

UNIVERSITY of WISCONSIN-MADISON  
Computer Sciences Department

CS 202: Introduction to Computation Professor Andrea Arpaci-Dusseau

## How can computation... create animated stories?

© Original Artist.  
Reproduction rights obtainable from  
www.CartoonStock.com



search ID: 1670044

## Old Checkup

What happens if you don't specify the initial state of your program?  
Which are likely to be used for initialization?



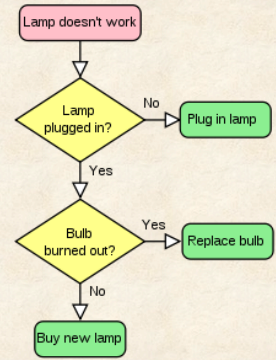
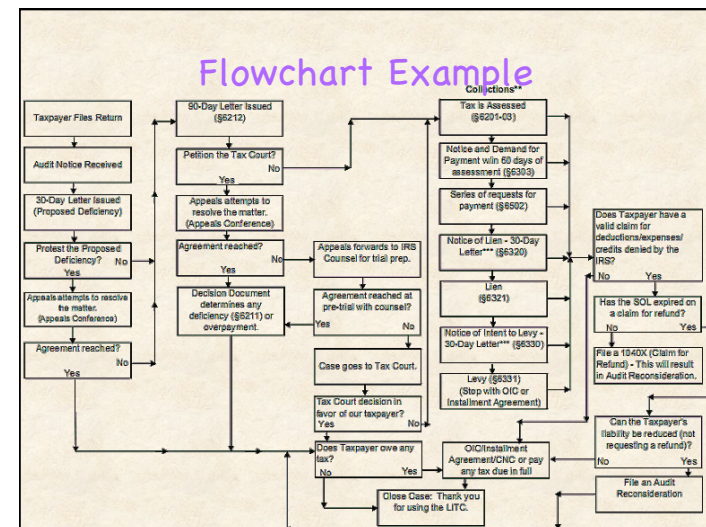
## Help Organize Program: Flowcharts

**Flowchart:**

- Visual representation of steps of algorithm
- Summarizes how algorithm behaves given specific answers

**Symbols**

- Boxes: Represent states (or actions or actions)
- Arrows (or edges): Show transitions (or decisions) between states

## Flowchart for Animated Story

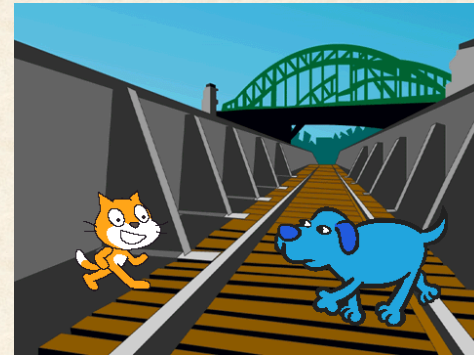
Animated Story: Behaves the same every time

- No decisions!
- Flowchart is summary of action of story

How to create flowchart?

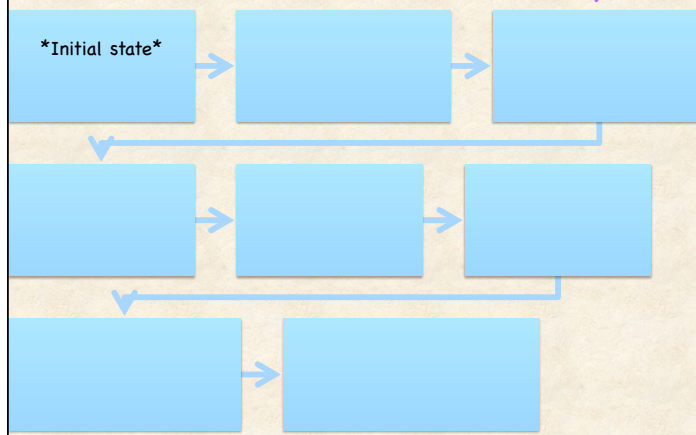
- Identify Initial State or Scene
  - Somewhat subjective (no right answer)
  - Label with descriptive name
- Identify characters of story
  - Specify actions of each character in scene
- Connect scenes sequentially

