

Programação Concorrente Orientada Por Objetos

Final Practical Assignment

Thief Catch

João Coelho, 80335
jcoelho@ua.pt

1- Apresentação do tema

Optei por escolher um tema para o meu projeto, que me agradasse a desenvolver. Consiste na simulação de um ambiente onde existem polícias e ladrões.

Os ladrões são entidades que começam no seu esconderijo e que saem dele para procurar objetos de valor. Esses objetos são trazidos para o seu esconderijo se, no caminho para o esconderijo, não forem apanhados pelos polícias.

Estes objetos de valor encontram-se em lojas na cidade, que ativam o alarme e comunicam com a central de informação que um assalto decorreu.

Ainda estou a planear como será comunicada a localização do ladrão após o assalto. As opções que já pensei, são um helicóptero ver e comunicar a sua localização de x em x tempo ou então, ter câmeras de vigilância espalhadas pela cidade que são acionadas quando um ladrão passa por elas.

O papel dos polícias é estar na esquadra preparados para o caso de haver um roubo. Nesse caso, os polícias saem da sua esquadra à procura do ladrão. Se o conseguirem encontrar antes de ele chegar ao esconderijo, este é trazido para a esquadra.

2- Entidades

2.1- Entidades ativas

Neste projeto existem as seguintes entidades ativas:

- Ladrão: indivíduo que sai do seu esconderijo à procura do objecto a roubar. Assim que o encontra, volta ao esconderijo para largar o objeto roubado;
- Polícia: os polícias começam na esquadra e permanecem parados até que sejam informados que houve um roubo. Assim que sejam informados, estes saem à procura do ladrão. Caso o encontrem, trazem-no para a esquadra para ser detido;
- Helicóptero: dá a localização dos ladrões à central de informação para que os polícias os consigam apanhar.

2.2- Entidades passivas

Neste projeto existe a seguinte entidade passiva:

- Store: comunica às autoridades quando um objeto é roubado.

2.3- Regiões partilhadas

Neste projeto existem as seguintes regiões partilhadas:

- Map: O mapa da cidade onde todas as entidades ativas se movimentam;
- PathFinder: uma região partilhada entre todas as entidades. Esta classe é responsável por dar as direções de um dado lugar x para outro y. Por exemplo, polícias vão para o último lugar visto do ladrão. Ou, caminho de volta do ladrão para o esconderijo depois de ter roubado o objeto.

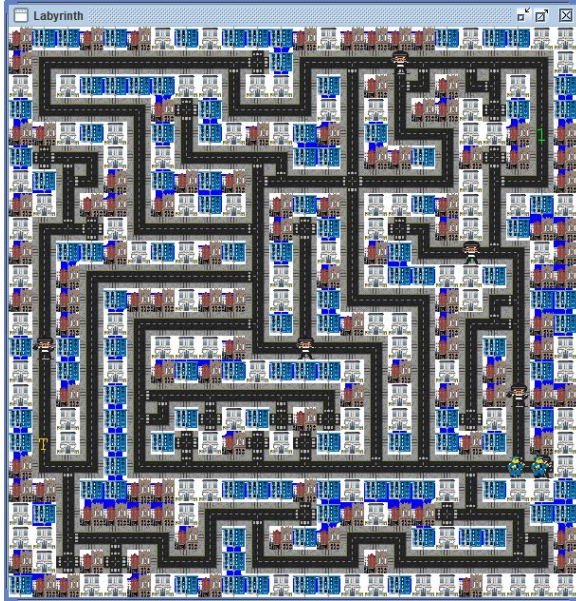
- Interpol: Centro de informações sobre as posições atuais das entidades ativas relevantes no mapa e interações entre elas.

3- Guia de utilização

Decidi desenvolver o projeto sem recurso a argumentos e todas as configurações são feitas manual e estaticamente na main CityRun.java (embora não seja difícil) pois foi sempre testado em ambiente NetBeans.

Pode alterar-se as coordenadas dos pontos (lojas, prisão e esconderijo) bem como o número de polícias e ladrões.

Basta importar como projeto de netbeans e correr a simulação na main CityRun.java.

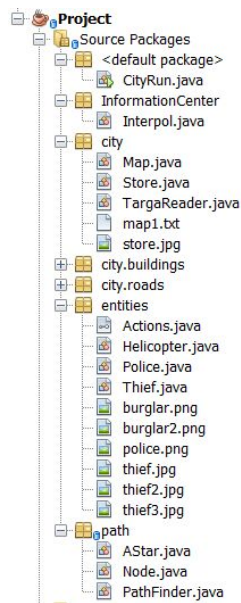


Ao correr a simulação, os ladrões começam o seu ciclo e eventualmente irão assaltar a loja. Após o assalto, os polícias vão acordar e tentar apanhar os ladrões.

4- Estrutura

O projeto é estruturado por 4 packages:

1. InformationCenter - Objeto partilhado pelas entidades.
2. city - Criação do mapa (TargaReader para ler imagens do tipo .tga), entidade passiva e todas as imagens usadas para o mapa.
3. entities - Entidades ativas do sistema.
4. path: - Classes usadas para encontrar o caminho através do algoritmo A*.



5- Decisões de concorrência

O projecto está separado em três conjuntos. As entidades activas, uma entidade passiva e um objeto partilhado. A classe main instancia o labirinto e este é passado às entidades ativas principais: ladrões e polícias.

O ciclo de vida do helicóptero é reportar as posições dos ladrões de 5 em 5 segundos depois de haver um assalto até que estejam todos ou presos ou seguros.

O ladrão, inicialmente, anda ao acaso pelo mapa durante algum tempo. Depois, decide ir roubar uma das lojas e vai até lá usando o PathFinder (caminho mais curto através do A star). Após algum tempo dentro da loja, o assalto é concluído e o alarme da loja toca. A loja comunica, então, com a central de informações da sua posição. O ladrão tenta voltar até ao seu esconderijo, novamente pelo caminho mais rápido. Se entretanto se cruzar com um polícia, o polícia “apanha-o” e vão os dois para a prisão usando o PathFinder.

O polícia começa a simulação permanecendo na prisão (wait) até que seja notificado que um assalto ocorreu. Quando acontecer, usando o PathFinder vai para o sítio onde foi visto o ladrão. Depois de chegar ao ponto onde foi visto o ladrão, o polícia vai à próxima localização onde foi visto pela última vez pelo helicóptero. Se se cruzar com um ladrão, vão os dois para a prisão.

No Interpol (central de informações) estão algumas informações como a posição atual do ladrão, se o ladrão foi encontrado, se o polícia encontrou o ladrão ou ladrão está na prisão. Quase todas estas informações são acedidas em exclusão mútua (synchronized).

6- Problemas e as suas soluções

Ao longo do trabalho desenvolvido, ocorreram situações de deadlock e na falta de sincronização das entidades ativas. Contudo, depois de algumas tentativas error, ponderar e fazer desenhos, a melhor solução foi criada de modo a não haver problemas de deadlock entre as entidades.

Um problema corrente, é o facto de as entidades parecer que se cruzam mas na verdade, nesse mesmo instante, estão em celas separadas e é por isso que o polícia não leva esse

ladrão para a prisão. Foi solucionado eventualmente, mas levantou outros problemas mais graves pelo que ainda não há solução existente.

7- Possibilidades de uma futura expansão

No futuro esta simulação pode ser melhorada contendo um ambiente ainda mais realista onde existam vários tipos de assaltos e vários tipos de polícias. Um assalto pode resultar em feridos e a necessidade de chamar uma ambulância de urgência para levar o paciente para o hospital.

O mapa também pode ser melhorado e maior.