

CompSci 4
Chap 7 Sec 2
Oct 25, 2007



Prof. Susan Rodger

Announcements

- Read Chapter 9.1 for next time
- Assignment 6 due Nov 6
- Today
 - Lecture on Chap 7 Sec 2 and Tips and Tech.
 - While loop – indefinite loop
 - Event Loops

Last time -Loop – definite number

- What happens when this code runs?

world.my first method *No parameters*

The image shows a Scratch code editor with the following blocks:

- 123 number = 1**
- number** set value to **6** more...
- Loop** **number** times **time** **show complicated version**
 - Chicken** move **up** **number** meters more...
 - Chicken** move **down** **number** meters more...
- Loop** **123** **index** from **0** up to (but not including) **number** times **incrementing by 1**
 - Chicken** move **up** **index** meters more...
 - Chicken** move **down** **index** meters more...
- Loop** **123** **index** from **0** up to (but not including) **number** times **incrementing by 2**
 - Chicken** move **up** **index** meters more...
 - Chicken** move **down** **index** meters more...

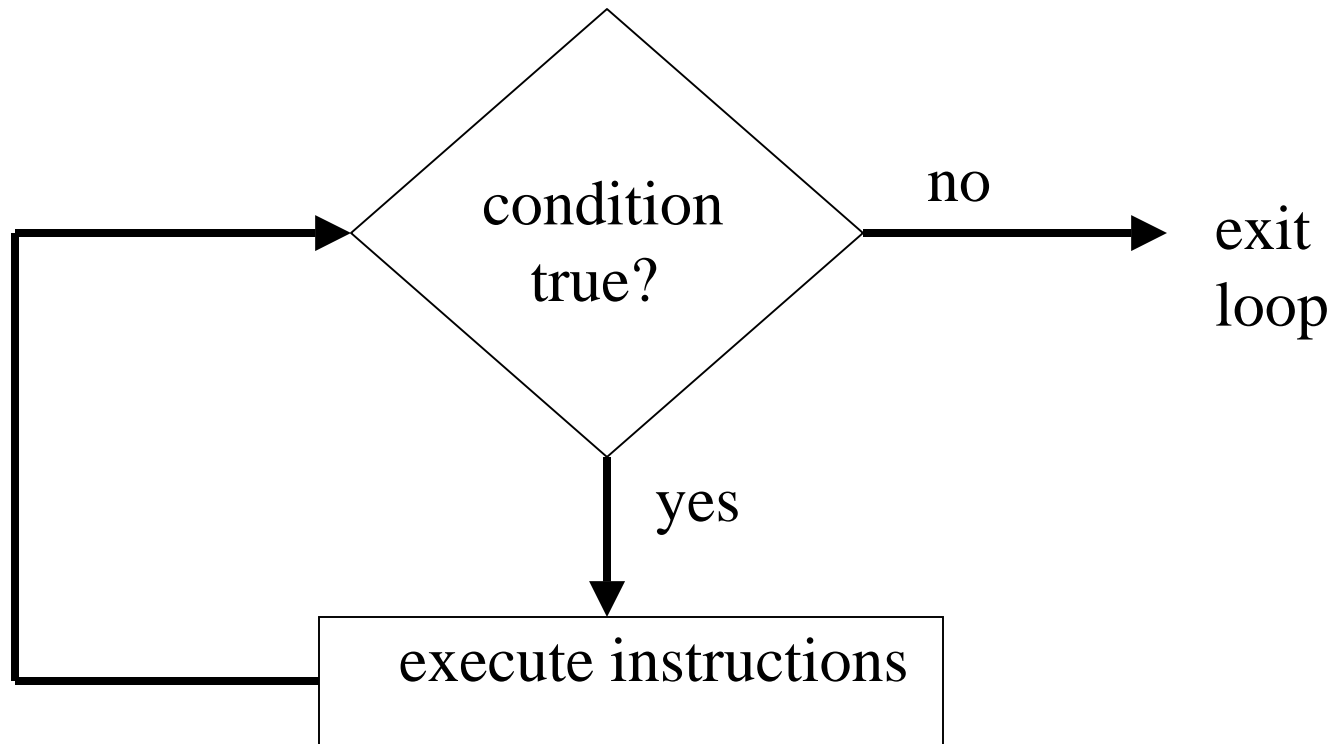
Repetition

- Sometimes don't know exactly how many times a set of instructions are repeated.
- Stopping is based on a condition
- Example:
 - Game of Chess, how many moves until win
 - Stop: when markers are in check mate position

