Game 12

Fuzzy Logic

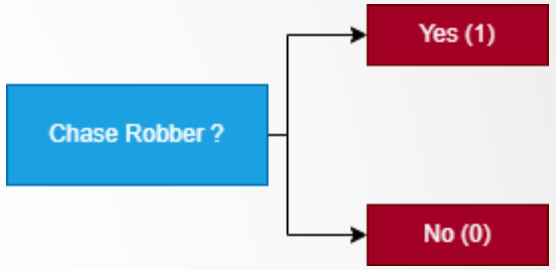
Fuzzy Logic ~ Introduction (1)

- What is Fuzzy Logic ?
- Computers work by utilising Boolean logic, i.e., values can have two states, 0 or 1/ “True” or “False” [1].
- In some cases, **Boolean Logic** can be quite beneficial to work with, however, this type of logic can be a little constraining at times, and sometimes one would need to classify something based on varying **Degrees of Truth**.
- In Traditional Boolean Logic, changing from states/responses will seem abrupt, however utilising **Fuzzy Logic** will provide smooth transitions, since there are various degrees of truth for each state [2].

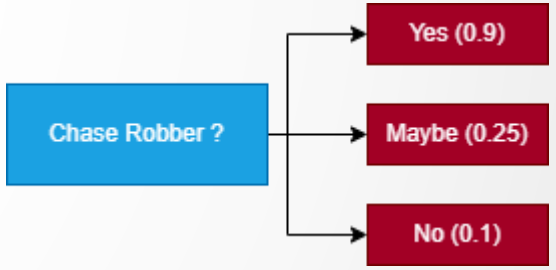
Fuzzy Logic ~ Introduction (2)

- Fuzzy Logic can be applied to different games, as to imitate human thinking and reasoning [1].
- From the following diagrams, portraying a Police NPC (Non-playable character) decision surface, one can observe that:
  - In the **Boolean Logic Model**, there are only two degrees of truth, thus transitioning between states would seem unrealistic when compared to real-life scenarios.
  - In the **Fuzzy Logic Model**, there is more than one degree of truth, however for brevity the diagram is only showing 3. Transitions between states would seem more realistic, as to mimic real-life scenarios, where maybe the Police is feeling tired, or the Police is feeling hungry.

Boolean Logic Model



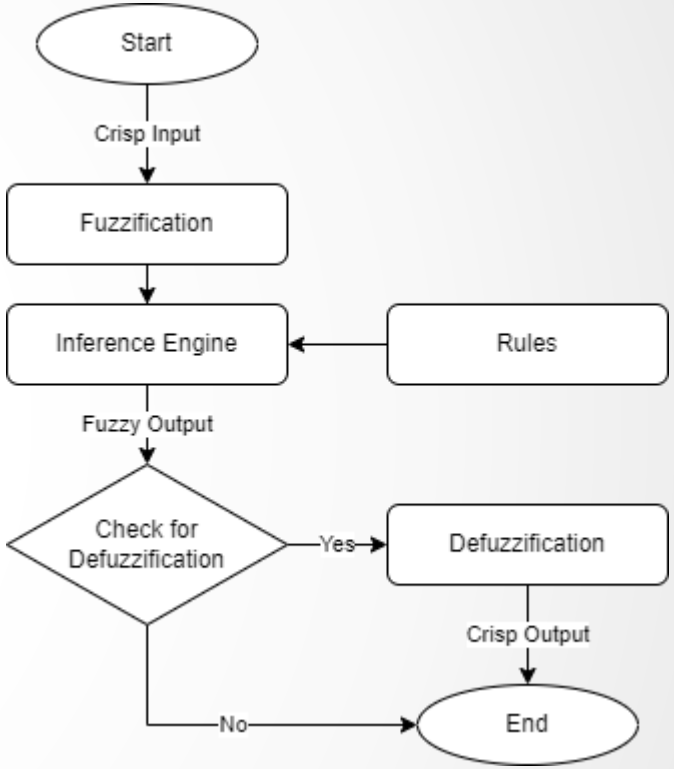
Fuzzy Logic Model



Fuzzy Logic ~ AI Explanation (1)

