



ADDETC – Área Departamental de Engenharia Eletrónica e Telecomunicações
e de Computadores

LEIM -Licenciatura Engenharia informática e multimédia

Modelação e Simulação de Sistemas Naturais

Projeto Final

Turma:

LEIM-32D

Trabalho realizado por:

Miguel Silvestre N°45101

Miguel Távora N°45102

Pedro Dias N°45170

Docente:

Arnaldo Abrantes

Data de entrega:19/01/2020

Índice

1. INTRODUÇÃO.....	3
2. DEFINIÇÃO DA NARRATIVA	4
3. RESULTADOS OBTIDOS.....	5
3.1 INTERFACE.....	6
3.2 CARROS	7
3.3 VACAS.....	8
3.4 ESTAÇÕES	9
3.5 CARRO DOS BOMBEIROS	10
3.6 SISTEMA DE PARTICULAS EXPLUSÕES	11
4. DISCUSSÃO DOS RESULTADOS	12
5. DIAGRAMAS UML	13
6. CONCLUSÃO	15

Índice de Figuras

Figura 1-Menu	6
Figura 2-Settings	6
Figura 3 - Posições de nascimento dos boids	7
Figura 4 - Boids "vaca" implementadas.....	8
Figura 5 - Boids "vaca" com imagem.....	8
Figura 6-Primavera.....	9
Figura 7 - Verão	9
Figura 8 - Outono.....	9
Figura 9 - Inverno.....	9
Figura 10 - Carro dos bombeiros a perseguir a explosão.....	10
Figura 11 - Sistema de particulas de colisões.....	11
Figura 12 - Sistema de particulas de despiste.....	11
Figura 13 - Diagrama Classes de Boids.....	13
Figura 14 - Diagrama de atividades paralelas	14