## Animated Story

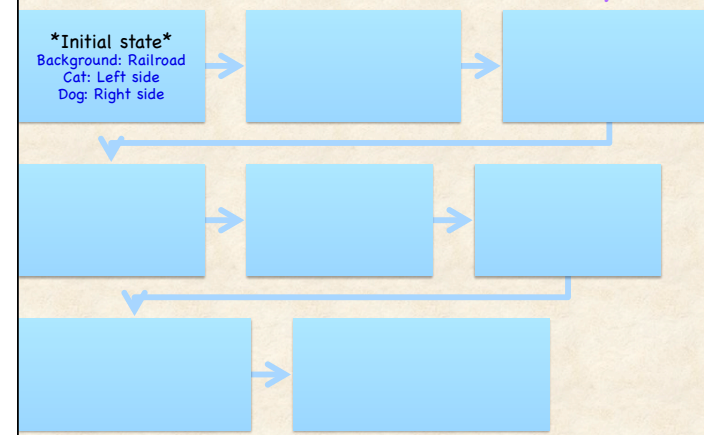


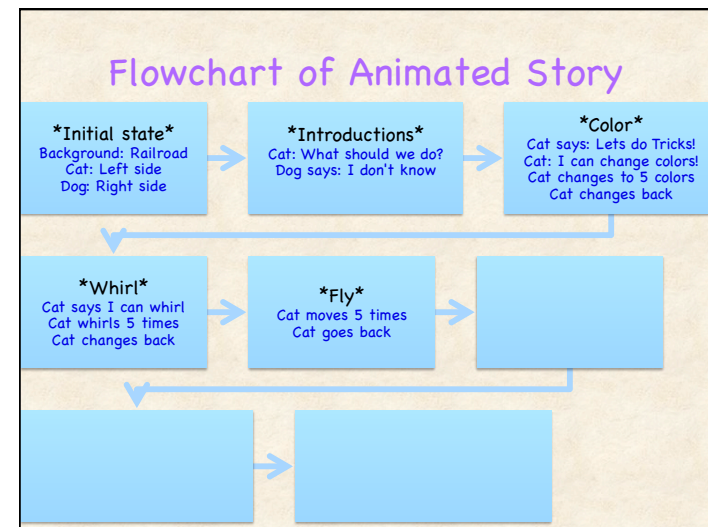
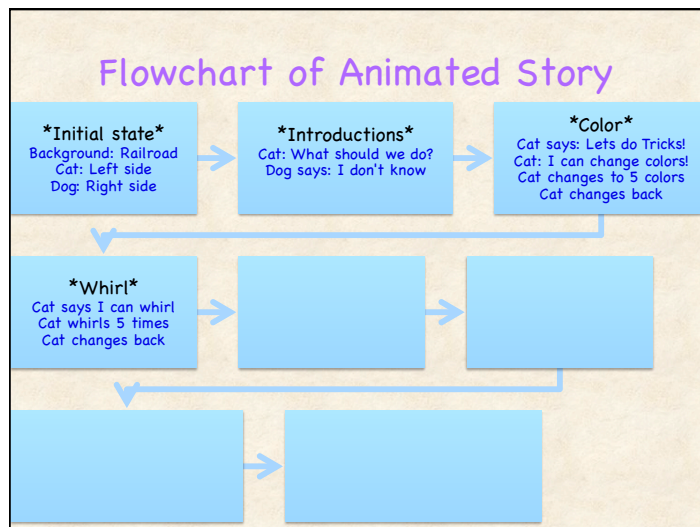
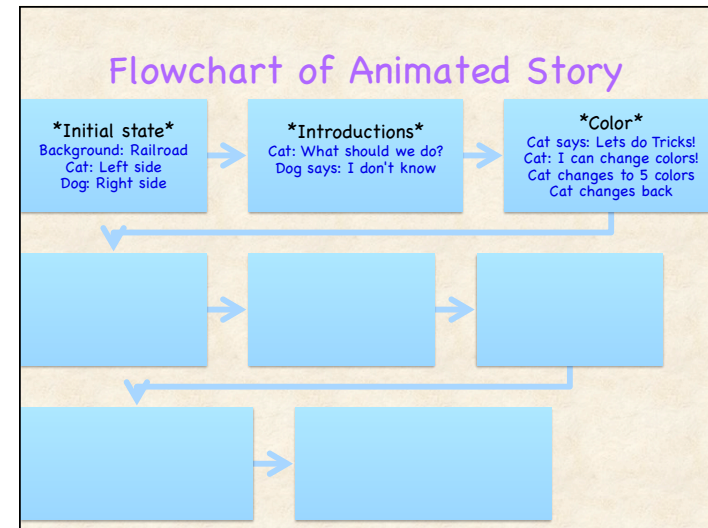
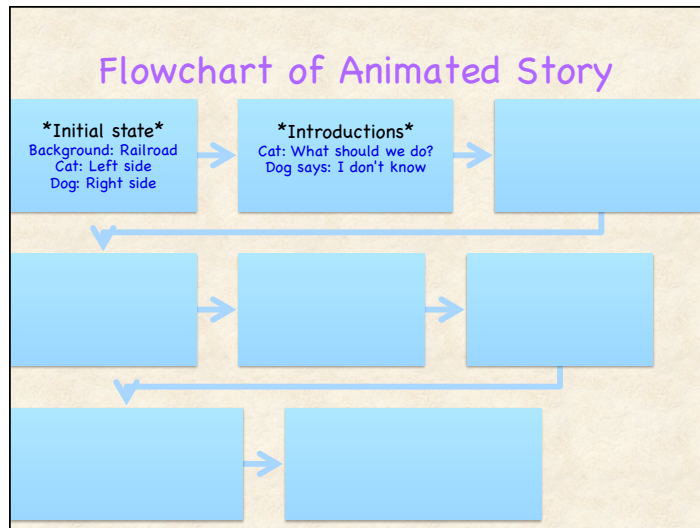
See Scratch code posted on web site!