- Implementation of the Fuzzy Logic System can be partitioned in the following steps:

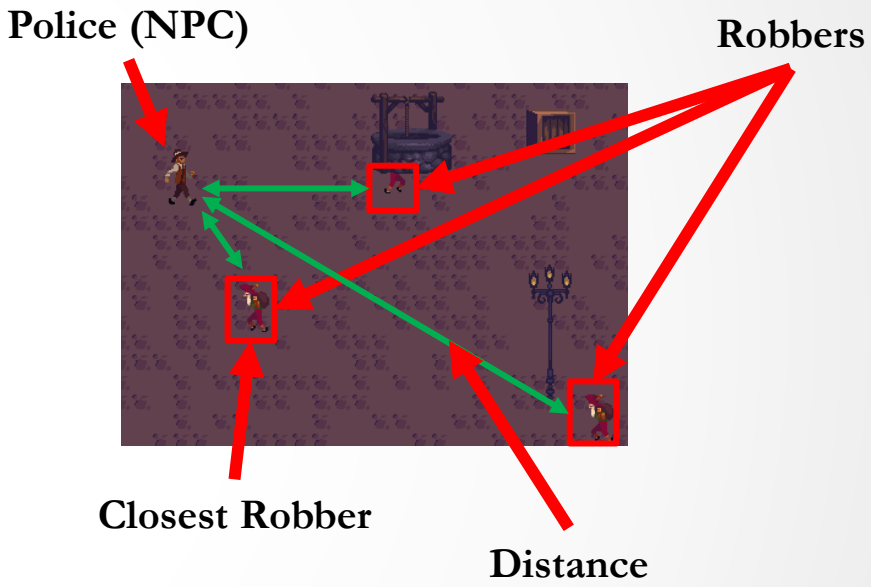
1. **Fuzzification**
2. **Inference Engine utilising Rule Base**
3. **Defuzzification**



Fuzzy Logic ~ AI Explanation (2)

1. **Fuzzification:**

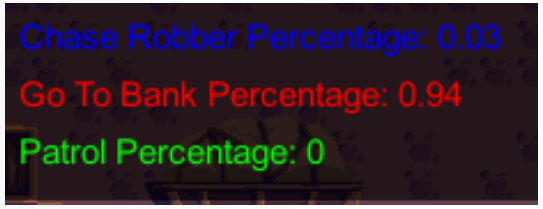
- What is Fuzzification?
- Fuzzification is the Process of converting crisp input data into Fuzzy data [2].
- The Process of Fuzzification maps the input value to the degree of membership in a Fuzzy Set.
- Membership Degree [2]:
  - A Degree of 0 marks undeniably "False".
  - A Degree of 1 marks undeniably "True".
  - Any Degree in between can mark True or False, depending on the extent.
- Continuing with the example from above regarding the Police NPC, as can be seen in the following image, let the input value for Fuzzification, be the distance between the Police and the Closest Robber.



Fuzzy Logic ~ AI Explanation (3)

