

Session 1

Intermediate/Advanced Session Notes

What's a statement about Scratch

- I know they can do more than just games
- Scratch helps students make connections
- Scratch does not really expose them to the actual coding language/kids aren't "coding" with Scratch
- Scratch is a tool
- Scratch is not suitable for K2
- Scratch is valuable, but takes a lot of time
- Scratch users don't document their
- Sometimes hard to figure out what to do next
- Scratch as an infinite number of outcomes
- On 2.0, all the awesome Minecraft, etc. adult-created content can be distracting and disappointment for students
- We should have a grown up session of Scratch
- Some kids excel at Scratch while others lag behind
- Scratch has potential for Mathematical research
- Scratch catalyzes an unexpected interest in coding
- Scratch is a platform for creativity that teaches logical thinking

Round 2 - one word

- Connections
- Coding
- Inventive
- Creative
- Differentiation
- time
- documenting
- fun
- learning
- infinite
- differentiation
- research
- catalyst
- logical

Popcorn Round

- Inventive
 - Can do more than games
 - Need to make something functional
- Using Scratch as a reward - but then he gets to play games

- Well, not really programming - how do we clear up the misconception with other teachers
- Teacher Education - take curriculum from classrooms
 - Presentation with Scratch about Canada
 - Maps
 - Research
 - Public Speaking/Presentation
 - Group Work
- Different from other things, can use it to talk about causality which is really important in Scratch
- Models
 - Science modeling
 - and modeling the programming
 - Statistics modeling
- Ownership
 - Comes for free with Scratch
 - Tool for knowledge building
- What's the difference between a game and a model/simulation?
- Computers as a tool, then we need to teach computers, now back to a tool
- Programming to learn, and programming to teach other kids
- Identifying concepts to both students and teachers
- How can we use programming to solve problems?
 - Game
 - Have to have an understanding
 - Model/Simulation
 - You don't know what's going to happen
- Programming is teaching you how to think
- How can you be more efficient
 - Fewest number of sprites/fewest number of lines of code
- How can you use the model to try out different things?
 - Model could be in the form of a game
- Backpack
- Scrolling
 - Key movements change the background