

Lesson 5 - Artificial Intelligence

<http://www.stencyl.com/learn/act5/>

In this activity, you'll write some AI (artificial intelligence) for some enemies.

1 Get Started

- After launching Stencyl, open up **Project 5**.
(Can't find Project 5? Visit <http://www.stencyl.com/learn/downloads/> to get it)
- **Run the game.** You'll see our Hero (who can move) and an enemy that does nothing.



2 Add an Enemy

- Open up the Actor Type called **Enemy**.
- Make it so that he will **kill the player upon contact**. *(Hint: Use a Collision event)*
- Run the game.

3 Follow the Leader

Make it so that the Enemy will constantly follow the Hero.

Hints

- Use the "push towards xDir/yDir" block and tell it to push sharply at 5 force.
- What information do you need to calculate this? You need to know the Hero's position and the Enemy's position.
- How do you get the Hero's position? The Enemy doesn't know this. Given what you've learned, is there a way to get the position? *(Answer: Use Game Attributes)*

You'll find that it'll take multiple tries to get this right. As always, we recommend coding a little bit and then running the game rather than trying to get the entire script correct at once.

At the end, you'll have an enemy that blindly follows the Hero.

But what happens if there's a wall between the Hero and the enemy? This AI is not smart enough to walk around the wall. The final challenge in this session will help you make a smarter AI.

4 Avoid

- First, drag out the blocks you just wrote and set them aside. We'll reuse them in Step 5.
- Let's now do the opposite of what you did in Step 3. Make the enemy **avoid** the player.

Improving an AI

Making a simple avoid script requires a few simple edits. However, upon testing, you'll find that the enemy can get stuck in one of the corners fairly easily.

Using basic math, can you think of a smarter avoid AI that doesn't fall into this trap? This is the subject of one of the extra activities.

5 State Switching

Our enemy is a bit boring. Instead of doing just one thing, let's have the enemy switch between Following and Avoiding every 10 seconds.

You'll want to use the **[switch animation] block** and then modify the code, so that it runs the follow / avoid AI's when those animation states are active.

You'll also need an event to perform the animation switches.

Hint

You'll be using the following blocks in your solution. Type in the animation's name into the anim attribute field for the switch animation block, which in this case will be either **follow** or **avoid**.

A Scratch 'switch animation to' block. The text 'switch animation to' is on the left. In the middle, there is a text input field containing 'anim attribute'. To the right of the input field is the word 'for' followed by a dropdown menu showing 'Self' with a small downward arrow.A Scratch 'current animation for' block. The text 'current animation for' is on the left, followed by a dropdown menu showing 'Self' with a small downward arrow.

Now that you've created some simple AI's, put together everything you've learned so far to complete the extra activities, which involve creating more complex AI's.