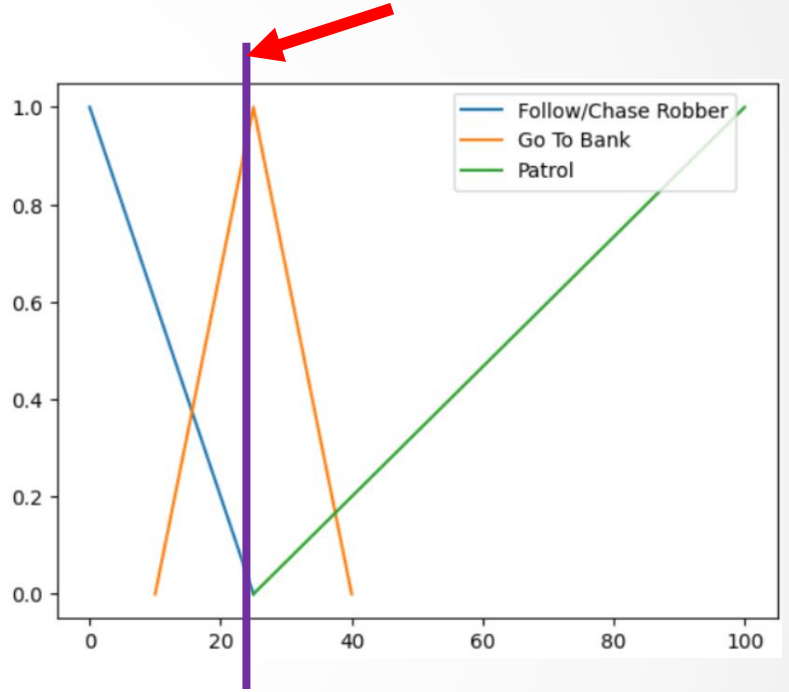
1. **Fuzzification (Continue):**

- The following Diagram shows the Mapping of the different **Fuzzy Membership functions** for the Police NPC.
- Given a Closest Robber Distance input of 23, the system can determine the Degree of Membership for each Fuzzy Set.



Degree of Membership For all the Fuzzy Sets

Given a Closest Robber Distance of 23 as input (x-axis value)



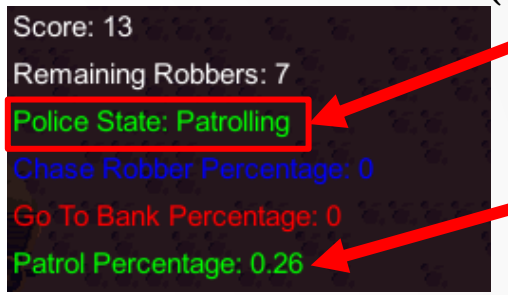
Visualisation of Membership Functions, and how the system can determine the Degree of Membership for each Curve

Fuzzy Logic ~ AI Explanation (4)

2. Inference Engine with Rule Base:

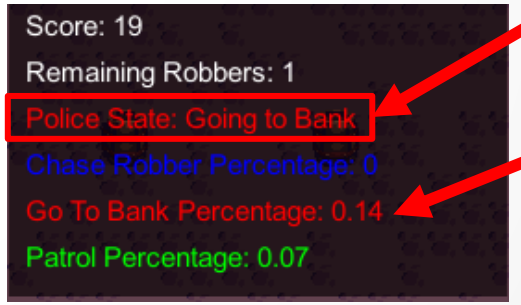
- What are Fuzzy Rules ?
- After Fuzzification, the system must utilise the Degree of Membership for each Fuzzy Set to produce a Fuzzy Output. Determining the type of Output, is done through the utilisation of **Fuzzy Rules or Axioms**.
- Fuzzy Rules or Axioms, are in essence a bunch of if-statements, like Expert Systems, whereby, through these rules, the system would infer the type of Fuzzy Output.
- Continuing with the example from above, knowing the Degree of Membership for all the Fuzzy Sets for the Police NPC, the system would apply a simple Fuzzy Axiom which takes the highest Degree of Membership between all the Fuzzy Sets.
- After applying the Fuzzy Axiom, the system proceeds to determine the respective Police State.