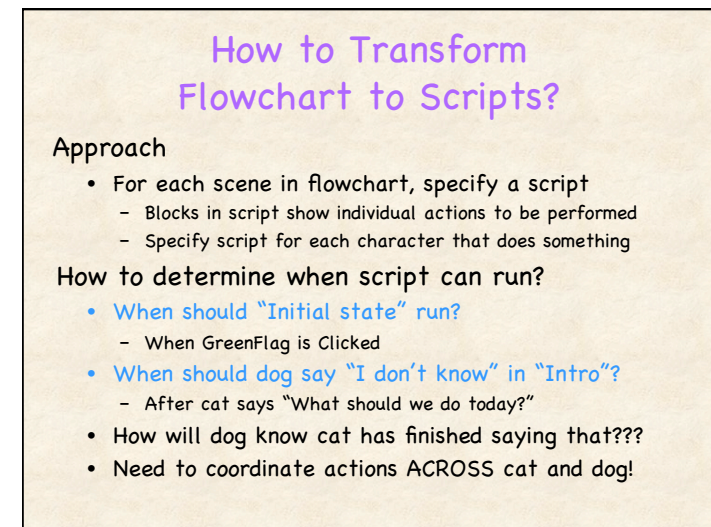
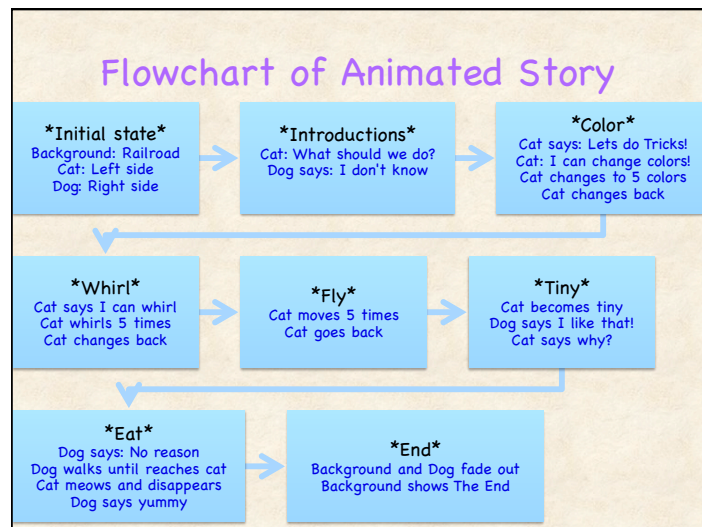
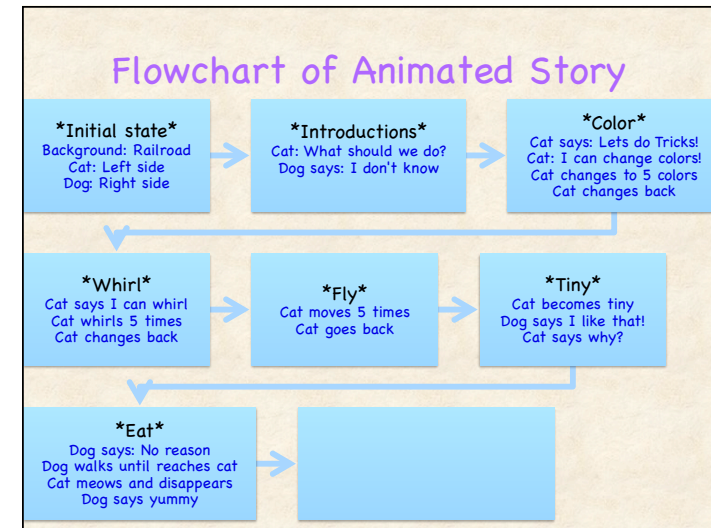
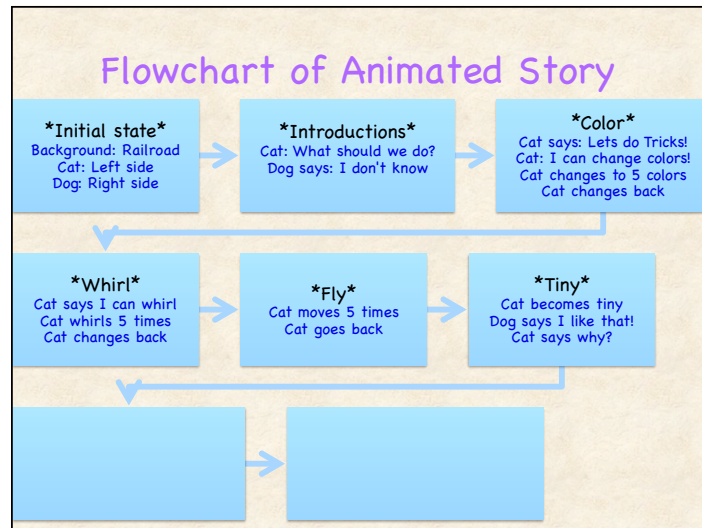
## Flowchart of Animated Story



