



Explanation:

- I have four variables and three states as stated above in the pic.
- The diagram illustrates the transition to each state and forms a cycle
- The initial state is set to idle when the NPC goes on patrol then transitions to attacking or defending state based on the variable values for instance; the NPC checks for nearby enemies and if enemy_spotted value increments to more than 3 it transitions into defending state. On the other hand, if while defending the enemy_spotted drops below 3 it reverts back into ideal state and regains stamina.
- While in attacking state if enemy_strong increments greater than 5 the state is transitioned into defending and the reverts back into attacking if the enemy_weak decrements below 5.

Output:

```
lab2.py x
un: lab2 x
↑
↓
↺
↻
⌂
🗑
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
too weak, I'm going to beat you Ah! ...
too weak, I'm going to beat you Ah! ...
too weak, I'm going to beat you Ah! ...
too weak, I'm going to beat you Ah! ...
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
Terminal Python Console ▶ 4: Run ⚙ 6: TODO
```

```
lab2.py
un: lab2 x
↑
↓
↺
↻
⌂
🗑
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
hm..I'm bored, I want to sleep
too weak, I'm going to beat you Ah! ...
Pretty strong for a short stack!, ah...
-- The End --

Process finished with exit code 0
Terminal Python Console ▶ 4: Run ⚙ 6: TODO
```

CODE:

```
# variables
tired = 0
enemy_spotted = 0
enemy_strong = 0
enemy_weak = 0

states = ['idle', 'attacking', 'defending',]
current_state = 'idle'

alive = True
running = True
max_limit = 100
game_time = 0

while running and alive:
    game_time += 1

    # idle: checks for enemies and tiredness reduces
    if current_state is 'idle':
        # Do things for this state
        print("hm..I'm bored, I want to sleep")
        tired -= 1
        enemy_spotted += 1
        # Check for change state
        if tired < 5:
            current_state = 'attacking'
        if enemy_spotted > 3:
            current_state = 'defending'

    # attacking: gets tired, has to defend
    elif current_state is 'attacking':
        # Do things for this state
        print("too weak, I'm going to beat you Ah! ...")
        tired += 1
        enemy_strong += 1
        # Check for change state
        if tired > 5:
            current_state = 'idle'
        if enemy_strong > 5:
            current_state = 'defending'

    # defending: saving stamina. regains strength, kills weak enemy and patrol for
    more
    elif current_state is 'defending':
        # Do things for this state
        print("Pretty strong for a short stack!, ah...")
        enemy_spotted -= 1
        enemy_weak -= 1

        # Check for change state
        if enemy_weak < 5:
            current_state = 'attacking'
        if enemy_spotted < 3:
            current_state = 'idle'
```