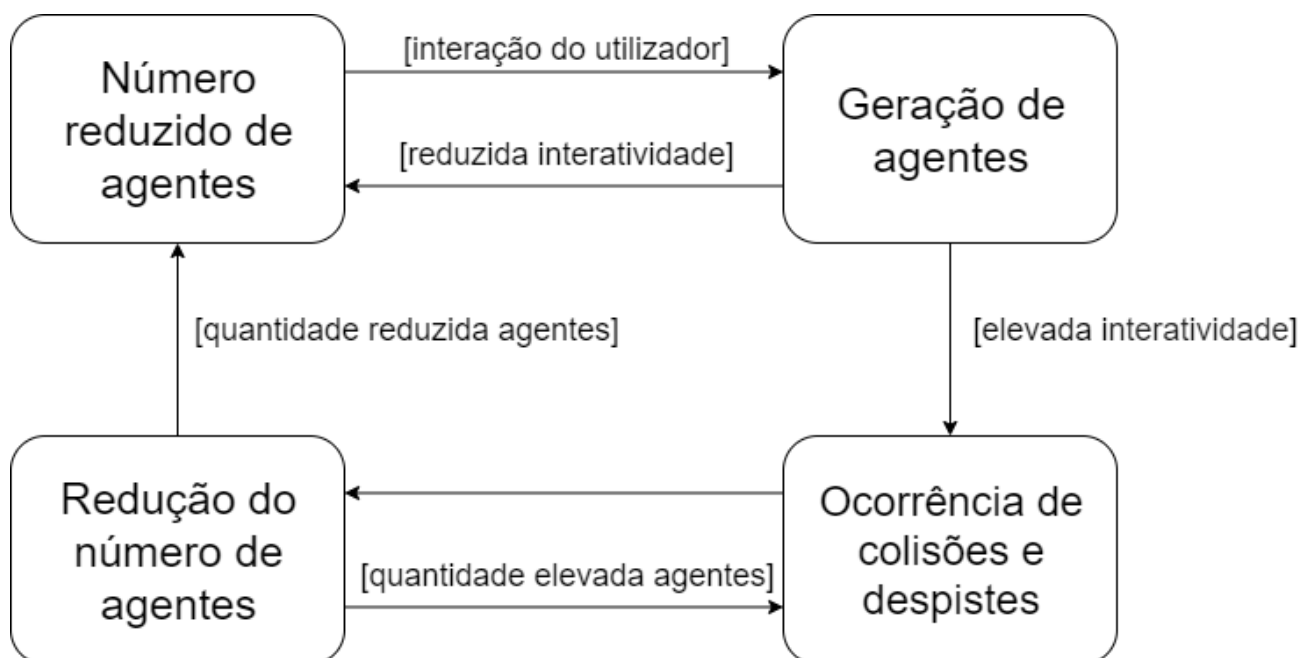
1. Introdução

No contexto da cadeira de MSSN surge o projeto final. Este projeto tem como principal objetivo o desenvolvimento de um ecossistema a partir dos conhecimentos adquiridos ao longo do semestre, no qual o tema é a escolha do grupo. Neste contexto o grupo decidiu de imediato uma simulação em vez de jogo, visto que o grupo prefere visualizar quais os resultados ao fim de algumas iterações sobre um grupo de indivíduos. Para isso era fundamental a utilização de sistemas autónomos nomeadamente Boids e também sistemas de partículas. Os sistemas de partículas surgem para dar alguma interactividade visual ao utilizador aquando da visualização dos resultados.

O surgimento da ideia que acabou por ser implementada decorreu durante o semestre numa aula prática sobre agente autónomos. No qual o docente reduziu a visibilidade nos agentes autónomos e o comportamento dos agentes autónomos passou a ser igual ao de uma auto-estrada. Foi então que o grupo se lembrou de implementar algo relacionado com uma estrada. A partir dessa mesma ideia surgiu todo um processo de desenvolvimento do projeto com base nessa ideia, acabando por escolher um cruzamento onde os boids são carros e cada boid tem o seu próprio trajeto.

2. Definição da Narrativa

A narrativa conta a história de o que acontece frequentemente na vida de quem anda muito de carro. Um dos grandes factores responsáveis pelos acidentes rodoviários são em cruzamentos e também em ultrapassagens. Um outro factor que facilita o número de acidentes é o número de tráfego. Neste contexto surge o trabalho no qual simula um cruzamento de três estradas no qual os utilizadores passam para um caminho sem olhar para os carros que circulam. Este comportamento ocorrem bem quando há uma quantidade reduzida de tráfego, porém com um grande número de tráfego isto provoca diversos acidentes.



No presente diagrama é possível visualizar a ideia por de trás da forma como evolui o programa ao longo do decorrer do tempo. Pelo diagrama é possível visualizar que o ecossistema tende sempre a ir para a atividade de número reduzido de agentes. Ou seja, ele evolui de acordo com as leis do universo que tende sempre para a estabilidade.

3. Resultados obtidos

Os componentes do projeto dividem-se no comportamento principal e alguns comportamentos que acontecem em paralelo. Sendo os comportamentos principais da narrativa os boids a:

- Sair da pista, saindo da pista são eliminados.
- Colidir com outros boids, na colisão são eliminados.
- Sempre que chegam ao fim da pista são destruídos e gerados novos no seu lugar.
- Em algumas colisões surge um boid que são os bombeiros que vai até ao local da colisão onde ocorreu a colisão e depois é destruído.
- Na colisão surge um sistema de partículas.
- No despiste surge um outro sistema de partículas diferente, mas parecido visualmente ao sistema de partículas das colisões.

Em paralelo com os boids temos:

- Um sistema de estações do ano que vai mudando a imagem de fundo conforme a evolução no tempo.
- Boids que representam vacas espalhados pelas zonas sem estrada que andam pela erva aleatoriamente.
- Menu com interatividade com o utilizador.
- Música nas settings caso o utilizador assim pretenda.

