Name: Sanad Masannat CCID: sanad Date: 8/10/2021

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**) Below is a set of statements on rule systems. Circle the **incorrect** statement.

a. They require balancing emergence/over-engineering b. They are easier to debug than FSMs

c. They don’t require writing a response to every world state d. They are more general than FSMs

**Q2**) What is the clearest and most accurate comparative statement for FSMs and Btrees?

a. Btrees are faster than FSMs b. Btrees are more in**t**uitive than FSMs

c. Btrees can represent behaviour that FSMs cannot d. None of the above

**Q3**) Which of the below is the **clearest** indication that we cannot apply APSP to a game environment?

a. There are many agents moving simultaneously b. The environment’s graph has edges with negative weights

c. The environment is generated at runtime d. The player can alter the environment and its graph

**Q4**) Imagine you have a game with a massive map where you regularly need to send AI agents between nodes on the corners of the map (merchant caravans travelling between the four big cities on the corners). The map changes at three fixed points: (1) a canal that can be open or closed, (2) a drawbridge that can be raised or lowered, and (3) a mountain road that can be open or caved in. All three of these can change independently and randomly at fixed intervals, but the map does not change otherwise. Assume memory is of no concern, but runtime computation is limited, which of these options would lead to fastest pathing system for the merchant caravans?

a. A\*, recalculated anytime the map changed

b. APSP, with a unique nav table for each map configuration

c. MCTS with a small number of rollouts

d. A\* where we just wait after a path becomes invalid for it to become valid again

**Q5 (1 point).** Given the below final nav table give a graph that would lead to this nav table when using APSP. This is essentially asking you to do the opposite task from the practice quiz last Friday. Your graph will need to have 5 nodes (A, B, C, D, and E) and around 8 edges. You can draw your graph or write out each edge (e.g. A->B with a cost of 5, B->C with a cost of 1, etc…). Every edge will have a cost. There are multiple correct answers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E |
| A | A | B | B | B | B |
| B | D | B | C | D | D |
| C | D | D | C | D | D |
| D | A | E | E | D | E |
| E | A | B | B | B | E |

6

A

3

B

C

2

#

1

8

7

4

**=Cost**

E

D

5

**Q6 (2.5 points).** Answer the below parts to the best of your ability.

**Q6.A. (0.5 points**).

Variables and possible values:

1. who: ?anyone, sally and trish
2. action: ?any, hit, miss, or heal
3. source: ?anyone, sally, trish, and alien
4. health: ranges from 0-1
5. currMap: ?any, sewer, UFO, university

Create 5 rules using the above variables and possible values and conditions. These rules will be pieces of dialogue (called “barks” in video games) that the character Sally says. The game is an alien invasion adventure game, but you’re welcome to invent any other details you like. For example:

|  |  |  |
| --- | --- | --- |
| **Rule ID** | **Condition** | **Effect (What Sally Says)** |
| 1 | {who = ?anyone, action=hit} | “Ouch!” |
| 2 | {who = sally, action=hit, currMap = sewer} | “I hate sewers!” |
| 3 | {who = sally, action=miss, health <0.25} | “That was close!” |
| 4 | {who = sally, action=hit, source=alien} | “Bad E.T.!” |
| 5 | {who = trish, action=hit, health <0.25, source=alien} | “I always knew an alien would kill Trish.” |

Your answer must include at least

* 1 rule with 2 conditions
* 3 rules with at least 3 conditions
* 1 rule with at least 4 conditions (hint: no rule can have more than 5 conditions)

There should be at least 5 condition changes between your answer and my example. You should not have the same answer as any other student. Only make use of the possible condition values given.

|  |  |  |
| --- | --- | --- |
| **Rule ID** | **Condition** | **Effect (What Sally Says)** |
| 1 | {who=sally, health<0.5} | “I need healing!” |
| 2 | {who=?anyone, action=miss, source=alien} | “Can’t hit us!” |
| 3 | {who=trish, action=hit, source=sally} | “I can’t control my actions! Sorry!” |
| 4 | {who=sally, health<0.1, currMap=university} | “I knew university would be the death of me…” |
| 5 | {who=sally, health>0.75, action=hit, currMap =Sewer} | “Ugh, the smell is making my reflexes horrible!” |

**Q6.B. (1 point**). Given your list of rules above, give what effect (if any) would occur in each of the states in sequence (a-e, **0.2 point*s* each**), using a heuristic that prefers activated rules with the most conditions followed by how infrequently the rule has fired. Specifically, # of conditions minus times the rule has fired. Choose randomly if there are equivalent rules. You may give the # of the Rule ID, write outthe effect, or write N/A if no rule would fire.