Determining Fuzzy Output Set (Patrolling)



Highest Degree Of Membership

Determining Fuzzy Output Set (Going to Bank)

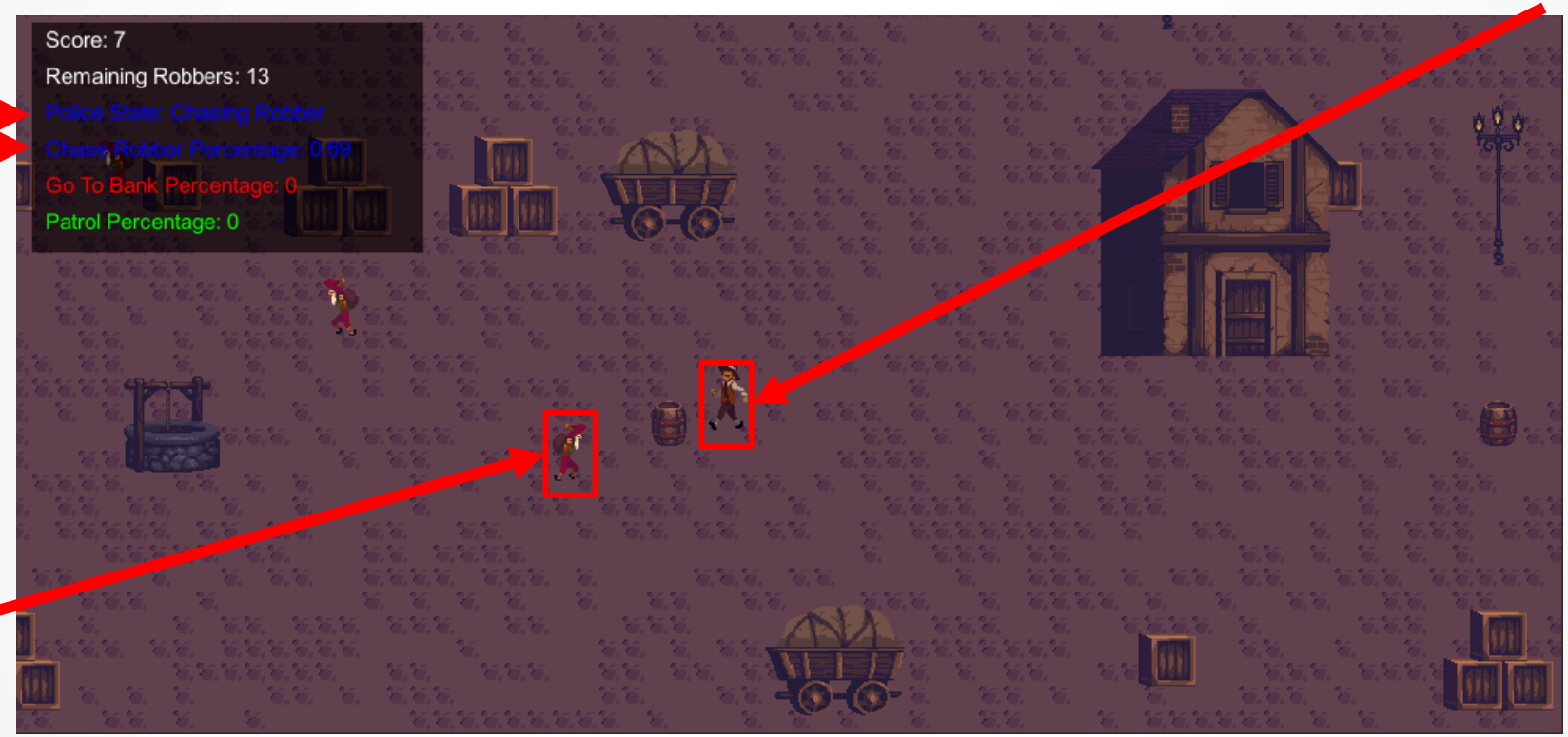


Highest Degree Of Membership

Fuzzy Logic ~ AI Explanation (5)

Highest Degree of Membership (Following Robber):

Police Chasing after Closest Robber



Closest Robber

Fuzzy Logic ~ AI Explanation (6)

Highest Degree of Membership (Going to Bank):

