



Universidade de Coimbra
Faculdade de Ciências e Tecnologia
Departamento de Engenharia Informática

Introdução à Inteligência Artificial
2008/2009 - 2º Semestre

Trabalho Prático 1: *Reactive Pac-Man*.

Data Limite de Entrega: 17/4/2009 – 17 Horas

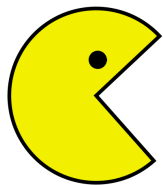
Nota 1: É obrigatório indicar no relatório **por cada aluno**:

- tempo de estudo
- tempo de implementação
- contributo para o trabalho

A omissão destes elementos implica a não consideração do trabalho prático para efeitos de avaliação.

Nota 2: A fraude denota uma grave falta de ética e constitui um comportamento inadmissível num estudante do ensino superior e futuro profissional licenciado. Qualquer tentativa de fraude levará à anulação da componente prática tanto do facilitador como do prevaricador, independentemente de acções disciplinares adicionais a que haja lugar nos termos da legislação em vigor. Caso haja recurso a material não original as **fontes** devem estar explicitamente indicadas.

1 Introdução



O Pac-Man. é uma dos jogos electrónicos mais emblemáticos da década de 80. Criado por Tohru Iwatani, designer de jogos da Namco, destaca-se dos restantes jogos da época quebrando com a hegemonia do estilo “shoot-em-up”. A idéia do desenho original ocorreu durante um jantar, é inspirada numa pizza à qual tinha sido retirada uma fatia, fazendo recordar uma boca aberta. Na versão original Japonesa o jogo chama-se Pakkuman, um nome que derivado da expressão Japonês “pakku-pakku” usada para descrever o barulho da boca de alguém a abrir-se e a fechar-se enquanto come. Em 1980, a Bally comprou os direitos nos Estados Unidos e o nome foi alterado para Pac-Man. O jogo tornou-se famoso de imediato atingindo uma popularidade invulgar. No ano de estreia foram vendidas cerca de 100.000 máquinas de arcades do Pac-Man. Mais tarde, foram produzidas várias sequelas e versões para inúmeras plataformas.

A mecânica do jogo é simples: o jogador controla o personagem Pac-Man posicionado num labirinto simples repleto de comida, cápsulas e 4 fantasmas que o perseguem. O objetivo era comer todas as pastilhas sem ser alcançado pelos fantasmas. Comendo uma cápsula o Pac-Man assusta temporariamente os fantasmas, podendo atacá-los.



Figura 1: A máquina arcade original e labirinto.

2 Objectivos Genéricos

Em termos genéricos pretendem-se alcançar os seguintes objectivos:

1. Aprender a desenhar um agente reactivo adequado a um problema e ambiente específico
2. Aprender a especificar e formalizar os seguintes aspectos de um agente reactivo:
 - (a) Sensores
 - (b) Acções
 - (c) Sistema de produções
 - (d) Memória
3. Aprender a fazer uma descrição de alto-nível do comportamento desejado implementável através de um agente reactivo.
4. Aprender a criar um agente reactivo a partir de uma descrição de alto nível.

Estes objectivos genéricos serão alcançados através do trabalho em grupo e da experimentação, promovendo-se, assim, estas capacidades.

3 Enunciado

No presente trabalho prático irá desenvolver um agente reactivo para controlar o Pac-Man. Para o efeito utilizará as bibliotecas disponibilizadas que implementam a mecânica do jogo, ambiente gráfico, e outras funcionalidades necessárias para que se possa concentrar na tarefa em questão.

O objectivo do seu agente reactivo é semelhante ao de um jogador humano, i.e. ganhar o jogo. isto implica, sobreviver, evitar fantasmas, comer comida e cápsulas, etc.

Tem liberdade para especificar e implementar comportamentos, desde que estes sejam adequado ao objectivo genérico acima definido. Deverá ainda ter em conta as seguintes restrições:

Sensores – Ao contrário de um jogador humano que vê todo o tabuleiro o seu agente reactivo está inserido no ambiente do jogo. Isto é percepção o mundo da mesma forma que o Pac-Man o poderia perceber, nomeadamente: não pode ver através de paredes; os sensores de “visão” não dobram esquinas; não adivinha as posições dos fantasmas, cápsulas ou comida que estejam fora do seu campo de visão, etc.

Memória – A memória do agente está limitada a 20 bytes (ou equivalente)

Acções – As acções do Pac-Man estão limitadas pelo motor do jogo. Não pode realizar acções que não estariam disponíveis a um jogador humano, exemplos: saltar, tele-transportar-se, atravessar paredes, etc.

