

Learning Trajectories

Scaffolding

CMSC 19929: Creating Computer Games for Learning

Reminder

- Readings responses are due prior to lecture (9AM), **not accepted late for credit** (unless hospitalization, etc.)
- Coding practice exercises are due at 11:59PM (same day as when they are released in class)

Game Design Project

- Work in group of 2-3
 - 1 with special permission (not recommend but you're free to come to me after lecture to discuss this)
- Choose a subject to teach
- Choose what level to teach (3rd – 9th grade)
- Main target must be different from yourself in meaningful ways (age, interests, opportunities, culture, language, etc.)

We (sort of) practiced the last three bullet points yesterday, and will do more practice today to get you ready for this!

Game Design Project

- **Design**

- Complete design!
- No resource constraints (time, \$)

- **Implementation** (i.e., actually code the game)

- Agreed-upon subset of full design
 - All your group members will need to agree upon whether to they want to implement the game for extra credit or not
- Reasonable to complete by end of Week 3
 - On July 18, you'll need to submit an MVP description alongside your Final Game Design document (Milestone #3) to get my approval to implement your MVP
- Illustrates the design of the game and principles of course (EDUCATIONAL!)

Content Attributes

- **Must have**

- A character that moves
- A character that interacts with other objects on screen
multiple “scenes” (background)
- Opening / welcome screen with multiple choices (e.g.,
About, Play)
- Internal screen with game

Content Attributes

- **Cannot be**
 - memorization (e.g. flashcards)
 - teach through “quiz” teaching things that non-learning games often teach, for example:
 - Physics of projectiles (e.g., angry birds)
 - Finances through running a farm or restaurant (e.g., making money by harvesting / spending money to buys stocks for the farm)

Important Due Dates

- **Milestone #0:** Finalize your group (today!)
 - Do people need more time for this?
- **Milestone #1:** Target Users and Learning Content (Sat 7/11)
- **Milestone #2:** Basic Game Design (Tue 7/14)
- **Milestone #3:** Final Game Design and (optional) MVP Description (Sat 7/18)
- **Milestone #4:** MVP Implementation (Thu 7/23) - extra credit

Project Milestone #1

- Choose subject / content area
- Choose age range (narrow, lacking important general skills)
- Define three user profiles
- **Choose specific learning goals** (today lecture!)
- **Create a learning trajectory** (today lecture!)

Basic Learning Strategies

Incrementally teach material

Build on prior knowledge (in school and at home)

What is a Learning Trajectory?

A **learning trajectory** (LT) is a path from students' existing knowledge to some particular **learning goal** (LG).

One useful way to conceptualize an LT is to think of it as having three components:

1. An overarching learning goal;
2. A (partially) ordered list of incremental learning goals that suggest a pathway to the main learning goal; and
3. A set of learning activities that help students move along the path.

We will focus on **learning goals** and their **relationships**

Learning Goals

A sentence describing a single “nugget” of learning that can be **assessed**.

- Knowledge - something they can answer questions about
- Skill - something they can do

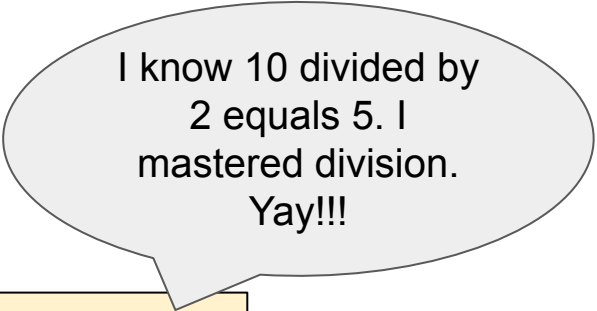
Is “Algebra” a learning goal?

Nope! It’s a subject or a branch in Math (if you want to consider Math as a major subject instead of Algebra).

Learning Goals

A sentence describing a single “nugget” of learning that can be **assessed**.

- Knowledge - something they can answer questions about
- Skill - something they can do



I know 10 divided by 2 equals 5. I mastered division. Yay!!!

