CompSci 94
While loops, randomness
October 18, 2018

Prof. Susan Rodger
Class Today

• Looping based on a condition – while loop
• Randomness
  – Random variable
  – Making choices based on a random number
Setup for Lecture today

• Put in the following objects:
  – Flyer: Chicken
  – Props: pumpkin, cake, colaBottle, pricklyPearCactus
Story for lecture

• The chicken will walk over to each item and eat it.
• One of the items will randomly move around to make it harder to
Fancy walk –

Taught my chicken how to strut

• Take one step with legs and neck moving…
Looping – exact number of times

• Count loop

• How many steps to get to the cake?
  – 3, 6, 8?
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
  – Where number of repetitions not known in advance
  – Use **while** statement
While statement

- While some condition is true
  - execute instructions

CompSci 94 Fall 2018
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
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
While – walk towards until close

- While condition is true – keep going
- Must have an update inside the loop
EatItem Finish up – eat the item

• After loop, eat the item
Random Numbers

• Random numbers are used in certain kinds of computer programs

• Examples
  – Security for web applications
  – Encryption for satellite transmissions
  – Gaming programs

• We will look at examples of using random numbers in animations

• Example – variable assigned a random number between 0.0 and 1.0
Where is the randomness? How many random numbers are we using?

• This code is inside the while...
Where is the randomness? How many random numbers are we using?

• The variable amount has a value between 0 and 1.
  – Four ways to move – assign them based on the value of amount
  – If amount is 0 to 0.25 – move left
  – If amount is .25 to .50 – move right
  – If amount is .50 to .75 – move forward
  – If amount is .75 to 1.0 – move backward
Random number is spread out around 0.0 and 1.0

- We want to try to have each direction choice equally likely, so we split the possible values into four parts.
Where is the randomness? How many random numbers are we using?

- Other randomness? Yes!
- There is randomness in how far to move
- Two types of randomness
  - 1) randomness in which way to move
  - 2) randomness in how far to move
    - We aren’t saving this one in a variable but we could (see alternate solution on later slides)
  - Note these are DIFFERENT Numbers – note the different types of boxes for the two numbers on the next slide
Final Code eatMovingItem

```plaintext
declare procedure eatMovingItem with parameter: SJointedModel, food, Add Parameter...
do in order
  this turnToFace food add detail
  while this getDistanceTo food > = 0.5 is true
  this strut
    DecimalNumber amount <- nextRandomRealNumberInRange 0.0, 1.0
    if amount < 0.25 is true then
      food move LEFT nextRandomRealNumberInRange 0.0, 0.75 duration 0.5 add detail
    else
      if amount < 0.5 is true then
        food move RIGHT nextRandomRealNumberInRange 0.0, 0.75 duration 0.5 add detail
      else
        if amount < 0.75 is true then
          food move FORWARD nextRandomRealNumberInRange 0.0, 0.75 duration 0.5
        else
          food move BACKWARD nextRandomRealNumberInRange 0.0, 0.75 duration 0.5
    loop
```

Final Code (the rest of eatMovingItem)
Another way – use two random number variables

- Variable 1 - store the random number between 0 and 1 in amount and use that number to determine which direction to move (we already do this)

- Variable 2 – store a random number between 0 and .75 in a variable named howFarToMove and then use that variable for the distance to move.
eatMovingItem (version 2, first part of code)
Final Code MyFirstMethod
This lecture covered

• While loops – loop that continues based on a condition
  – When the condition is false it halts
  – There must be some kind of update in the loop

• Random numbers