

# **Relatório Starfield 2**

## **Autores**

Diogo Santos Baptista - N°79405

Mário Liberato - N°84917

## **Resumo**

O programa inicia-se com um splash screen com o nome do jogo e a musica (Imperial March) é iniciada. Após o Splash screen é apresentado o Menu principal, aqui pode-se ver os créditos, autores, recordes e começar a jogar. Quando o jogador clica “enter” o jogo é iniciado e logo é lhe apresentado um Menu de escolha de nave, onde o jogador pode escolhe a nave que prefere (cada nave tem diferentes vidas), agora após a escolha da nave o jogador pode desfrutar das várias funcionalidades do jogo tal como, pontuação, aumento de dificuldade, sistema de disparos, asteróides e vidas extra.

Introdução

Neste trabalho foi-nos proposto a realização de um jogo (Starfield 2) com o objetivo de aprofundar os nossos conhecimentos de programação em java e de aprender a fazer um bom trabalho em grupo. Após a realização do nosso jogo foi-nos também proposto a realização de um relatório para suportar alguma informação sobre o jogo. Este relatório inicia-se com um pequeno resumo, em que nele estão descritas o principal aspeto do trabalho/jogo, de seguida (após esta introdução) é feita uma descrição detalhada sobre o jogo em que são mencionados os aspetos gerais do jogo e todos os requisitos de programação que usamos para a realização do trabalho. Por fim apresentamos uma conclusão, aqui é descrito um breve resumo do que foi feito e possíveis melhorias a adotar, e em seguida são apresentados os recursos externos usados.

## **Aspetos gerais do jogo**

O jogo inicia-se com o Splash Screen, durante 3 segundos, e a música também inicia (Imperial March). Esta última não para de tocar, estando sempre em “loop”.

De seguida o menu é apresentado, este usa estrelas geradas da mesma forma que as do jogo em si como fundo. Neste menu, podem ser consultados os autores (créditos), pode-se iniciar o jogo, ou podem-se alterar as opções de controlo (rato/teclado). Ao clicar “enter” o jogo é iniciado, com isto é apresentado o menu de escolha de nave onde são apresentadas várias naves de cores diferentes, cada uma com um número de vidas inicial diferente.

Durante o jogo, estrelas (bolas brancas) são geradas. A nave (triangular) não pode colidir com elas ou com os asteróides, ou perderá uma vida. Quando as vidas chegam a zero e a nave é uma vez mais destruída, o jogo acaba, e o ecrã de “Game Over” é apresentado. O jogo também apresenta aumentos de dificuldade (são feitos em intervalos temporais, aumentando o “framerate, o que torna o jogo mais rápido), vidas extra (podem ser apanhadas durante o jogo, geradas aleatoriamente), sistema de pontuação (calculada com base no framerate, podem-se ainda ganhar pontos destruindo asteróides), asteróides (gerados de forma semelhante às estrelas, ao chocar contra eles destrói-os, e faz a nave perder uma vida e podem dar pontos e vidas), sistema de disparo (usado o botão esquerdo do rato para destruir asteroides; com os controlos de teclado, pode ser usada a tecla “o”).

### **Implementação e discussão**

Para a música/som, usámos a biblioteca Minim.

É mostrado um splash screen (“Chanelure Gmnpolicemata Logo”), o qual fica no ecrã durante 3000ms. A função splashScreen (em splash.pde) usa uma variável para contar os loops. No primeiro loop, o tempo do sistema em milisegundos é guardado, e é comparado com o tempo actual para ver se já passaram os 3 segundos. Se sim, vai alterar a variável curScene, a qual é utilizada pela função game (starfield.pde, linha 407). Esta função, dependendo do valor do argumento (curScene), vai mostrar ecrãs diferentes. Esta função permite ainda activar/desactivar o modo de debug (B). Enquanto o modo de debug é usado, podem ser regulados o framerate e vidas.

No jogo, foram usadas duas funções para calcular colisões entre a nave e asteróides/estrelas, checkCollisionWithCircle (starfield2.pde, l.485) e checkCollisionWithPoint. (starfield2.pde,l.500) A primeira chama a segunda para todos os pontos à volta de um ponto de grau a grau. Para as balas, foram usadas a classe Bullet (classes.pde) e um ArrayList para as armazenar (starfield2.pde).

### **Conclusão**

Na realização deste trabalho em “Processing” foram aplicados todos os conhecimentos de programação em Java aprendidos na disciplina de Programação I, cumprindo os requisitos impostos pelas regras do trabalho. Com isto podemos afirmar que todos os objetivos inicialmente propostos foram atingidos com sucesso.

### **Referências**

“Imperial March” ~ John Williams, “Chandelure Gmnpolicemata Logo” ~ Mário Liberato, “1UP” ~  
Nintendo