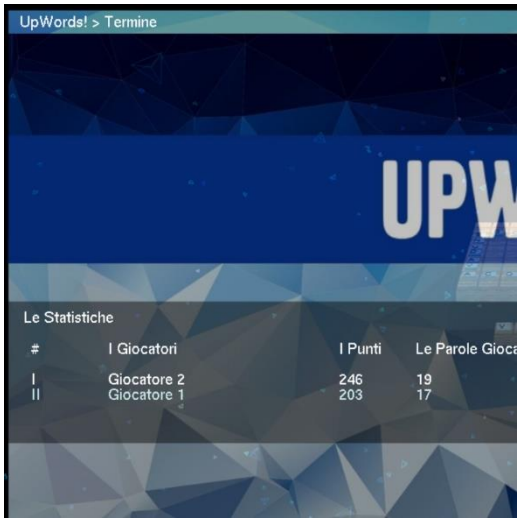
Schermata della scelta dei giocatori. Essi sono immagazzinati in un vettore di strutture che contiene tutte le informazioni necessarie. Per salvare una partita, il vettore stesso viene crittografato e scritto su file.



Parola formata accodando due lettere nuove alla precedente. Le lettere bianche sono fisse mentre quelle verdi sono da validare; le celle centrali sono blu. La scacchiera è disegnata in modo da ricordarne una in legno massello mentre le tessere simulano un marmo.



La schermata degli spunti si aggiorna in tempo reale visualizzando le migliori combinazioni; esse sono interattive e mostrano la configurazione al passaggio del mouse. Come sfondo un Jack Nicholson nullafacente invita i giocatori a non farne uso troppo spesso.



Schermata finale, nella quale sono stampati i punti e alcuni dati statistici interessanti.