

# Tecnologia em Jogos Digitais

## Projeto Integrador II / Geometria Analítica / Física para Jogos / Algoritmos e Programação II

### Asteroids<sup>1</sup>

#### Introdução

Asteroids é um jogo de fliperama lançado em 1979 pela Atari Inc. O jogo Asteroids foi um dos jogos mais populares e influentes da Idade de Ouro dos Jogos Arcade. Asteroids usa gráficos de vetor e uma visão bidimensional que envolve em torno de dois eixos tela.



O jogador controla uma nave espacial em um campo de asteroides, que periodicamente é atravessado por discos voadores. O objetivo do jogo é atirar e destruir asteroides e discos voadores enquanto não colidir com qualquer um, ou ser atingido pelo fogo os discos voadores. Uma outra curiosidade sobre o jogo é que ele foi um dos primeiros jogos com gráficos vetoriais, som, explosões e disparos de projéteis, tornando a experiência do jogo multissensorial e imergindo ainda mais os jogadores no game.

Asteroids foi, e ainda é, inspiração para inúmeros outros games, além de jogadores e programadores que querem começar apreender o desenvolvimento de games. Onde existir algum tipo de nave tentando sobreviver em meio a uma tempestade de pedras e afins, vagando abobalhadamente pelo universo, provavelmente, esse jogo, foi inspirado no Asteroids.

#### Objetivo

O objetivo do projeto integrador desse semestre é desenvolver um game **inspirado** no famoso Asteroids que era jogado nos fliperamas da década de 70. O seu jogo deverá ter uma nave que dispara **tiros** (vários de uma só vez) nos asteroides. Além disso, a nave deverá girar para esquerda/direita e avançar para frente.

Na tela os asteroides se movem em direções aleatórias e quando um asteroide ultrapassa uma borda da janela ele reaparece em outra borda, por exemplo, se sumir na borda superior ele reaparece na parte inferior e continua se movendo na mesma direção, quando o jogador acerta um tiro em um asteroide, esse se divide em asteroides menores. Os asteroides grandes se movem com a mesma velocidade, quando ele é dividido os asteroides menores devem se mover mais rápido que os grandes, e quando um asteroide colide-se com um outro não deve acontecer nada, simplesmente eles continuam seu caminho

#### Estrutura do Jogo

O jogo deverá ter alguma estrutura global de fluxo (menus). Um jogo que já começa dentro de uma fase, e termina ("fecha") assim que se chega ao final não é muito amistoso. Uma sugestão de configuração para a estrutura é a seguinte:

Introdução → Menu → Jogar

---

<sup>1</sup> **Importante:** A especificação desse trabalho pode sofrer modificações de acordo com discussões que tivermos em sala de aula.

Ao iniciar o jogo deve ser apresentada uma tela de introdução, possivelmente com algum logotipo e/ou a história do jogo, e então o jogador **é levado a uma tela de menu com as opções de navegação tais como Novo Jogo, Ajuda e Sair.**

O seu jogo deve ter pelo **menos duas fases** e durante uma fase o jogador deve possuir um certo número de vidas, caso aconteça uma colisão entre a nave e um asteroide o jogador perde uma vida, caso não tenha mais vidas então o jogo termina e volta ao início ou vai para o Menu. Para informar o jogador o seu jogo deverá ter contadores de vida e pontos que serão atualizados conforme o progresso do jogador, tente estabelecer critérios para incrementar e decrementar os contadores do seu jogo, por exemplo: destruir um asteroide pequeno ganha 5 pontos. Estabeleça também critérios para definir quando o jogador ganha.

A seguir são descritos mais detalhadamente como devem funcionar e serem implementado a nave e os asteroides no seu jogo

### Nave

A nave deve ter a possibilidade de girar em torno do seu eixo e avançar na direção que o canhão apontar, e pode ser representada por um triângulo, uma imagem e qualquer coisa que possa ser considerada uma nave. A interação para o giro e avanço pode ser feita tanto com teclado e/ou com o mouse e os conhecimentos necessário para implementar esses requisitos serão abordados na disciplina de **Geometria**.

O canhão da nave pode disparar vários tiros por vez e o tiro deverá sair na direção que o canhão aponta, quando o tiro ultrapassar a borda da janela ele deve sumir. O armazenamento dos vários tiros deve ser feito utilizando uma **lista de tiros**.

### Asteroides

As informações dos asteroides: centro ( $x, y$ ), raio e cor, deverão ser lidos de um arquivo de configuração, e deverá ser possível informar vários arquivos de configuração. Os asteroides lidos serão e armazenados em uma **lista de asteroides**.

Quando houver colisão entre o tiro e o asteroide, este deve se dividir em pelo menos dois pedaços menores, e caso os asteroides menores sejam atingidos, finalmente eles somem da tela.

### Entrada

A configuração do jogo é feita através de um **arquivo**, e o seu jogo deve estar preparado para se adaptar a essas diversas configurações. O nome de arquivo de configuração deve ser informado por **linha de comando** e o layout do arquivo pode seguir o exemplo abaixo:

#### Formato do arquivo de entrada

```
janela 400 600
passo_simulacao 100
vidas 5
asteroide 100 100 50 ff0000
asteroide 150 150 30 80eca7
asteroide 300 200 20 fff600
asteroide 50 50 35 fff6ab
asteroide 0 0 0
```

#### I – Linha janela

A linha janela indica o tamanho da janela do jogo, a primeira medida é a largura da janela e a segunda é a altura.

#### II - Linha passo\_simulação

Delay (em milissegundos) que será usado para simular o FPS (frame rate second);

#### III - Linha vida

Quantidade de vidas do jogador

#### IV - Linhas asteroide

As linhas a seguir representam os asteróides e não tem o limite de quantidade podendo variar de 0 até várias, e cada linha temos o seguinte formato (asteroide x y raio cor) sendo que :

- $x$  e  $y$  -> Posição inicial do asteróide, se for um círculo o  $x$  e  $y$  seriam o centro onde ele deve ser desenhado, lembrem-se de que  $y$  começa no canto superior da tela e cresce para baixo;

- `raio` -> raio do círculo usado para representar os asteróides;
- `cor` - cor em hexadecimal (base 16), cada par de caracteres representa os valores R, G e B (cada 2 caracteres em hexa codificam um valor de cor)

A lista de asteroides é finalizada com a linha com os valores **asteroide 0 0 0**

### Restrições do projeto

O programa deve estar bem documentado e implementado na linguagem Python com utilização da biblioteca PygGame. A entrega consistirá em uma apresentação do projeto, por todos os integrantes do grupo, a uma banca de professores das disciplinas envolvidas no projeto, a data da banca será definida posteriormente.

É fortemente aconselhável a utilização de **vetores** (vistos em **Geometria**) para calcular direções de tiro e movimentação dos asteroides, isto irá ajudar na eficiência do projeto. O trabalho será avaliado de acordo com os critérios abaixo:

- Funcionamento do programa;
- Atendimento da descrição realizada acima;
- Modularização do programa utilizando funções e clareza na nomenclatura de variáveis e funções;
- Clareza na(s) estrutura(s) de dados usada.

Este projeto pode ser desenvolvido em grupos de **até 4 alunos**, e é claro o seu grupo pode “*discutir*” com outros **grupos** “*dicas*” para chegar às soluções de problemas de desenvolvimento, mas seu grupo deve ser responsável pela solução final do jogo, ou seja, qualquer tentativa de fraude será punida com a **nota zero**.