

Modern Computer Games

COMP 521, Fall 2018

Assignment 4

Due date: Tuesday, December 4, 2018, by 6:00pm

Note: Unless otherwise specified, late assignments will only be accepted with prior **written** permission of the instructor. You must **explain** all answers and **show all work** to get full marks! Please make sure your code is in a professional style: **well-commented**, properly structured, and appropriate symbol names. Marks will be very generously deducted if not!

Description

In this assignment you will develop some procedural content within a Unity simulation. This involves developing behaviour for different kinds of agent, based on the principles described in class. All agents should avoid collisions with obstacles and other agents, and move using some combination of steering forces. Use the same vehicle model of all agents, with turning and acceleration forces capped lower than braking.

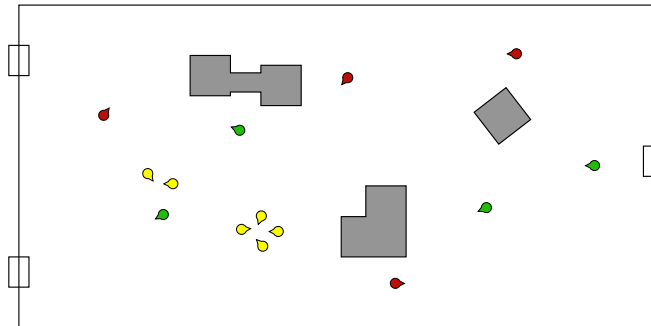
Do not use external assets that provide steering behaviours. You must implement all agent movement yourself.

Note that the simulation builds in complexity. Provide one solution and include clearly indicated options to demonstrate the different sub-solutions.

1. Build a game level following the overhead view of the design shown below. This consists of a large rectangular area with a doorway on the extreme right, and two doorways on the extreme left. Inside are $n \geq 2$ obstacles. **10**

The general shape and doorways of this level are fixed. Obstacles, however, are procedurally generated, and should be different on every playthrough. Each obstacle consists of at least 4, and at most 16 vertices, and should form simple polygons (potentially convex or non-convex). Shapes should be of varying size, roughly capable of blocking between 1/8 and 1/2 of the level width. Provide an input field in the editor for selecting n .

You must guarantee that the obstacles generated still allow for some paths (but not always the same path) for agents from the right doorway to both of the left doorways. At least for $n \leq 5$ it should always be possible to generate such a level without significant delay.



2. Travelling agents, shown in red, spawn from the rightmost doorway and attempt to get to one of the exit doorways on the left. They move at varying, but relatively fast speeds. The choice of exit each aims at should be set randomly at spawn time, but if they are unable to reach their intended exit doorway for an extended time they may change their mind. **15**

Note that while these agents need to know of a path to their destination, their movement should still be controlled by steering forces. It should be the case that in the absence of interference and with low obstacle density, traveller agents can usually reach their goals.

Once an agent reaches the goal it respawns to ensure a constant population. Provide an input field in the editor for selecting the number of travelling agents.

3. Some agents simply wander around the level, moving at varying but relatively fast speeds. These agents are shown above in green. **10**

A wanderer that is just wandering, and which comes close (within a radius of about a doorway width or two) to a traveller should attempt to interfere with the traveller's goal by interposing itself between the traveller and the traveller's intended exit.

Provide an input field in the editor for selecting the number of wandering agents.

4. Social agents, shown in yellow in the figure above also move randomly through the level. When one comes close enough to another social (yellow) agent, it may choose to enter into "conversation" with the agent, either forming or entering the resulting social group. **15**

Define appropriate steering forces that allow agents to form or enter social groupings. There is no upper bound on number of agents in a group. Agents may also leave a social group and return to wandering following a (randomized, 0.5s–2s) timeout. Determine an appropriate cooling off time, during which an agent will act purely as a wanderer (i.e., becomes green) and not attempt to enter a group to avoid agents re-entering the same group they just left.

Provide an input field in the editor for selecting the number of social agents.

5. What is the relative importance of the number of obstacles, number of wanderers, and number of social agents on whether traveller agents can reach a goal? Set the number of travellers to a relatively high number, and measure the number of travellers that reach the exit over a short time span. Find a setting of your other 3 parameters that results in 1/2 to 3/4 of the travellers spawned reaching their goal. Vary the parameters, one by one to determine their effect on traveller throughput. **10**

Provide a *separate document* that describes your settings, shows your data (as either tables or graphs), and includes a brief discussion to explain your results.

What to hand in

Assignments must be submitted on the due date **before 6pm**. Submit your assignment to *MyCourses*. Note that clock accuracy varies, and late assignments will not be accepted without a medical note: **do not wait until the last minute**.

For the Unity questions, hand in an exported project containing all files needed in order to reconstruct and run your simulations.

For non-Unity questions, submit either an ASCII text document or a .pdf file *with all fonts embedded*. Do not submit .doc or .docx files. Images (plots or scans) are acceptable in all common graphic file formats. You may submit a single document for such questions, as long as each answer is clearly delineated.