COMP 4106 - Artificial Intelligence
Winter 2015

Assignment #1

Due date: February 16, 2015

The Snake Game

Introduction

In this assignment you will be implementing a variation of the *snake game*. This game has been written for almost every platform from ancient UNIX systems to the latest mobile phones. In all likelihood you've played this game at some point in your life.

The game is played on a varying size grid board. The player controls a snake. The snake can move in four (4) directions. Also present on the board is "food" for the snake. The goal is for the snake to eat as many food items as possible without running into the wall or itself.
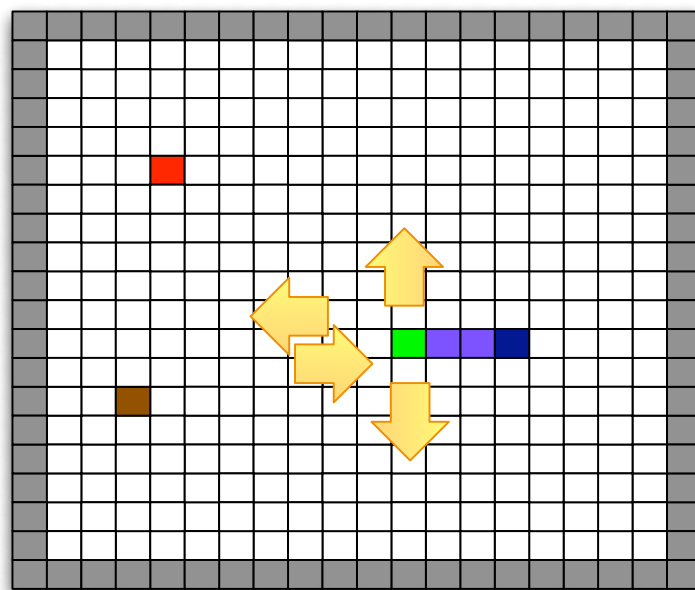
Game Definition

Figure 1: Snake game illustration

The above figure illustrates the game. The red box is the food for the snake. The green box is the head of the snake. The blue box is the tail of the snake. The purple boxes are the body of the snake. The snake can move in the directions indicated by the arrows. The board may contain impassable blocks, illustrated here in brown.

In this simplified version of the game, the snake is not constantly moving. This means that the snake is "still" (dormant) until it is told which direction to move in. The speed of the snake does not increase with time, either, and thus, the speed of the snake is constant.

When the snake eats a piece food its length does not change. The snake length remains constant, five (5) squares, for the duration of the game. Once the food item is eaten another one appears in a random location on the board. The snake then "knows" the location of the next food item.

The game is over if the head of the snake collides with the border of the room, itself or an impassable block.

The video at http://www.youtube.com/watch?v=z_Ct-1KwSgo will give you an idea of how the snake moves. If you watch the video, keep in mind the restrictions placed on the game in this assignment. The video was not made for this assignment and should only be used to get an idea of what the game is like.

Assignment Objectives

- Implement the *snake game*.

- Implement a *Breadth_First Search* for the snake to play the game.

- Implement *Depth_First search* for the snake to play the game.

- Implement *A* search*.

  - Implement two (2) different heuristics for the snake to play the game.
  - Implement a third heuristic which takes the average of the first two heuristics.

- Write a short report (no more than one (1) page) about the state space of the game, and about the choice of your heuristics. Please bring a hard copy of this report with your name and student number to your demo.

Questions

During the demo you should be prepared to discuss the following questions:

- Which search worked best?

- Which heuristics did you use?

- Why did you choose these heuristics?

- Does the combination of the two heuristics work better or worse than they do individually?

- How well do the searches work if you increase the size of the board to 30x30 or 50x50.

- Which of the searches work best when you introduce impassable blocks onto the board? To verify this, for example, you could cover 10% of the board with impassable blocks.

Bonus

The following items are considered as bonus. You should work on these if you have completed the required objectives.

- The length of the snake varies from 4 to 7 squares.

- Multiple snakes in the same room compete for the same food item.

Tips

Don't spend too much time on the graphics. A command line interface is perfectly acceptable, so long as it is readable. If the search is slow, think about how you can optimize it, but your first priority should be making sure that the search actually works.