Is “Division” a learning goal?

Nope! It's getting closer, but division is a topic in algebra, still too broad to be assessed.

Let's brainstorm some **LGs for division!**

- (addition, multiplication)
- simple division (divide number < 100 with integer result, no remainder)
- conversion to fraction
- ability to perform long division w/o remainder
- ability to perform long division with remainder
- simple polynomial division w/o remainder
- simple polynomial division with remainder
- complex number division

What is a Learning Trajectory?

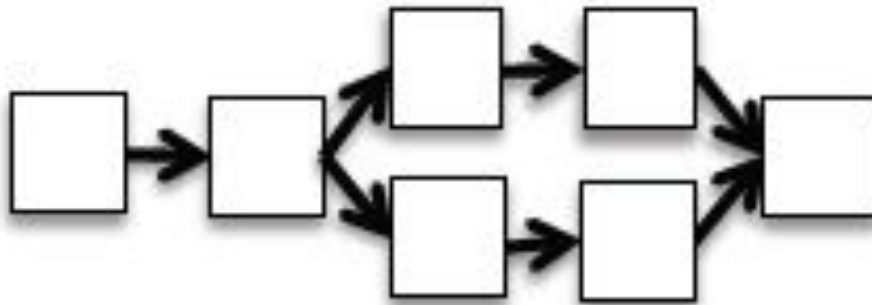
A **learning trajectory** (LT) is a path from students' existing knowledge to some particular **learning goal** (LG).

One useful way to conceptualize an LT is to think of it as having three components:

1. An overarching learning goal;
2. A (partially) **ordered list** of incremental learning goals that suggest a **pathway** to the main learning goal; and
3. A set of learning activities that help students move along the path.

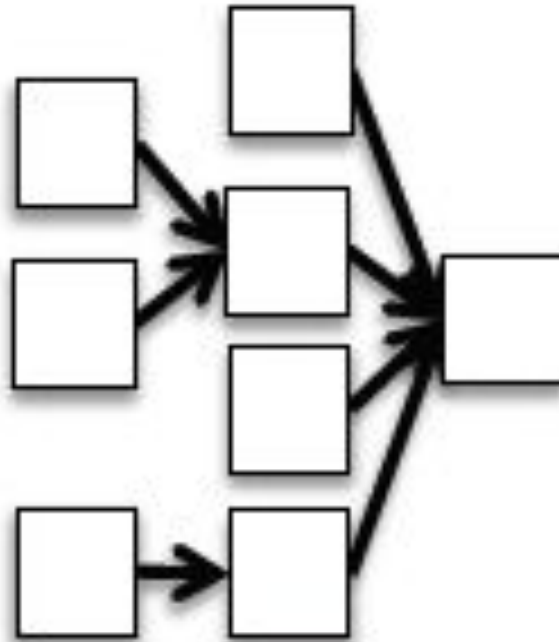
How do we create a pathway to the main learning goal?

How does theory influence Learning Trajectory shape?



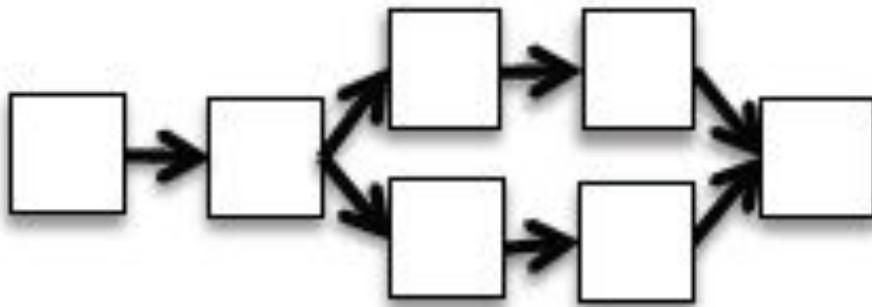
(a) Learning Progression

How does theory influence Learning Trajectory shape?

