

# Progetto Programmazione 3 e Lab Programmazione 3

Anno 2014-2015



Michael Nicoella, matr. 0124/45



# Introduzione

L'obiettivo di questo progetto è quello di realizzare un gioco di tipo piattaforma 2D con Java in versione desktop.

Il gioco, che si chiama “Rhaegal”, è composto da 4 livelli principali con difficoltà sempre crescenti, più un livello boss.

Come si può notare dalla diapositiva precedente, appena si avvia il gioco abbiamo a nostra disposizione 4 opzioni:

- Start avvia il gioco;
- Commands visualizza i relativi comandi di gioco;
- Scores visualizza gli ultimi 15 punteggi effettuati giocando;
- Quit termina il gioco.

L'obiettivo del gioco è quello di, come in tutti i giochi del genere, riuscire ad arrivare alla fine del livello vivi e con il maggior numero di vite e punteggio.

# Struttura

Il gioco è strutturato in questo modo:

Una volta avviato, il programma cicla all'infinito in una condizione del tipo `while(true)`

```
while(true)
{
    update();
    draw();
    drawToScreen();
}
```

Questi metodi principali, `update()`, `draw()` e `drawToScreen()` servono rispettivamente per aggiornare tutte le dinamiche di gioco, creare e infine disegnare su schermo l'oggetto grafico finale.

# Struttura, metodo update()

I metodi update() e draw() in particolare sono ereditati e, in quasi tutti i casi, sovrascritti dalle varie classi che li implementano.

Il metodo update() è utilizzato dalle varie classi per aggiornare tutto l'ambiente (variabili, oggetti, liste, ecc), come la posizione ed i movimenti del player e le sue caratteristiche, le posizioni e i movimenti enemies, il punteggio, il controllo delle collisioni e tutti i calcoli necessari al corretto funzionamento del gioco.

# Struttura, metodo draw()

Il metodo draw() è utilizzato per creare oggetti grafici bidimensionali che possano essere in seguito visualizzati a schermo. In pratica ogni classe che implementa questo metodo crea oggetti grafici e stabilisce dove doverli visualizzare a schermo. Tutte le immagini (sprites, backgrounds...) sono mantenute nella cartella /resources e vengono quindi caricate da qui.

Le animazioni di gioco vengono create tramite sprites, figure bidimensionali che possono variare posizione rispetto allo sfondo. Ogni sprite contiene più frame che disegnati in sequenza formano l'animazione.

# Struttura, metodo drawToScreen()

Il metodo `drawToScreen()` non è mai ereditato, né implementato, da nessun'altra classe. Viene chiamato come ultimo metodo ad ogni ciclo del gioco ed è utilizzato per disegnare a schermo l'immagine finale del gioco che le varie classi hanno creato tramite i vari metodi `draw()`. Infatti ogni volta che una classe invoca il metodo `draw()` non fa altro che aggiungere (sovrapporre) un oggetto grafico alla stessa immagine che si sta creando e che alla fine sarà disegnata su schermo come immagine finale.