

Name: \_\_\_\_\_

## Seek Algorithm

1. In class, you implemented a nearly-functional testbed for working with AI. In addition, you implemented the seek AI algorithm, the first of movement algorithms. In this homework, you will now tweak this algorithm to handle some of its quirks.

- 20 (a) Currently, there is a hysteresis issue with the girl character in that it constantly overshoots the target but never exactly reaches it. In linear systems, this hysteresis exhibits itself through a general form equation:

$$Y(t) = \chi_i X(t) + \int_{-\infty}^t \Phi_d(t - \tau) H(\tau) d\tau \quad (1)$$

where  $\chi_i$  is the instantaneous response and  $\Phi_d(t - \tau)$  is the impulse response at time  $t$  to an impulse at time  $\tau$ .

In game AI, this (and other control systems techniques) are difficult to implement. Instead we use simple thresholding to stop the seek algorithm when the agent is within  $n$  pixels of the target. For your homework, set  $n = 10$ . That is, if you are within 10 pixels to the target, you need no longer seek it.

- 10 (b) The game currently has an issue where the avatars are allowed to go past the edge to screen. You should treat the edges of the screens as walls; no avatars should be able to go past the edge of the screen, nor should they be able to wrap around. **Hint:** You can simply use the clipping techniques described in class (such as with color) to prevent the avatars from going past the boundaries of the screen.
- 15 (c) Add a second avatar, `Character Cat Girl`, to the screen at location (300, 300) with a speed of 50. The avatar should use the seek algorithm. Add a third avatar, `Character Horn Girl`, at location (400, 400) with a speed of 100, having the same AI behavior.
- 20 (d) Finally, add a fourth avatar, `Character Princess Girl`. This avatar should implement the flee algorithm, rather than the seek algorithm. Place the avatar at location (90, 90) with a speed of 10. In all likelihood, the Princess Girl will very shortly get stuck at one of the walls. For now, this is okay, since we have not yet talked about obstacle avoidance.

- 10 2. Submit the file `game.py`, `avatar.py`, and all of your sprites in a zip archive on Moodle.

Question	Points	Score
1	65	
2	10	
Total:	75	