3.1 Interface

Para melhorar a interatividade do utilizador com o a aplicação foi criado um menu que quando o rato está sobre os botões estes mudam de cor. Foi também criado uma opção de *settings* que permite meter música caso o utilizador assim o desejar. Para isso basta clicar nas settings clicar no botão [+] e clicar no botão OK. Caso já não desejar música basta clicar no [-] e clicar também OK. As interfaces gráficas têm o aspeto que se segue:



Figura 1-Menu

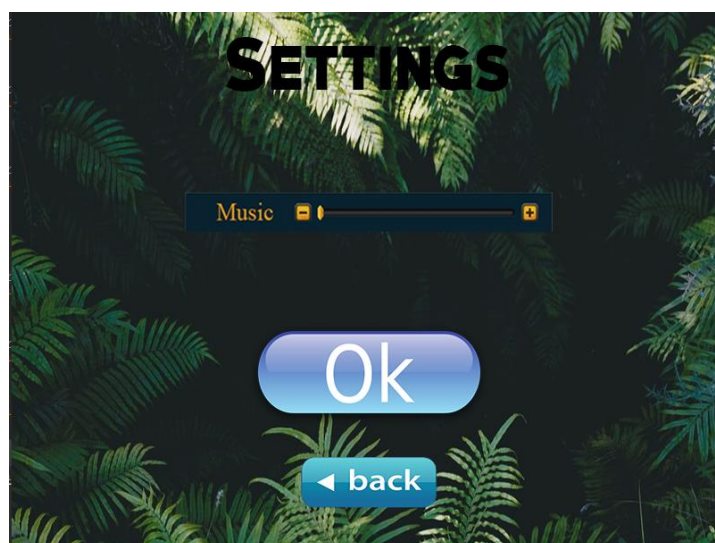


Figura 2-Settings

3.2 Carros

Cada carro é um boid guardado num ArrayList, e cada local onde pode nascer carros novos é um ArrayList diferente. Tendo portanto apenas para os carros cinco ArrayLists diferentes. Para eles fazerem a perseguição dos pontos foi utilizado o método seek. Para não estar constantemente a haver colisões entre os boids foram utilizadas as forças de alinhamento e separação em certos troços da pista. Para detectar colisões entre eles percorre-se as listas e compara-se as posições dos boids com as posições de outros boids caso a diferença de distância seja muito pequena é porque houve colisão são ambos os boids destruídos. Para a detecção de um boid estar fora da pista verifica-se se a posição do mesmo, caso seja maior ou menos que um Pvector em X ou Y conforme a estrada é eliminado.



Figura 3 - Posições de nascimento dos boids

3.3 Vacas

As vacas são boids que estão localizados nas zonas sem estrada. Estas possuem um comportamento simples, estas vagueiam pelas zonas onde não existe estrada. Foi implementado este comportamento em semelhanças ao comportamento das vacas que ficam no prado a pastar.

Era pretendido que as vacas fossem boids com imagens de vacas porém quando foi implementada essa metodologia o programa ficava lento e as vacas comportavam-se de forma totalmente aleatória, como se pode ver na figura 5 elas iam para a estrada e o seu movimento era muito pouco fluído.