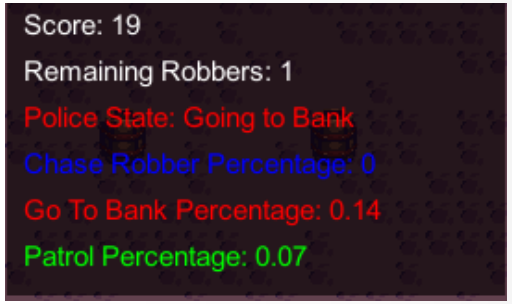
Fuzzy Logic ~ AI Explanation (7)

Highest Degree of Membership (Patrolling):

Fuzzy Logic ~ AI Explanation (8)

3. Defuzzification:

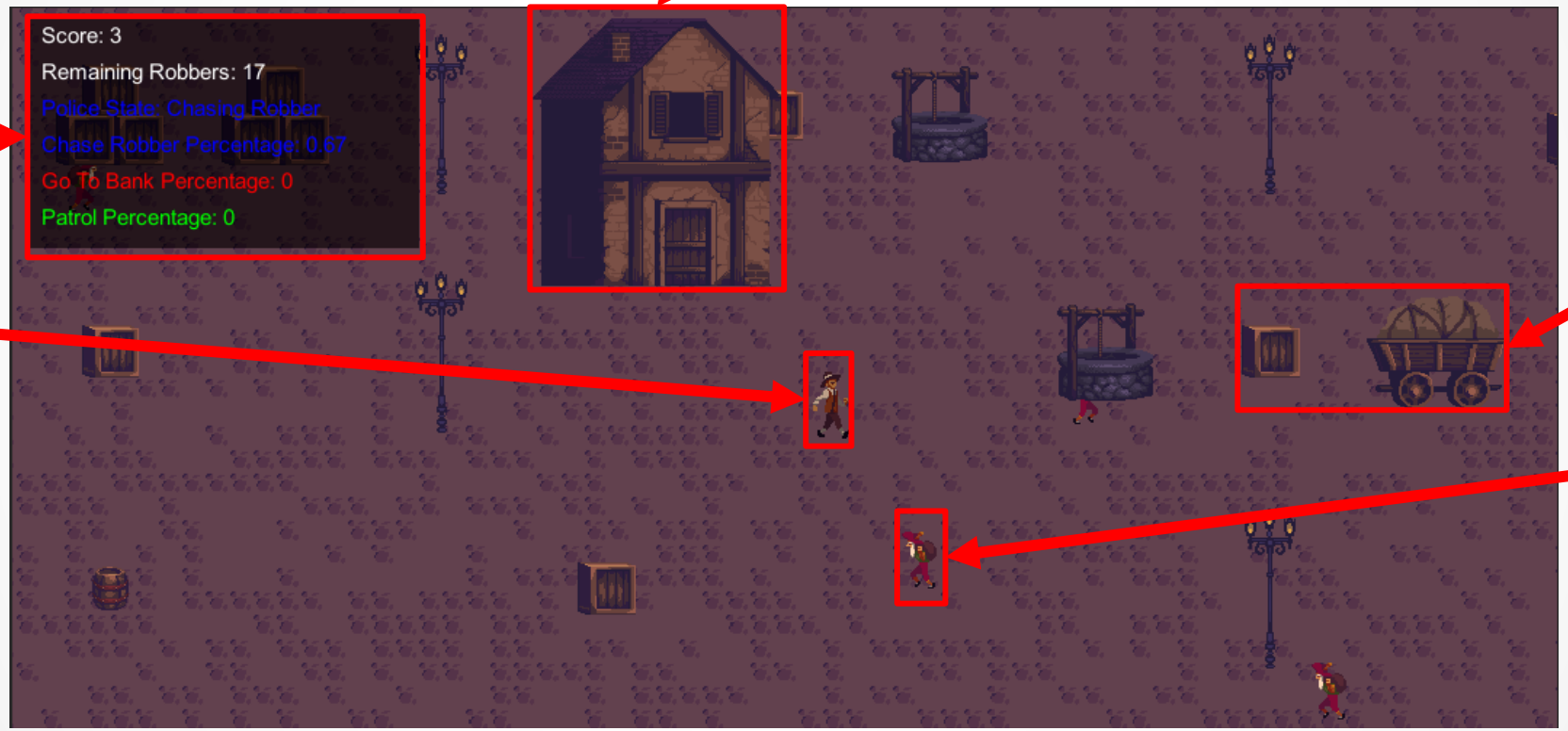
- What is Defuzzification?
- In some cases, the system would need to use the fuzzy output degree to determine a crisp output value which would be a real number [2].
- Furthermore, this crisp output value would be used for additional calculations. For example, to calculate an NPC's health or energy level.
- In the previously mentioned example with regards to the Police NPC, Defuzzification was not implemented, as it is not always needed.



