

Modern Computer Games

COMP 521, Fall 2018

Assignment 3

Due date: Wednesday, November 21, 2018, by 6:00pm

Note: 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 a Unity simulation of an AI agent.

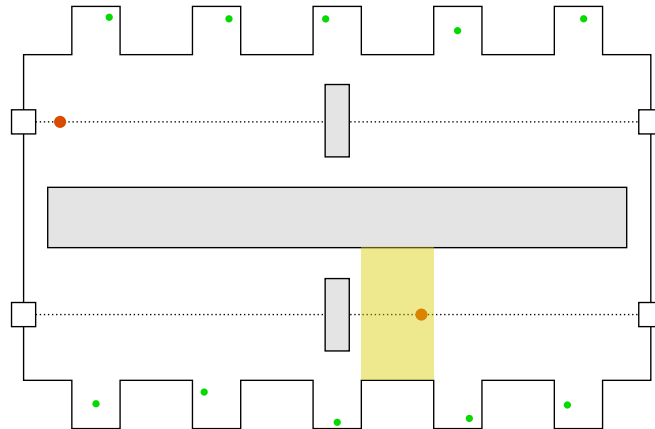
You must implement the decision making process (AI) yourself; *do not* use any built-in or external asset tools for AI development. You may, however, use any built-in features for performing basic navigation (pathfinding) to move from one location to another.

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 area, with a long central obstacle, 2 smaller obstacles above and below it, and 10 alcoves, 5 on each side. Two doorways are found on the left and two on the right. **10**

The layout is not necessarily to scale or in proper proportion. Build something that allows agents to generally move around freely, with the caveat that there is room for only one agent at a time to move vertically past the long center obstacle (on the extreme left or right).

Two agents and some number of enemies will populate this space. One of the agents and all the enemies are AI-controlled (see below).



2. Simple enemies are spawned at doorways and move horizontally, following the dotted lines going from one doorway on the left (or right) to the corresponding doorway on the right (or left). When they encounter the obstacle in the middle, with equal probability they either move through the obstacle unhindered, they disappear, or they reverse direction. Enemies do not collide with other enemies, and de-spawn when they reach a doorway (and then respawn at a doorway again). **15**

Enemies have a limited field of view. The yellow box in the above diagram shows the general proportions for an enemy moving leftward — they can see ahead a short distance, a much smaller distance behind, and (importantly) to the side sufficiently to fill the corridor area. If they detect an agent in their field of view then the agent is captured and removed from the game. (Enemies do not have field of view while inside the middle obstacle.)

3. The two agents compete. Each alcove has an item, and each agent aims to acquire as many of the items as they can, while avoiding being seen by any enemy. Items are acquired by simply passing over them, and once acquired disappear. Once all items are acquired (or both agents captured) the simulation ends, and either a tie or a winning agent is declared. **15**

Agents are initially spawned in random (different) alcoves. They cannot pass through each other or enemies (although the latter is somewhat moot, as that also means they are seen by the enemy).

Agents each have 2 “teleport traps”. When used, the closest enemy or agent is selected, and immediately moved to a random alcove (agent), or de-and-re-spawned (enemy).

One of the agents should allow for player control (the space bar selects use of a teleport trap). Provide a fixed camera view, giving a clear overhead perspective on the level.

4. Control of other other agent is through a *hierarchical task network*. Include, as a *separate document* a visual illustration of your HTN. Implement a simple forward sequential planner. **15**

Verify that with the player agent idle and not interfering or competing, and with just two enemies (one on the top route and one on the bottom) that the AI agent can usually win. Note that you will need to experimentally determine relative speed of the enemy agents to ensure this is possible.

5. Now, with the player agent active and competing, try to ensure your AI agent can still win. In particular, your AI agent should not get stuck and should recover from being blocked in movement (by the other agent), and from being teleported. It should be able to use its teleport on the player agent (sometimes). **10**

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.