Figura 4 - Boids "vaca" implementadas

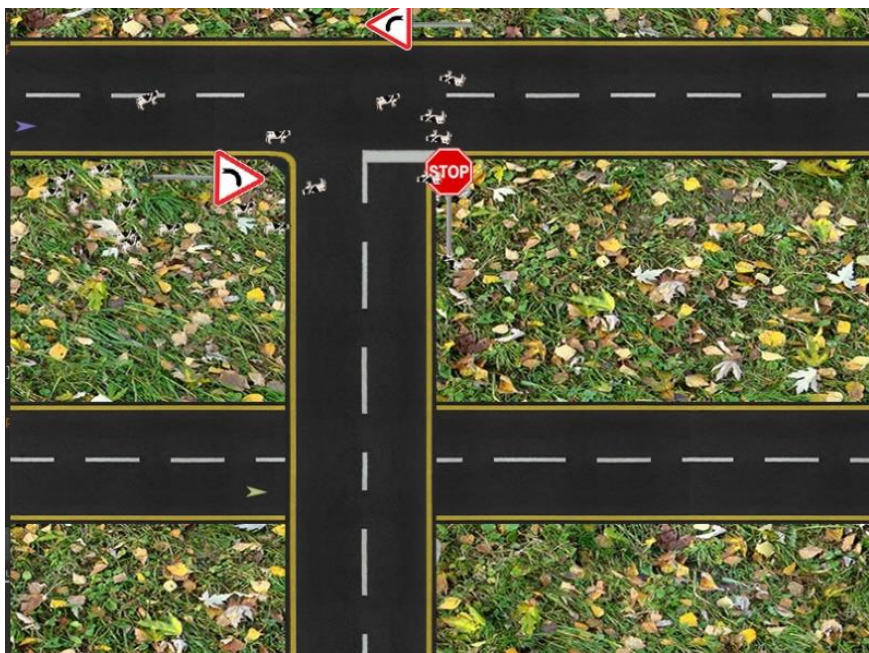


Figura 5 - Boids "vaca" com imagem

3.4 Estações

Foi implementado um sistema que vai mudando a estação do ano conforme o tempo. Possuindo assim primavera, verão, outono e inverno. Para a transição de estação foi criada uma animação que escurece o programa e quando está mais escuro muda a imagem. Esta animação foi feita a custa de um retângulo preto do tamanho da janela que vai aumentando a opacidade e depois reduz. Foi utilizada também um sistema de partículas que é inverno simula a neve e só é mostrada nessa altura do ano.