Degree Of Membership And Fuzzy Output

Fuzzy Logic ~ Mini-Game Implementation (1)

Playable Area:

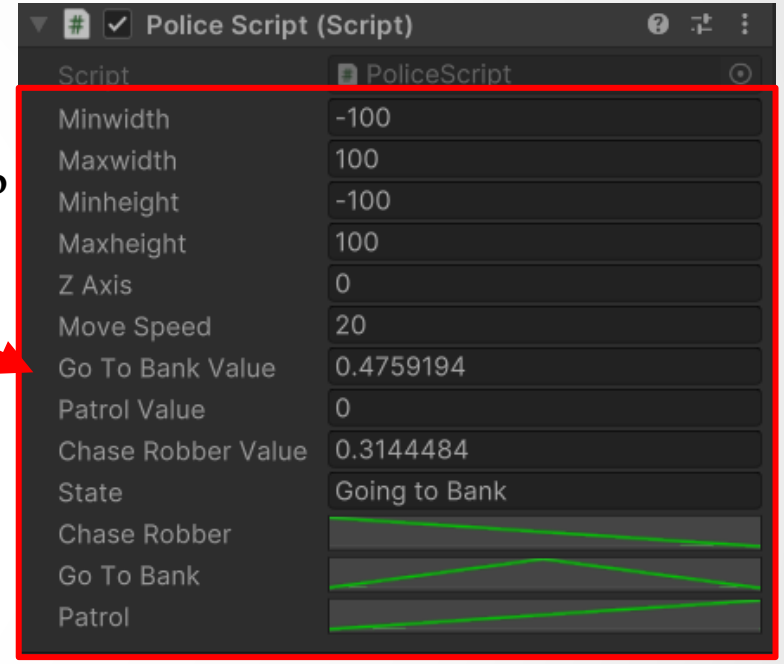


Fuzzy Logic ~ Mini-Game Implementation (2)

Developer Interface:

- As can be seen in the following image, developers, are given a large degree of freedom, in which they can configure various parts of the game, including prefabs located in Assets>Prefabs folder.
- Developers, can choose the area over which certain NPC's can roam, as well as change the different Membership Functions of the Police Prefab.