4 Desenvolvimento

Deve seguir os seguintes passos no desenvolvimento deste trabalho prático:

1. Modelização e Design – Note que esta é uma etapa fundamental deste trabalho tanto por motivos de natureza teórica como prática.
 - (a) Especificar os **sensores** que vai utilizar e os dados que estes devolvem
 - (b) Especificar que dados vão ser armazenados em **memória**
 - (c) Especificar o **comportamento de alto nível** que deseja implementar
2. Desenvolvimento e Implementação
 - (a) Construir o **Sistema de Produções** que implementa o comportamento desejado
 - (b) Implementar o Sistema de Produções
3. Teste e Análise – Deve testar o seu agente de forma exaustiva por forma a identificar as suas forças, fraquezas e desvios em relação ao comportamento desejado. Caso necessário, este passo poderá levar à reformulação de etapas anteriores. Para melhor avaliar o desempenho do seu agente, deve testar o seu comportamento não só no labirinto *default* como em outros labirintos.

NOTA: É importante validar o trabalho desenvolvido no primeiro passo antes de passar à fase de desenvolvimento e implementação. Para o efeito, deve apresentar este trabalho aos docentes **durante as aulas práticas laboratoriais**.

5 O Simulador

Será utilizada uma variante do simulador de Pac-Man maioritariamente desenvolvido por John DeNero na universidade de Berkley (<http://inst.eecs.berkeley.edu/~cs188/sp09/pacman.html>). Este simulador é constituído por uma série de bibliotecas Python que implementam todas as funcionalidades necessárias, permitindo que se centre no desenvolvimento do agente reactivo. Deve fazer **download da versão disponível no WOC**, visto que esta se encontra preparada para este trabalho prático.

Ficheiros que vai editar:

reactiveAgents.py – Deve editar a classe “ReactAgent” por forma a implementar o seu agente reactivo.

Note que este é o único ficheiro que pode alterar.

Ficheiros úteis para o desenvolvimento:

pacman.py – O ficheiro principal que corre o jogo. Descreve o tipo GameState que será usado neste projecto.

game.py – Implementa a lógica de funcionamento do mundo. Descreve vários tipos, incluindo AgentState, Agent, Direction, e Grid.

Ficheiros de suporte que pode (deve) ignorar:

graphicsDisplay.py – Gráficos para o Pacman

graphicsUtils.py – Biblioteca de suporte gráfico

textDisplay.py – Gráficos ASCII

ghostAgents.py – Agentes para controlar os fantasmas

keyboardAgents.py – Interface de teclado para controlar o Pacman

layout.py – Código para ler mapas e armazenar os seus conteúdos

5.1 Correr o Simulador

Seguem-se exemplos de diferentes formas de correr o simulador:

- `python pacman.py`
Corre o jogo em modo normal, o utilizador controla o Pac-Man através do teclado.
- `python pacman.py --pacman GoWestAgent`
Corre o jogo usando o agente “GoWestAgent” definido no ficheiro “reactiveAgents.py”.
- `python pacman.py --layout tinyMaze --pacman GoWestAgent`
Semelhante ao anterior mas usando o ambiente TinyMaze.
- `python pacman.py --pacman ReactAgent`
Corre o jogo usando o controlador desenvolvido por si. .

6 Relatório e Entrega

Num trabalho desta natureza o relatório assume um papel importante. Deve descrever de forma detalhada e clara:

- Os sensores utilizados e os dados que estes devolvem
- Os dados que são armazenados na memória
- O comportamento de alto nível esperado
- A forma como esse comportamento foi modelado através de um sistema de produções.

Posteriormente deve descrever a implementação do agente, dando particular destaque aos problemas e soluções encontradas.

A experimentação é uma parte essencial do desenvolvimento de aplicações de IA. Assim, deve descrever detalhadamente as experiências realizadas, analisar os resultados, extrair conclusões, e efectuar alterações (caso se justifique) em função dos resultados por forma a melhorar o desempenho do agente.

O relatório deve conter informação relevante tanto da perspectiva do utilizador como do programador. Não deve ultrapassar as 20 páginas, formato A4. Todas as opções tomadas deverão ser devidamente justificadas e explicadas.

A primeira secção do relatório é obrigatoriamente composta pela seguinte informação, para cada elemento do grupo:

- nome completo;
- e-mail;
- o tempo de estudo e de implementação gasto na elaboração do trabalho;
- as tarefas desempenhadas pelo elemento em causa.

Deverá ainda mencionar todos os detalhes relacionados com a divisão do trabalho pelos diferentes elementos do grupo que considere relevantes. O trabalho colaborativo é um aspecto fundamental da aprendizagem.

6.1 Modo de Entrega

Deve entregar na **secretaria do DEI** o seguinte material:

- Relatório impresso (max. 20 páginas)
- CD contendo:
 - O ficheiro “reactiveAgents.py” desenvolvido, devidamente comentados;
 - O ficheiro em formato pdf contendo o relatório

No acto de entrega deve inscrever-se num dos intervalos de tempo disponíveis para a defesa do trabalho prático.

Data Limite: 17 de Abril de 2009

Bibliografia

- **Inteligência Artificial: Fundamentos e Aplicações**
Ernesto Costa, Anabela Simões

Ernesto Costa, Penousal Machado, Alberto Moraglio – 2008/2009