Name: Sanad Masannat CCID: sanad Date: 10/22/2021

Instructions: Each question is worth the given amount of points, and the whole quiz is worth nine points in total. Answer each question to the best of your ability. Read all instructions carefully. Submit your quiz to eClass before Sunday at 11am. You may submit your quiz as a pdf, a docx, or as a zip file of images. **It is your responsibility to ensure the TAs and instructor can read your answers**, if you’re concerned about that please type your answers when possible.

**Multiple Choice (0.5 points each)**. Circle the answer closest to the one you would give.

**Q1 (0.5 points).** Given a game that makes use of Elo ‘s performance ratings, which of these situations incorrectly depicts how a player’s rating would change?

(A.) Victory against an opponent with a higher rating-> rating goes up a lot

(B.) Loss against an opponent with a higher rating -> rating goes down somewhat

**(C.)** Victory against an opponent with the same rating -> rating doesn’t change

(D.) Loss against an opponent with a lower rating -> rating goes down a lot

**Q2 (0.5 points).** Below is a set of comparisons between forward planning (FP) and backward planning (BP). Indicate the most **accurate** comparison.

(A). BP outperforms FP when there are many actions that accomplish the same effect

(B.) BP simulates cause to effect, while FP simulates effect to cause  
**(C.)** FP takes longer than BP when there are many irrelevant actions

(D.) FP performs worse than BP when there are many inconsistent actions

**Q3 (0.5 points).** Assuming general algorithm implementations are available, which of the following would require the **most** designer/developer authoring for the same AI behaviour?

(A). Behavior Trees (B.) Forward Planning

(C.) Partial Order Planning **(D.)** Hierarchical Task Networks

**Q4 (0.5 points)**. Given the Hierarchical Task Networks (HTNs) implementation from F.E.A.R. (2005) which of the following would **most** speed up the **average** computation cost?

**(A).** Adding another level of tasks above Squad Behaviors. (B.) Adding another action.

(C.) Adding another goal to each agent. (D.) Adding another squad behavior.

**Q5 (2.5 points).** Answer the below parts to the best of your ability.

**Q5.A. (0.5 points**).

Create a planning environment. The actions available to the agent are as follows:

* **Move**(x,y): If x,y is a neighbor (up, down, left, right) of the *agent* and the neighbor cell is unoccupied by an *enemy* or *obstacle* then move the *agent* to that location.
* **Pickup**(item): If *item’s* x,y is equivalent to *agent’s* x,y then put *item* in *agent’s* inventory.
* **Unlock**: If *agent’s* x,y is a neighbor (up, down, left, right) to the *gate* and the agent has two keys in its inventory then the *gate* is unlocked.
* **Kill**(enemy): If *agent’s* x,y is a neighbor (up, down, left, right) to *enemy* then remove *enemy*from the environment.
* **Heal**: If *agent’s* hitpoints are under 3 set them to 3.

The environment has the following entities

* Agent (A): The “player” or main entity in this environment. It starts with 3 hitpoints and an empty inventory.
* Enemies (E): Entities that are hostile to the player. If the agent does anything but use the Kill action in a neighbouring location (up, down, left, right) to an enemy then the agent loses 1 hitpoint (assume hitpoints are lost upon leaving the neighbouring cell or after doing some other action besides Move in the neighbouring cell, and that the agent’s actions occur first).
* Keys (K): A simple item.
* Gate (G): An entity that can be unlocked with the Unlock action. Can be locked or unlocked. Starts locked.
* Obstacles (O): An entity that cannot be passed through.

Fill in the below environment with 1 agent, 1 goal, 2 keys, (at least) 2 enemies, and 3 obstacles. Each key must neighbor (up, down, left, right) at least one enemy. The agent must start in the leftmost column and the goal must be placed in the rightmost column.

|  |  |  |  |
| --- | --- | --- | --- |
| A | E |  |  |
|  | K |  |  |
|  |  | K | E |
|  |  |  | G |

As an example (do not have an answer within 3 edits/changes of this):

|  |  |  |  |
| --- | --- | --- | --- |
| E | K |  |  |
|  |  |  | G |
| A |  |  |  |
|  |  | K | E |

**Q5.B. (1 point)** Give an optimal plan (fewest number of steps) that you could achieve with **forward planning** based on your environment answer to **Q5A** given a goal where the gate is unlocked and the agent has >0 hitpoints. **Note:** If you do not achieve these goals your entire path is invalid (i.e. don’t let your agent die). Assume your forward planner is making use of A\*. If you believe that the heuristic would impact your optimal plan, state it. Otherwise assume the heuristic is the number of differences between the current world state and a goal world state (with the Agent at the unlocked goal and both keys picked up). Assume the top left corner of the grid is position 0,0.

**Kill(enemy at 1,1)**

**move(0,1)**

**move(1,1)**

**pickup(key)**

**move(2,1)**

**move(2,2)**

**pickup(key)**

**kill(enemy at 3,2)**

**move(2,3)**

**unlock(gate).**

**Q5.C. (1 point)** Halfway through running through your plan from **Q5B** (round down if you have an odd number of actions) the goal (G) suddenly moves from the original position on your answer to Q5A to the agent’s starting position from Q5A. What would the new plan be after replanning from this point?