Indefinite Repetition

- In programs where number of repetitions not known in advance, can use
 - While statement

While statement



- While some condition is true
 - execute instructions

Example

- Common feature in popular “action films” is a chase scene
- Example: hungry shark chasing fleeing goldfish
 - Repeat: fish swim away from shark, and shark swim toward fish
 - Shark swim distance a little more than fish swim distance
 - Eventually, shark will catch up with fish and eat fish

Storyboard

World.chase

While goldfish more than .5 meters from shark

Do in order

shark point at goldfish

Do together

shark swim (toward goldfish)

goldfish flee (away from shark)

shark eat goldfish

shark.Swim, shark.eat and goldfish.flee in book

World.chase

World.chase *No parameters*

No variables

- Do in order**
 - While**
 - goldfish distance in front of shark more... > 0.5
 - Do in order**
 - shark point at goldfish duration = 0 seconds style = abruptly more...
 - Do together**
 - shark.swim
 - goldfish.flee
 - shark.eat what = goldfish

Shark will catch goldfish

- How do you know the shark will eventually catch the goldfish?
 - Shark always moves 0.4 meters towards goldfish
 - Goldfish moves randomly away from shark at most .2 meters
 - Shark 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 statements inside the loop change the conditions of the world such that condition for While eventually becomes false
- If While loop never ends
 - Infinite while loop

Practice – From Bunny eats Broccoli



Wait 2 seconds ▾

bunny ▾ turn to face broccoli ▾ more... ▾

☐ Loop 6 times ▾ times [show complicated version](#)

bunny.hop

World.jumpUpAndDown item = bunny2 ▾

- Replace with while loop, bunny hops until close to broccoli (< 1 meter)

Using While with Events

- Create new event “while something is true”



- Result

We call this
a "BDE"



BDE – Begin During End

- Event – actions occur at different times
- When the event is first true
 - Begin action
- While the event is still true
 - During action repeats
- When the event condition is false
 - End action occurs

Example – Penguins meeting

- While 2 penguins are > 3 meters apart
 - Turn to face each other (Begin)
 - Move towards each other repeatedly (During)
 - Bow to each other (End)



Penguins Event Code

Events create new event

When the world starts, do world.my first method

While penguin distance to penguin2 > 3

Begin: world.PenguinsFaceEachOther

During: world.PenguinsMoveForward distance = 1

End: world.PenguinsEnding

Penguin Begin and During parts

world.PenguinsFaceEachOther *No parameters*

No variables

☐ Do together

penguin ▾

turn to face

penguin2 ▾

more... ▾

penguin2 ▾

turn to face

penguin ▾

more... ▾

world.PenguinsMoveForward distance

No variables

☐ Do together

penguin ▾

move

forward ▾

distance meters ▾

more... ▾

penguin2 ▾

move

forward ▾

distance meters ▾

more... ▾

Penguin End Part

world.PenguinsEnding *No parameters*

No variables

☐ Do in order

☐ Do together

penguin ▾ turn forward ▾ .12 revolutions ▾ more... ▾

penguin2 ▾ turn forward ▾ 0.12 revolutions ▾ more... ▾

☐ Do together

penguin ▾ turn backward ▾ .12 revolutions ▾ more... ▾

penguin2 ▾ turn backward ▾ 0.12 revolutions ▾ more... ▾



How do we get this event to
happen again?

Classwork

- World 1 - Modify game from last time
- World 2 - Start a new world with snow background
 - Add a penguin and a fish resting on the ice
 - Create a BDE that moves the penguin over to the fish and eats it (fish disappears)

