**Assignment 4**

Due date: Thursday, December 7, 2017, 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**

This assignment focuses on game AI and procedural content generation. You can do this assignment in Unity, or in another appropriate environment. You can use built-in collision management, physics/geometry APIs, and/or pathfinding, but you must implement all aspects of the actual decision-making AI system yourself: do not use any built-in or external asset tools for behaviour tree construction, management or execution in Unity (or other environments).

1. This design consists of agents moving in a space consisting of two rows of small rooms, facing each other, and separated by several parallel laneways, as per the following (necessarily to scale) overhead view:

   ![Image](image.png)

   The (red) dots are meant to approximate relative character size. The amount of space in each room is not especially important as long as it fully accommodates a single character. Note that the scene extends further to the right past the end of the rooms than it does on the left.

   There should be at least 4 laneways, each of which is 2–3 characters wide, and which collectively fill the space between the rows of rooms. Note that the lane separators are meant only for organizing and controlling character movement choices, and while they should be visible in some way they are not physical barriers.

2. Build an AI agent based on *behaviour trees*. This agent will be spawned in one of the 8 rooms, randomly selected, and must be able navigate to the back of each other room (ie each red dot) before going to the extreme right (green area).

   Scale the movement (and/or geometry) so it takes the agent around 2-4s to move from one room to its opposite across the laneways, and 1-3s to move between adjacent room entrances in the same row of rooms.

   *All* actions of your agent should be controlled by evaluating your behaviour tree. You can build your own behaviour tree interpreter (and associated data structures), or hard-code it.

   In a *separate document* draw your behaviour tree. Show all internal nodes, and briefly explain the function of any specialized decorators, and all leaf nodes.

3. The laneways will contain moving obstacles, pedestrians, that your AI agent will need to avoid. Pedestrians spawn on the left (orange area) of the laneways, within a randomly chosen, specific lane, and move to the right, de-spawning at the end (green area). Pedestrians have a baseline movement speed faster than your AI agent, and if moving in a straight line should be able to complete a route across the scene about 1.5–2x as fast as your agent would.
Pedestrians, however, do not always just move in a straight line. Instead, they each follow their own pattern of movement, giving them a relatively unique cycle of movement actions they apply to get across the scene. Rather than defining patterns manually, patterns will be procedurally generated. Define a grammar that can be used to create different pedestrian movement patterns. Patterns should be based on various choices for interesting human-like pedestrian behaviour, including minor deviations in direction within a lane, changing lanes altogether, pausing, reversing direction (for a time), etc.

Your grammar design should place reasonable upper limits on pedestrian speed, guarantee that pedestrians do eventually get to the opposite side, and properly define a seamless cycle of movement behaviour.

Pedestrians do not collide with each other, but must stay within the laneway area, never entering rooms. Your physics/movement system should guarantee that, not your grammar, but keep the fact that movement may be constrained in mind when designing your grammar.

Implement your design, and include a parameter to specify the number of pedestrians that will be present in the scene at the same time (spawning new ones when one reaches the end). In a separate document draw your grammar, briefly describing/explaining the actions you allow and how you provide the guarantees.

4. Your AI agent must not get too close to pedestrians—being within a lane-width radius of a pedestrian, outside of a room, indicates failure.

Modify your behaviour tree to help your agent complete their room visits and reach the ending successfully, avoiding pedestrians. With only a few pedestrians your agent should be able to almost always succeed.

In a separate document describe and draw your modified behaviour tree, pointing out differences from the tree given in question 2.

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.

This assignment is worth 15% of your final grade.