**States**

1. {who = sally, action=hit, source=alien, health =0.2, currMap=university } -> 1
2. {who = trish, action=heal, source=sally, health =0.2, currMap=university } -> N/A
3. {who = trish, action=hit, source=alien, health =0.3, currMap=sewer } -> N/A
4. {who = sally, action=hit, source=alien, health =0.9, currMap=UFO } -> N/A
5. {who = sally, action=heal, source=alien, health =0.5, currMap=UFO } -> N/A

**Q6.C. (1 point**). Given your list of rules from Q6.A. write 5 states that would cause each of your rules to fire in **descending order**, so rule 5 should fire for state 1, rule 4 should fire for state 2, and so on. Assume you are using the same heuristic as in Q6.C. For some of you, you will need more than 5 states to ensure you have at least one state where all rules fire. You only need to write 5 states explicitly (as in Q6.B), and just describe the other state(s) in between them at a high level (e.g. what rule fired in them).

1. {who = sally, action=hit, source=alien, health =0.8, currMap=Univeristy } 🡪 5
2. {who = sally, action=heal, source=sally, health =0.03, currMap=university }🡪4
3. {who = trish, action=hit, source=sally, health =0.9, currMap=UFO }🡪3
4. {who = trish, action=miss, source=alien, health =1, currMap=Sewer }🡪2
5. {who = sally, action=hit, source=alien, health =0.4, currMap=UFO }🡪1

**Q7 (1 point)**. Given the fictional game from Q6, invent details aboutit such that it would be appropriate to apply MCTS to the AI of the game (for path planning and/or decision making). **State explicitly why the detail(s) you add would indicate we should use MCTS over other path planning and/or decision-making approaches.**

Parts of the large maps can be blocked off by the user and/or enemies at any possible time through various means like barricades or collapsed debris. This means that we can not use APSP as its size is large and our environment is constantly changing. A\* is very computational heavy as well so it will be very costly due to map size in the game so MCTS is a better option in comparison.

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

**Q8.A. (0.5 points)** Design a Finite State Machine (FSM). With the possible states (A, B, C, and D) and the possible variables (var1, var2, var3, and var4). Your answer must make use of every state and every variable at least once. Assume variables are Boolean, so negating (!) them or connecting them with some Boolean relationship like and (&&) and or (||) is valid.

You can draw out your FSM:

Diagram

Description automatically generated

Or represent it in a table

|  |  |  |
| --- | --- | --- |
| State | Transition | State’ |
| A | !var1 | B |
| A | !var3 | C |
| A | var4 | D |
| B | var1 | A |
| B | !var2 | D |
| C | var3 | A |
| D | var2 | C |

Your FSM must have **at least 7 transitions** and make use of every state and every variable at least once. In addition, it must not include a dead-end or terminal state. Your answer may not be within 3 changes (adding/editing/removing transitions) of my example. You should not have the same answer as any other student. **IDENTIFY YOUR INITIAL NODE/STATE.**

**C=Initial Node**

|  |  |  |
| --- | --- | --- |
| State | Transition | State’ |
| A | var1||var3 | B |
| D | var2 | C |
| C | var4&&!var3 | D |
| B | var1 | C |
| D | !var2 | A |
| C | var3 | A |
| C | var2||!var4 | B |
| A | var3 | D |

**Q8.B. (2 points)** Given the series of states below, give what state/node would fire in each time step (**0.25 points each**) given the world conditions for your answer to **Q8.A**. Assume that transitions occur *before* a state/node fires. You may not have answered **Q8.A** in such a way that **it would** lead to you giving the same state for every time step here. If that is the case, redo your answer to Q8.A.

|  |  |  |
| --- | --- | --- |
| Time Step | State | FSM State |
| 1 | var1=F; var2=F; var3=T; var4=F | A |
| 2 | var1=T; var2=F; var3=F; var4=T | B |
| 3 | var1=F; var2=T; var3=F; var4=T | B |
| 4 | var1=T; var2=T; var3=T; var4=T | C |
| 5 | var1=T; var2=F; var3=T; var4=F | A |
| 6 | var1=F; var2=F; var3=F; var4=F | A |
| 7 | var1=F; var2=F; var3=T; var4=F | D |
| 8 | var1=T; var2=F; var3=F; var4=F | A |

**EXTRA CREDIT (0.5 points)** With a rule or production system a single rule fires every frame. Similarly, a single state executes/runs every frame in a finite state machine. Using the rules you designed in Q6, create an equivalent finite state machine where each state is one effect/dialogue bark. Make up any additional variables you need but explain what they are. Your FSM should give the same **output** (the same line of dialogue spoken) for the states in Q6.B and Q6.C. You will get **0.1 points** for any attempt at this question.