## Flowchart of Animated Story









## How to Tell Another Script to Run?

**broadcast** sends the message "jump" to all sprites

**broadcast** sends a message to all the sprites (and the background)

This is useful if you want to tell other sprites when to do something.  
What do you want them to do when they receive the message?

See **when I receive**

## How to Run Desired Script?

**when I receive** whenever the message "jump" is broadcast

do this

change y by 40  
wait 0.5 secs  
change y by -40  
wait 0.5 secs

**when I receive** waits for the message **broadcast**

## Wait for Scripts to Complete!

**broadcast** and wait send the message "jump" and wait until all the jumps are done then do this

say That was fun!

**when I receive** whenever the message "jump" is sent do this

change y by 50  
wait 1 secs  
change y by -50

You can use **broadcast** and wait to send a message to all sprites to tell them to do something, and wait until they all finish before continuing.

**broadcast** and wait Click to choose which message gets sent.

**new** Choose "new" to type in your own message.

## Beauty of Abstraction

Abstraction: Separation of high-level view of entity from low-level details of implementation

When sender broadcasts "jump", doesn't know how "jump" is implemented by different Sprites

Why is this good?

- Simplifies concerns of sender (don't need to know everything)
- Can change implementation of "jump"

Of course, receiver might not implement "jump"!

## Naming Convention for Messages

Use good descriptive names

- Purpose of names = help others understand your code
- Suggestion: Name matches name of scene
  - Intro, Color, Whirl, Fly, Tiny, Eat, Fadeout

Problem:

Hard to follow flow of messages across Sprites

Solution:

1. Use Stage to control action as much as possible
2. Use naming convention to help understanding
  - SceneName : Receiver
  - e.g., "Eat : Dog"

## Develop Code Now

## Programming Concepts

General

- Divide high-level functionality into logical units
- Descriptive naming is important
- Initial state must be specified
- Incrementally test code as you go
- Scripts must be activated to run
  - When flag clicked; When receive message
- Execution within script proceeds sequentially
- Control : forever, repeat <times>, repeat until
- Parameters (to blocks) specify behavior
- Goal is to make "non-fragile" code

## Check-Up

- In your animated story all the scripts are running at the same time. What did you do wrong?
- Can a Sprite receive a message it broadcast?
- When will this code work correctly? when not?

