



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE
COIMBRA



Introdução à Programação e Resolução de Problemas

2022/2023 - 1^o Semestre

Mini-Projeto:
Cobras e Comidas

Nota: 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

Cobra (do inglês *Snake*) é um jogo de computador em que o jogador controla uma criatura artificial cujo o objetivo é colecionar pedaços de comida que vão aparecendo no ecrã de forma aleatória. O jogador controla a cabeça da cobra, que se pode movimentar em 4 direções (Cima, Baixo, Esquerda, Direita), sendo que o resto do corpo acompanha o movimento da cabeça. Quando a cobra come um pedaço de comida, o seu corpo cresce. Por outro lado, sempre que a esta tocar com a cabeça nos limites da área de jogo ou no seu próprio corpo, o jogador perde e o jogo termina.

O jogo foi originalmente desenhado em 1976, mas ficou bastante popular no fim dos anos 90 com a sua inclusão nos telefones da Nokia. A Fig. 1 exemplifica o ambiente do jogo.

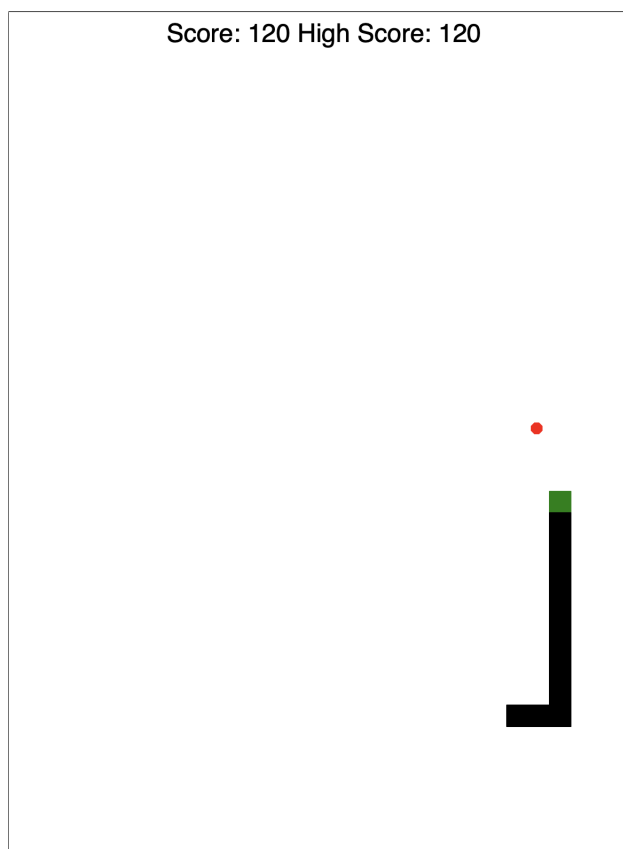


Figura 1: Exemplo do Jogo Snake. A cabeça da cobra é representada pelo quadrado verde, o ponto vermelho representa a comida, e os quadrados pretos correspondem ao corpo.

2 Enunciado

Este trabalho prático tem como objetivo principal a programação do jogo da cobra. Pretende-se que os estudantes consolidem as competências adquirida relacionadas com a resolução de problemas e programação em Python. Assim, os alunos terão de completar e desenvolver as funcionalidades básicas do jogo, nomeadamente:

1. Movimentar o cobra para Cima, Baixo, Esquerda, Direita usando o teclado;
2. Detetar colisões com os limites do ambiente;
3. Gerar comida em posições aleatórias;
4. Apanhar comida e fazer crescer o corpo da cobra;
5. Detetar colisões com o corpo;
6. Programar um sistema de pontuação - Sempre que um pedaço de comida é ingerido pela cobra o jogador ganha 10 pontos. Sempre que houver um *high-score*, este resultado deverá ser guardado em ficheiro. O ficheiro deverá conter todos *high-score*, mas o quadro de pontuação deverá exibir o resultado mais alto.

É fornecido um ficheiro com o nome `snake.py`, que os alunos deverão descarregar da UCTeacher. Este ficheiro já contém o código necessário para inicializar o ambiente de jogo, nomeadamente no que toca a criação da janela, a criação da cabeça da cobra, a criação do quadro de pontuação e deteção do carregar nas teclas `W,A,S,D`. As restantes funcionalidades deverão ser programadas pelos alunos, de forma a permitir ter um jogo completamente funcional. O ficheiro contém algumas funções definidas cujo código deverá ser completado. No entanto, isto não significa que sejam apenas necessárias as funções fornecidas. Assim, deverá criar todas as funções adicionais que ache necessárias para resolver o problema.

Relativamente às funções que estão definidas e que deverá completar, elas são as seguintes:

- `move(state)` - Função responsável pelo movimento da cobra no ambiente.
- `create_food(state)` - Função responsável pela criação da comida. Note que elas deverão ser colocadas em posições aleatórias, mas dentro dos limites do ambiente.

- `check_if_food_to_eat(state)` - Função responsável por verificar se a cobra tem uma peça de comida para comer. Deverá considerar que se a comida estiver a uma distância inferior a 15 pixels a cobra pode comer a peça de comida.
- `boundaries_collision(state)` - Função responsável por verificar se a cobra colidiu com os limites do ambiente. Sempre que isto acontecer a função deverá retornar o valor booleano `True`, caso contrário retorna `False`.
- `check_collisions(state)` - Função responsável por avaliar se há colisões. Atualmente apenas chama a função que verifica se há colisões com os limites do ambiente. No entanto deverá escrever o código para verificar quando é que a tartaruga choca com uma parede ou com o seu corpo.

Finalmente, e após terminar a programação de todas as funcionalidades pedidas, poderá ainda acrescentar novas funcionalidades que serão consideradas como bónus, permitindo compensar algumas perdas de pontuação.

3 Datas e Modo de Entrega

O trabalho deverá ser realizado maioritariamente durante as aulas e por grupos com uma dimensão máxima de 2 alunos. Irá existir uma defesa obrigatória, em que todos os elementos do grupo têm de estar presentes.

Modo de Entrega:

O trabalho deverá ser entregue eletronicamente através do Inforestudante.

Data Limite: 23h59 do dia 02 de Dezembro de 2022

Nuno Lourenço, Ivo Gonçalves, Tiago Baptista, Amílcar Cardoso, João Correia, Carlos Fonseca – 2022/2023