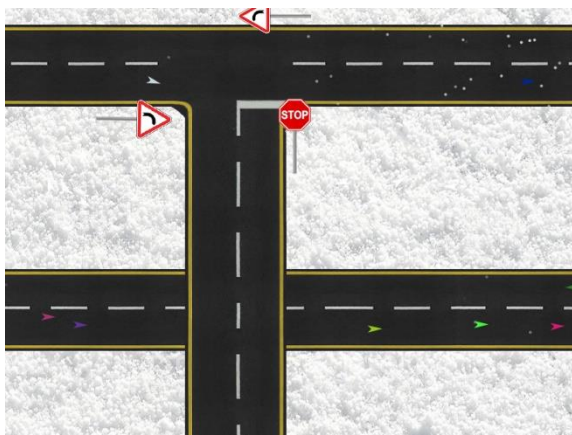


Figura 9 - Inverno

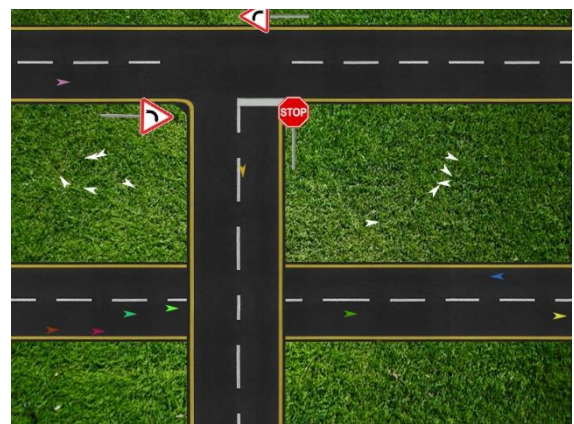


Figura 6-Primavera

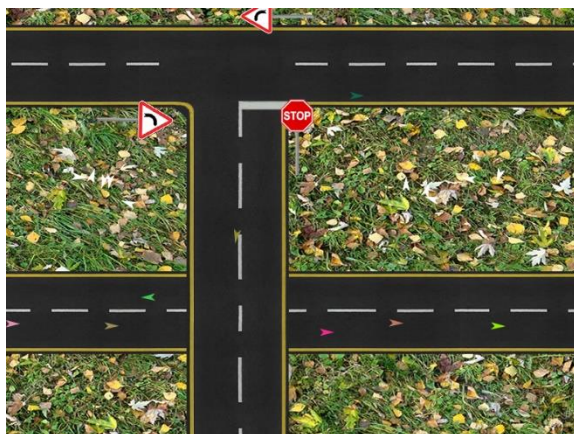


Figura 8 - Outono



Figura 7 - Verão

3.5 Carro Dos Bombeiros

Os carros dos bombeiros são boids semelhantes aos carros, mas que só aparecem quando ocorre acidentes. Este comportamento foi feito a partir do comportamento de colisão onde se guarda o local da colisão num ArrayList de PVectors caso o acidente seja numa das duas retas é criado um carro dos bombeiros. Os bombeiros só surgem uma vez, ou seja, só é criado um carro dos bombeiros. Caso haja mais do que uma explosão num mesmo dado instante onde já tenha sido criado um carro dos bombeiros não é criado um segundo carro dos bombeiros. O carro dos bombeiros não colide com os diferentes boids que representam carros.

