While: Indefinite Loops

Alice
Overview

- Why we need indefinite loops
- Indefinite loops: While statement
- Example: Shark chase
  - Random motion
  - How do we know the chase ends?
- Infinite While loop
Loops

In Alice, a Loop instruction requires a count of the number of times a block of instructions should be repeated.

What if we don’t know ahead of time what the count should be?

For example, in a board game like chess or checkers, we don’t know how many moves it will take for a player to win or lose the game.
Indefinite Loops

In programs where a count of repetitions is not known, an indefinite loop can be created using the *While* statement.

While some condition is true perform instruction(s)
To write a While statement, we need to know the condition that determines whether the loop will be repeated.
Example

A shark is hungry for a seafood dinner. The shark is going to chase after and catch a fleeing goldfish.

This is a typical chase scene, popular in action films.
Problem

The problem is how do we get the shark to chase the goldfish in a chase-like action?

- The shark should not immediately catch the goldfish (otherwise, there would be no chase)
- Any goldfish with self-preservation instincts should appear to be fleeing
Solution

To create a chase scene,
- the shark swims a short distance toward the fish and, at the same time, the fish swims a short distance away from the shark.
- the fish flees to a random (but nearby) location.
- as long the goldfish is still 0.5 meters away from the shark, we want to repeat the actions.
Storyboard

While the goldfish is more than 0.5 meters away from the shark

Do in order

Point the shark at the goldfish

Do together

Shark swim towards the goldfish
Goldfish flee away from the shark

The shark eats the goldfish
Methods must be written.
Altogether, the shark swims 0.4 meters forward
Flee with random motion

Making the goldfish flee to a random, but nearby location requires the use of a

- *random number* question
- *move to* instruction (object moves to a specific position in the world)

See Tips & Techniques 6 for details on the *random number* question and *move to* instruction.
A random number question is used to adjust a location value.

Range of values is -0.2 to 0.2, which limits the movement to a nearby location.
Goldfish.flee

<table>
<thead>
<tr>
<th>Do together</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goldfish.Tail</strong></td>
<td>turn left</td>
</tr>
<tr>
<td><strong>Goldfish.Tail</strong></td>
<td>turn right</td>
</tr>
<tr>
<td><strong>Goldfish.Tail</strong></td>
<td>turn left</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>move to</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goldfish</strong></td>
<td>move to</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Shark will catch goldfish

How do we know the shark will eventually catch the goldfish?

- shark always moves 0.4 meters toward the goldfish
- goldfish movement is restricted to 0.2 meters in any direction. Geometrically, the fish can never move more than 0.35 meters away
- The shark has a distance advantage and will eventually catch up. The loop will end.
General “Rule of Thumb”

As a general rule, a *While* loop should be written so the loop will eventually end.

- Requires that statements within the loop change the conditions of the world such that the condition for the *While* statement will eventually become false.

- If the *While* loop never ends, it is an **infinite while** loop.
Intentional Infinite Loop

There are times when we want to write a loop that continues forever.

In the shark chase example, each time the shark catches a fish, a “new fish” could appear in the scene and the shark could catch it. (Typical arcade game situation.)
While true

To make a while loop infinite, write the condition as “true”
Demo: Goldfish.reappear

To create a “new fish”, make the old fish reappear in a new location.
Assignment

Read chapter 7 section 2

- Indefinite While loops
- Chase example
  - (You may want to re-read random number and move to from Tips & Techniques 6)
- Infinite While

Read Tips & Techniques 7