Customization of Prefab parameters



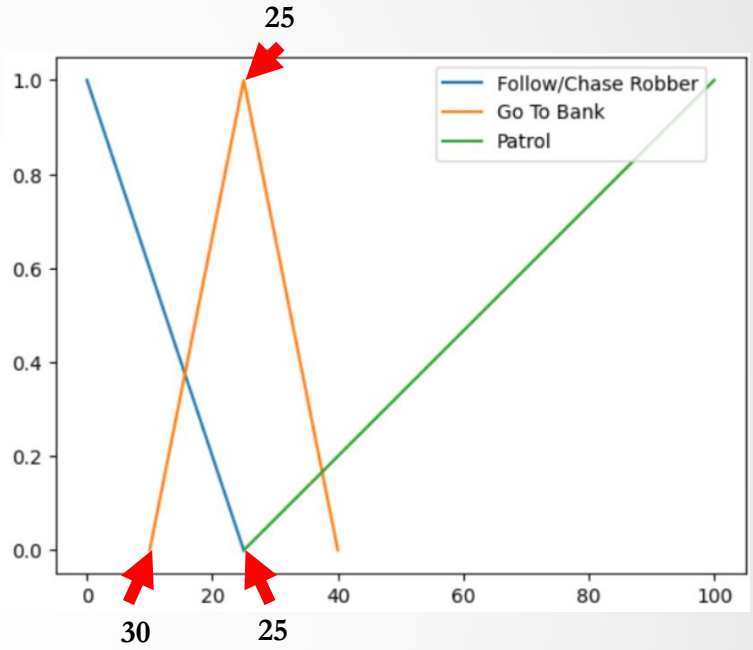
Fuzzy Logic ~ Mini-Game Implementation (3)

- Implementation of the Mini Game was inspired from [3-4], and sprites used to create the game were retrieved from [5].
- **The Game is Composed of the following scripts:**
  - **Background** script – This script is being used to generate the Game Background/Ground.
  - **Robber** script – This script is being used to control the Robber Game Objects, to move from the different points on the Map.
  - **Police** script – This script is being used to control the Police Game Object, to move around the map, and try to catch all the Robbers. Furthermore, this script contains the entirety of the Fuzzy Logic AI component in this game. **Note that the Police Game Object has a slower speed than the Robber to show the benefits of Fuzzy Logic.**
  - **CameraFollow** script – This script is being used to ensure that the Camera is always following the Police Game Object.
  - **GameManager** script – This script is being used to spawn the respective NPC Game Object Prefabs on the map, as well as handling the user interface via OnGUI() method. This script is also responsible for spawning the relevant structures on the map, at random positions, thus acting as a Procedural Generation Map.

Fuzzy Logic ~ Exercise (1)

Now Its your turn to Code! – Let's implement the Fuzzy Logic Algorithm ☺

1. Navigate to the Assets> Prefabs>People folder, and click on the Police Prefab
2. Modify the 3 Animation Curves on the Police Prefab labelled chaseRobber, goToBank and patrol through **Unity's curve editor window**, so that the fuzzy curves would mimic the following diagram.



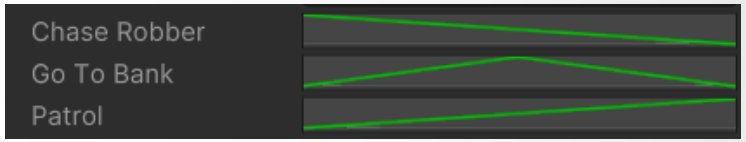
Visualisation of Membership Functions

Fuzzy Logic ~ Exercise (2)

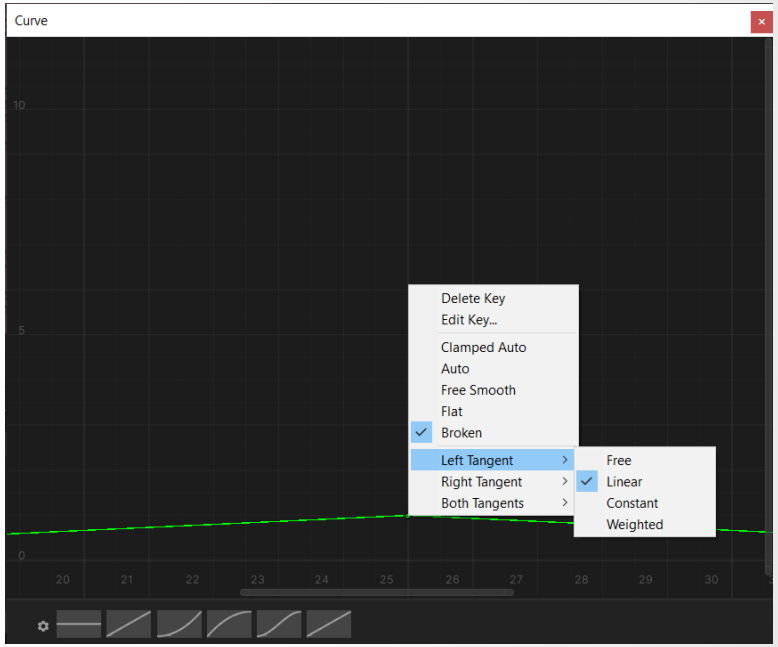
Utilising **Unity's curve editor window:**