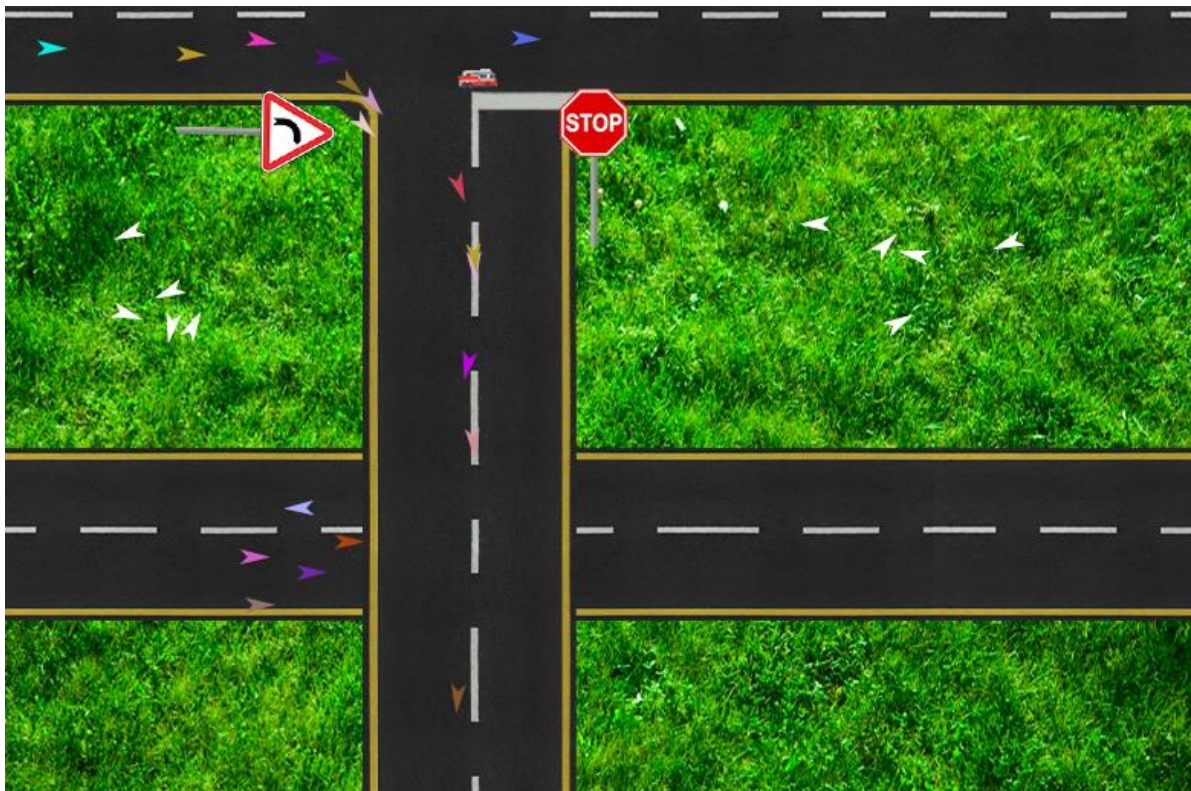


Figura 10 - Carro dos bombeiros a perseguir a explosão

3.6 Sistema de partículas explosões

Para criar uma explosão a partir do sistema de partículas de explosão pesquisou-se o método de criar e pintar o fogo. O grupo por sua vez fez toda a implementação da adaptação dos métodos para um sistema de partículas que funcione com o trabalho, nomeadamente trabalhe com as posições do SubPlot e feito a partir de PVectors. Para saber a localização onde deve ocorrer a explosão é semelhante ao carro dos bombeiros onde se guarda o local da colisão ou despiste e de seguida é criado uma instância da classe e é chamado o método que pinta o fogo.



Figura 11 - Sistema de partículas de colisões



Figura 12 - Sistema de partículas de despiste

4. Discussão dos resultados

Perante os resultados obtidos o grupo está satisfeito com o resultado obtido do projeto final.

O trabalho no ponto de vista da implementação foi complexo devido á numerosa manipulação de ArrayLists simultâneamente e dos processos paralelos. Sobre a parte estética o grupo tentou ao máximo embelezar o trabalho com cores, fundos sugestivos, interface interativa e sistemas de particulas diversos para melhorar a experiência do utilizador. Este contudo surge com alguns problemas devido á numerosa utilização de ArrayLists, onde os índices surgem como principal problema. Caso haja mais do que três colisões simultâneas a explosão surgirá num local onde não deveria, porém a probabilidade de ocorrência desse caso é muito pouco provável, daí a não ser um grande problema no projeto. Um outro problema é o numeroso número de ciclos durante o processo, onde o seu impacto foi reduzido pela utilização de for each pelo grupo.

Existe contúdo algumas ideias que não foram implementadas por falta tempo, nomeadamente ter plantas espalhadas pelas zonas sem vacas e sem estrada. As plantas evoluem conforme a estação do ano e reduzem a evolução em algumas estações. Transformar os boids em imagens com carros, esta ideia foi implementada porém com muitos problemas e por isso foi abandonada. Na estação do verão as vacas reproduzem-se e na altura do inverno algumas morrem, para simular em parte a evolução de um ecossistema e a lei de Darwin.