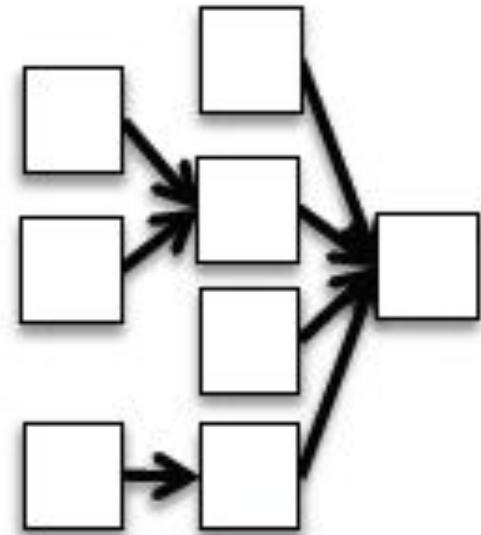


(b) Pieces of Knowledge

How does theory influence Learning Trajectory shape?



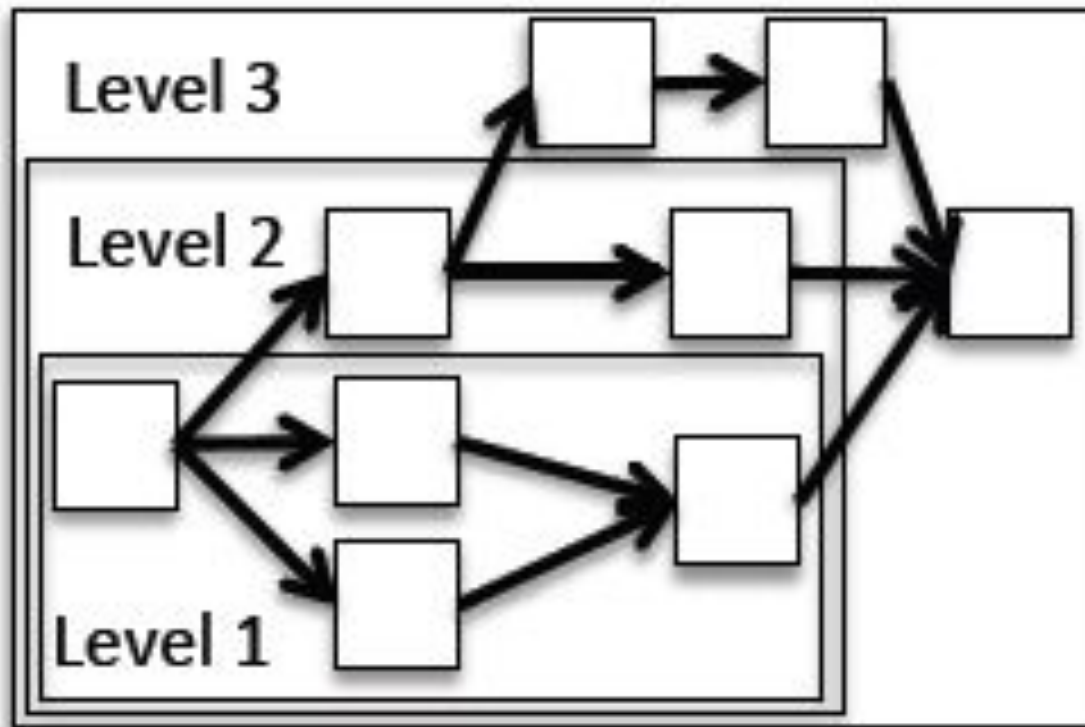
(a) Learning Progression



(b) Pieces of Knowledge

What are the general differences or similarities of (a) and (b)?

How does theory influence Learning Trajectory shape?