1) First, Click on one of the Curves.
2) Then arrange the Curve, to create a new point on the Cure, right Click on any point on the Curve and select "**Add a Key**".
3) To arrange a point value, right Click on that point, and arrange the specific specification, one wishes to change.
4) **NB: The Go To Bank Curve requires that its peak point, would have Linear Left and Right Tangents.**

1)



3)



Fuzzy Logic ~ Exercise (3)

1. Navigate to the Assets> Scripts folder, and open the **Police** script
2. In the Police script find the **Determine State()** method
3. Utilise the following Pseudocode to populate this method (Determine State() method)

Pseudocode:

1. Retrieve the Closest Robber distance.
   - (Hint: use the GetClosestRobber() function)
2. Utilise the Evaluate() method for each curve to retrieve the degree for each fuzzy set, given the closest robber distance as an input.
   - (Hint: store the results in the chaseRobberValue, goToBankValue and goToBankValue variables)
3. Calculate the maxValue between the chaseRobberValue, goToBankValue and goToBankValue variables.
   - (Hint: you can use the Mathf.max() function)

Fuzzy Logic ~ Exercise (4)

Pseudocode Continue…

4. Check whether the maxValue is the chaseRobberValue, if so, call the SetTargetPosToClosestRobber() function.
5. Check whether the maxValue is the goToBankValue, if so, call the SetTargetPosToBank() function.
6. Check whether the maxValue is the patrolValue, if so, call the SetTargetPosToRandomPos() function.
7. Call the MoveToTargetPosition() function to make the police game object move.
    - (Hint: call the function after the if statements)

Fuzzy Logic ~ Conclusion

Output Message when Police Catches all the Robbers

- A simple Fuzzy Logic System can be quite simple to implement and would provide a better representation to human cognition rather than a Boolean Logic System.
- The algorithm which was covered in the following PowerPoint, only focussed on a simple implementation of a Fuzzy Logic System, i.e., the Police only had 3 states. In reality, there are other models which implement more complex Fuzzy Logic Systems which also utilise Defuzzification.
- Through this PowerPoint, the Student would be able to know and identify ways of how Fuzzy Logic can be implemented, as well as its key components, and experiment with different types of Membership Functions.

Fuzzy Logic ~ References

[1] – W. Chai, "DEFINITION fuzzy logic" 2021 [Online]. Available: https://www.techtarget.com/searchenterpriseai/definition/fuzzy-logic [Accessed: 19-Mar-2023]

[2] – Prof. A. Dingli, ICS3209: "Level9_FuzzyLogic"

[3] – K. Chaudhari, "Implementing fuzzy logic to bring AI characters alive in Unity based 3D games" 2018 [Online]. Available: https://hub.packtpub.com/fuzzy-logic-ai-characters-unity-3d-games/ [Accessed: 19-Mar-2023]

[4] – kiyan23, Fuzzy Logic in 3 minutes!, 2011 [Online video]. Available: https://www.youtube.com/watch?v=_YMgaVT3E44 [Accessed: 19-Mar-2023]

[5] – Ansimuz, "Unity Asset Store: GothicVaniaTown" 2017 [Online]. Available: https://assetstore.unity.com/packages/2d/characters/gothicvania-town-101407 [Accessed: 19-Mar-2023]