5. Diagramas UML

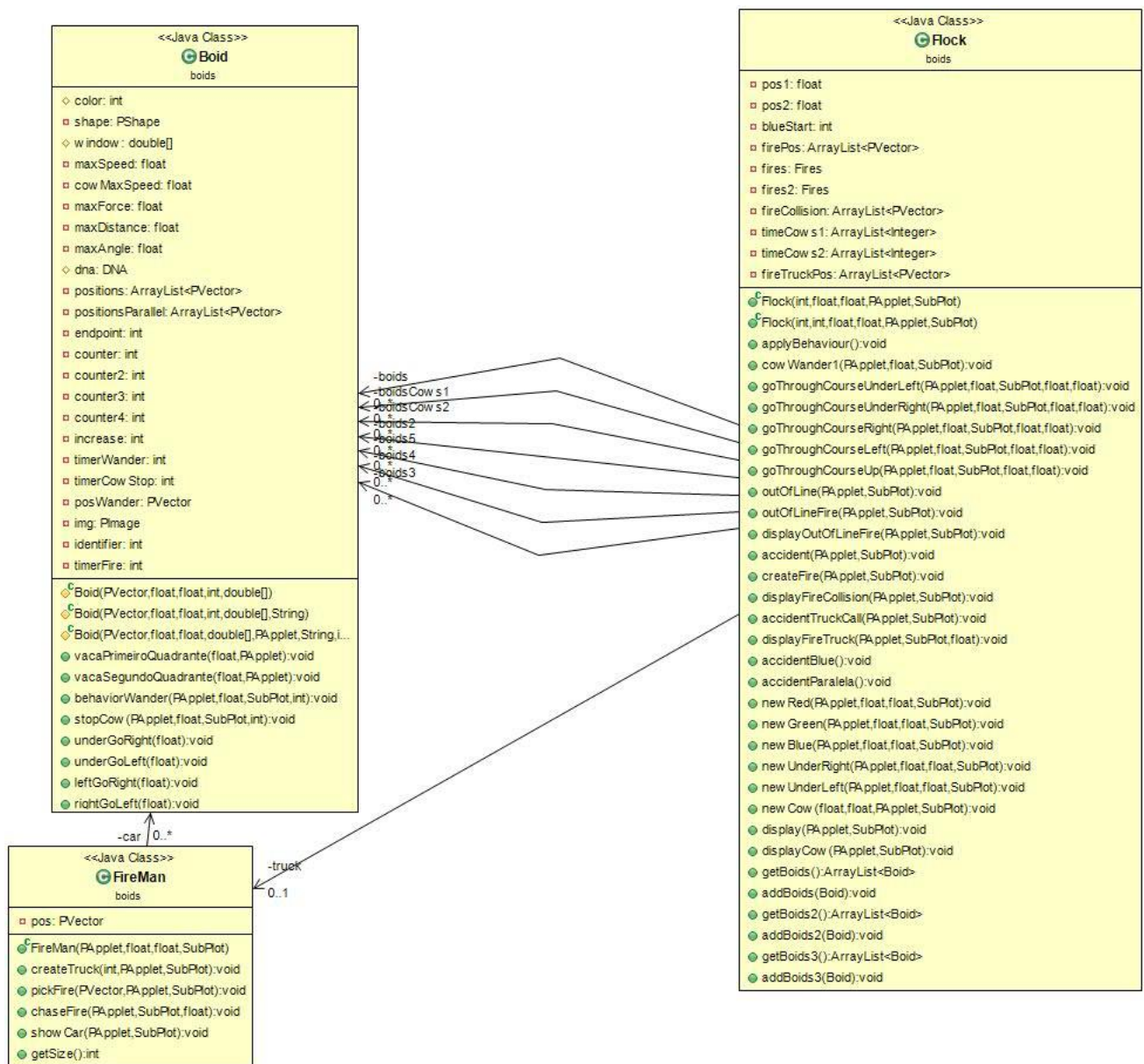


Figura 13 - Diagrama Classes de Boids

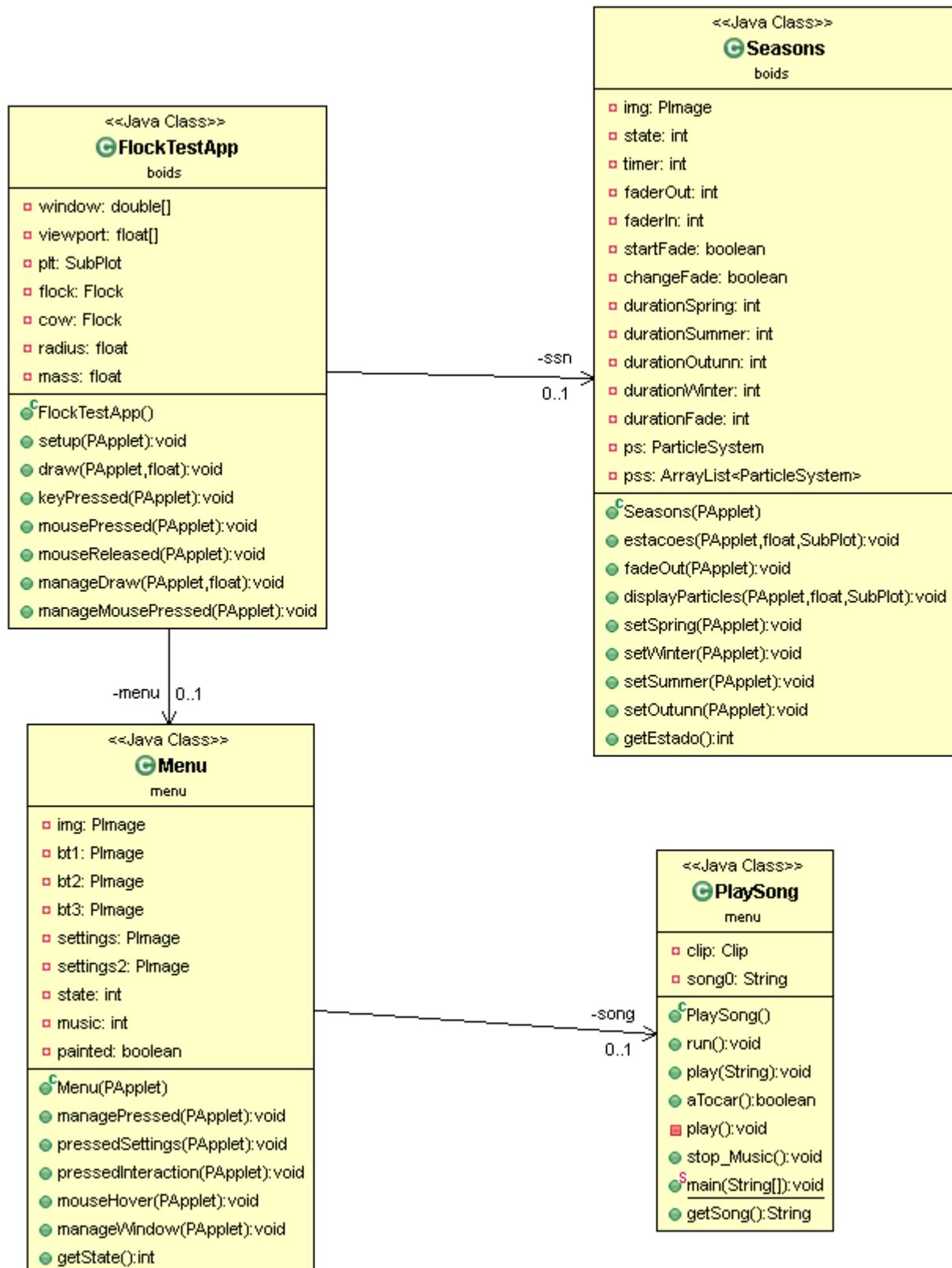


Figura 14 - Diagrama de atividades paralelas

6. Conclusão

O projeto permitiu algo nunca antes realizado em nenhuma cadeira que foi a manipulação de diversos ArrayLists para comportamento de agentes. Numa perspetiva global o projeto enriqueceu muito também na área de trabalhar com diversas classes, objetos, partilha do mesmo objeto por todas as classes, em geral programação orientada a objetos. No que toca á utilização de agentes autónomos houve um enorme progresso no aperfeiçoamento da sua utilização quando comparado ao tpc4. O projeto o facto de ser um tema livre acho que facilita para os alunos que possuem muita imaginação, e que gostam de criar o trabalho ao seu gosto.

Relativamente á cadeira, o que o grupo gostou mais de utilizar foi a biblioteca com o *processing*, visto a ser uma biblioteca fácil de usar e onde é possível fazer todo o tipo de coisas gráficas. Na cadeira propriamente dita as matérias são super interessantes visto que com poucas linhas de código é possível meter muitas coisas a mecher gráficamente. Em geral na perspetiva do grupo é uma cadeira interessante para quem gosta de trabalhar com partes gráficas e para quem gosta de programar para as mesmas.