

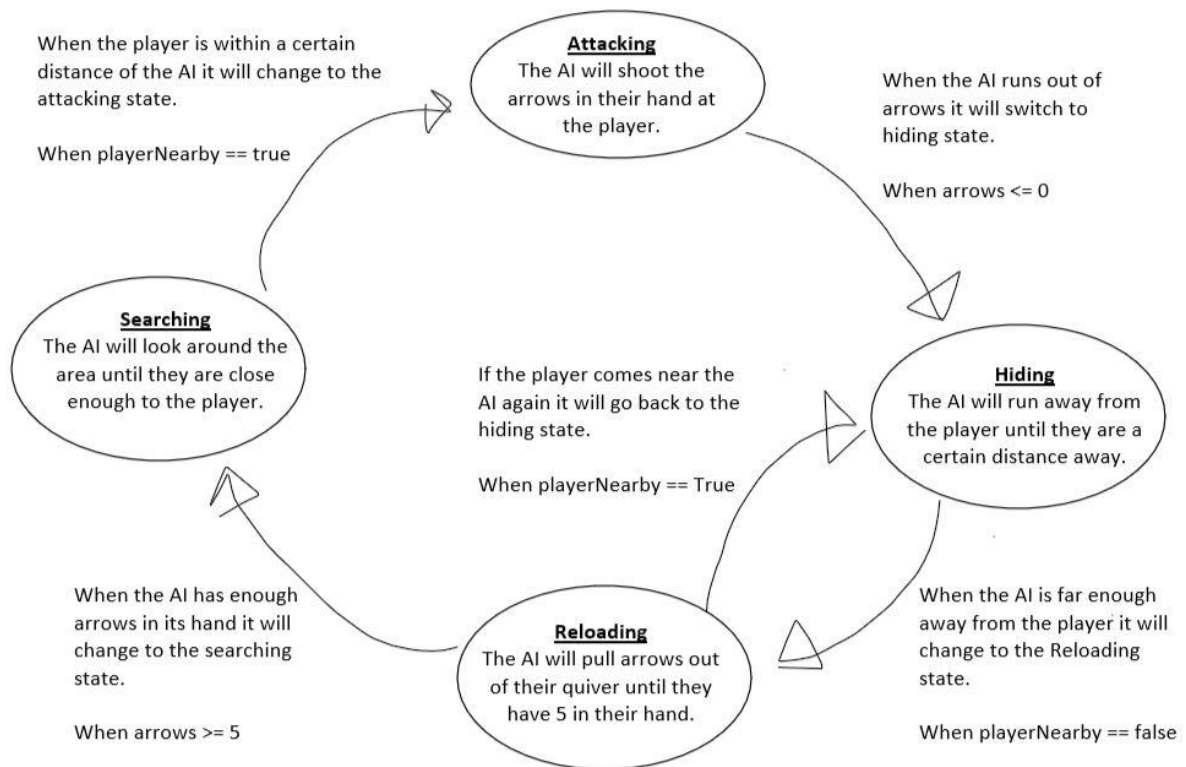
COS30002

Lab Report – Task 2 (7/3/2021)

Ryan Chessum 102564760

- Designed a FSM
- Drew a Diagram for FSM
- Implemented a version of the FSM in python
- Updated the readme to include my name and what the repository was for

Bow and arrow enemy AI FSM



Finite state machine

My idea for an AI is an enemy with a bow and arrow. When the player is nearby, they will attack the player until they are out of arrows in their hand. After they run out, they will start to hide from the player and once they are hiding, they will pull more out of their quiver. Once they have more arrows, they will look for the player again and repeat.

States: Attacking, Hiding, Reloading, Searching

Variables: Arrows (Integer), playerNearby (Boolean)

Code:

```
# Bow and Arrow enemy FSM code - Ryan Chessum 102564760
import random

# variables
arrows = 5
player_nearby = False

states = ['attacking', 'hiding', 'reloading', 'searching']
current_state = 'searching'

running = True
max_limit = 100
game_time = 0

while running:
    game_time += 1

    # Searching: search for a player to shoot at
    if current_state is 'searching':

        # Do things for this state
        print("Searching...")

        rn = random.randint(0, 1)
        if rn == 0:
            player_nearby = False
        if rn == 1:
            player_nearby = True

        # Check for change state
        if player_nearby == True:
            current_state = 'attacking'

    # Attacking: Shoots arrows at the player
    elif current_state is 'attacking':

        # Do things for this state
        print("FIRE!!!!")

        arrows -= 1

        # Check for change state
        if arrows <= 0:
            current_state = 'hiding'

    # Hiding: Runs away until the player is no longer nearby
    elif current_state is 'hiding':
```

```

    # Do things for this state
    print("RUN AWAY!!")

    rn = random.randint(0, 1)
    if rn == 0:
        player_nearby = False
    if rn == 1:
        player_nearby = True

    # Check for change state
    if player_nearby == False:
        current_state = 'reloading'

# Reloading: reloads arrows into hand unless the player comes near
elif current_state is 'reloading':

    # Do things for this state
    print("Reloading...")

    arrows += 1

    rn = random.randint(0, 1)
    if rn == 0:
        player_nearby = False
    if rn == 1:
        player_nearby = True

    # Check for change state
    if arrows >= 5:
        current_state = 'searching'
    elif player_nearby == True:
        current_state = 'hiding'

# check for broken ... :(
else:
    print("AH! BROKEN .... how did you get here?")

# Check for end of game time
if game_time > max_limit:
    running = False

print('-- The End --')

```