**Kill(enemy at 1,1)**

**move(0,1)**

**move(1,1)**

**pickup(key)**

**move(2,1)**

**move(2,2)**

**pickup(key)**

**kill(enemy at 3,2)**

**move(3,2)**

**move(3,1)**

**move(3,0)**

**move(2,0)**

**move(1,0)**

**unlock(gate).**

**Q6 (1 point)** Now imagine just like in the Bethesda’s Oblivion example from the lecture you have complaints about the agent from **Q5** being too violent. Thankfully you know about Partial Order Planning. (A) Write out the heuristic (describe it or use pseudocode) that you would write so that a POP approach would find the optimal path with minimal violence (minimal usages of the Kill action) (**0.5 points**). (B) How would your optimal plan from **Q5B** change if at all? You only need to describe the change (or lack of change), though you may write out the new optimal plan that you would get using POP and your new heuristic (**0.5 points**).

**The new heuristic I would use is that if there is a enemy unit that is at least 2 spaces away from you and there is other paths to our current goal, take a different path away from it. However, this heuristic will not change the plan at all due to the environment and the enemy locations.**

**Q7 (1 point)**. Given the environment from **Q5A**,what would have to change, if anything, to make it appropriate to use a Hierarchical Task Network (**0.5 points**)? What additional information would you need to author to make HTNs work (**0.5 points**)?

**To make HTN’s work given my environment, we would first need to know everything about the world state (items available, how many enemies where our goal is) and we will need to have weighted goals so we can be able to use A\* to get from our current position to our intended goal. You would also need to allow the units to move from one space to the other or allow them to hide behind the objects so they can avoid being killed. We would also need to implement some behaviours into the enemy units.**

**Q8 (2.5 point).** Answer the below parts to the best of your ability.

**Q8.A. (0.5 points)** Design a Behavior Tree with the actions: (A,B,C,D,E,F) and the variables: (var1, var2, var3). You may **not** use Boolean operators (and, or, or not). You must use all actions once with no repeats. You must use at least 1 of the 3 variables. You must use at least one priority list and at least one sequence. Each priority list and sequence must have at least two actions as children.

For example, a valid B tree would be (do not include an answer within 4 edits of this):

**Diagram

Description automatically generated**

You may either draw out your behavior tree or specify the edges. For example, the above could be represented as:

root->priority list(1)

priority list(1)->A,B

A->var1

var1->sequence

sequence->C,D

B->var2

var2->priority list(2)

priority list(2)->E,F

Where (1) and (2) are arbitrary indicators for the two different priority lists.

**root->priority list(1)**

**priority list(1)->E,F**

**F->var3**

**Var2-> priority list(2)**

**priority list(2)->A,B**

**E->var2**

**var2-> sequence**

**sequence ->C,D**

**D-> var1**

**Q8.B. (2 points)** For the given sequence of states below give what one action node would fire. Hint: In other words you should only ever answer A,B,C,D,E or F (a single letter). “A: None” means that A would return none for this state if it fired. Assume these states occur in sequence from 1-8. Assume any variable (var1, var2, and var3) transitions occur if they are true and if the B tree would otherwise attempt to fire the action node that it is connected to.

|  |  |  |
| --- | --- | --- |
| id | State | Action Node Fired |
| 1 | A: None, B: None, C: None, D: None, E: True, F: True var1: True, var2: True, var3: False | E |
| 2 | A: True, B: True, C: None, D: None, E: True, F: True var1: True, var2: True, var3: True | E |
| 3 | A: True, B: True, C: True, D: True, E: True, F: True var1: True, var2: True, var3: True | D |
| 4 | A: False, B: True, C: False, D: None, E: False, F: True var1: True, var2: False, var3: True | F |
| 5 | A: True, B: True, C: None, D: None, E: None, F: None var1: False, var2: True, var3: True | A |
| 6 | A: False, B: False, C: False, D: False, E: False, F: False var1: False, var2: False, var3: False | A |
| 7 | A: True, B: True, C: True, D: True, E: True, F: True var1: False, var2: False, var3: False | B |
| 8 | A: None, B: None, C: None, D: None, E: None, F: None var1: True, var2: True, var3: True | B |

**EXTRA CREDIT (0.5 points)** In Alien: Isolation from 2014 the player is stalked by the titular Alien (a xenomorph if we want to get specific). Reviewers praised the game for its intelligent Alien AI that “learned” and “adapted” to the player. However, the game’s creators only used one pre-authored Behavior Tree for the Alien AI. They accomplished the desired behaviour that seemed to “learn” by having more of the Behavior Tree “unlock” as the game went on.

Given your answer from **Q8A** redesign it as an “Alien: Isolation-style” Behavior Tree where at level 0 the AI agent only has access to actions A and B, at level 1 it has access to A, B, C, and D, and at level 2 it has access to all actions. Your tree should remain the same such that at level 3 it would give the same answers that you gave for **Q8B**. State any implementation details you are assuming for this to work.