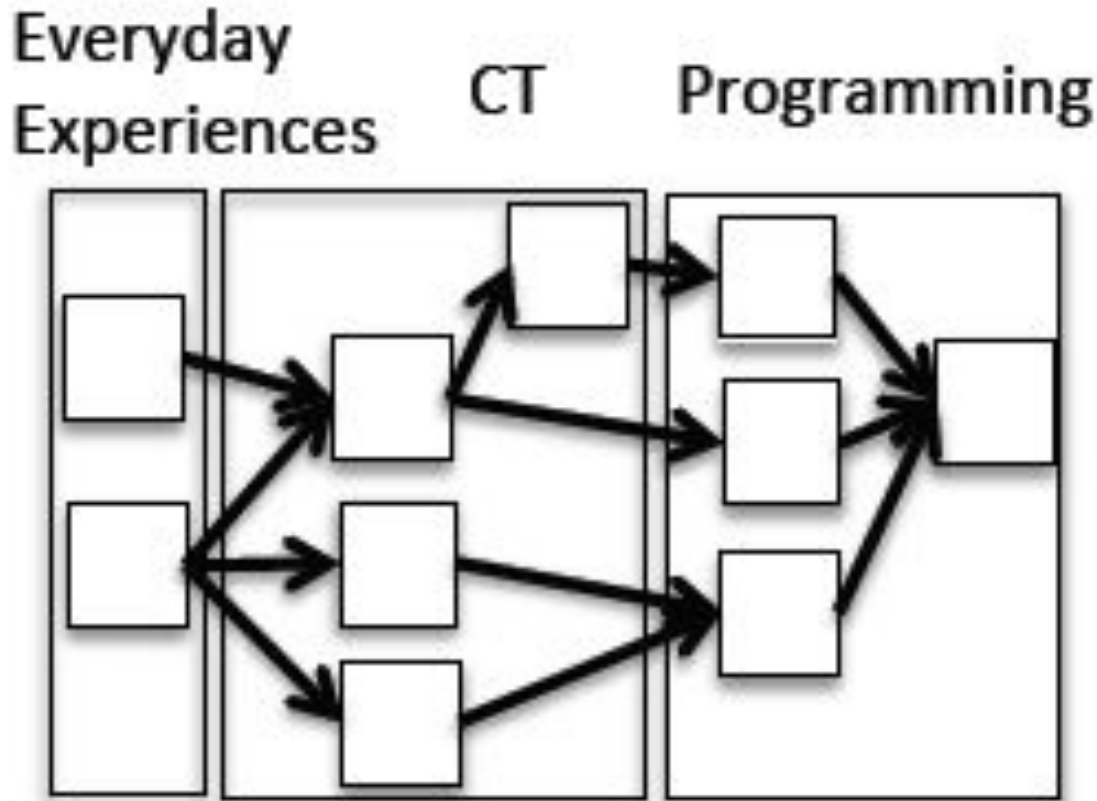
(c) Spiral Curriculum

Spiral Curriculum – 3 key aspects

- Same concepts revisited
- Revisited in more depth
- Connections made to previous visit

How does theory influence Learning Trajectory shape?

CT = computational thinking



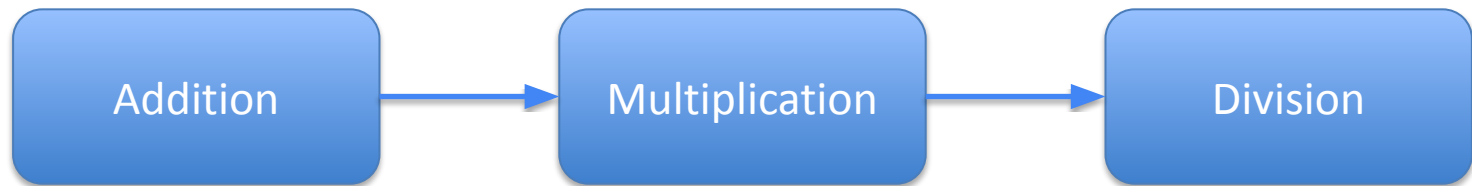
(d) Constructivism

Learning Trajectory Purposes

- Content, not teaching method
- Provides ***possible orderings*** for presenting material that builds upon itself
- Identifies points of understanding to focus on before going to next level

Let's make an LT for division

- Learning Goal (endpoint)
- Prior knowledge necessary / helpful



Is this useful?

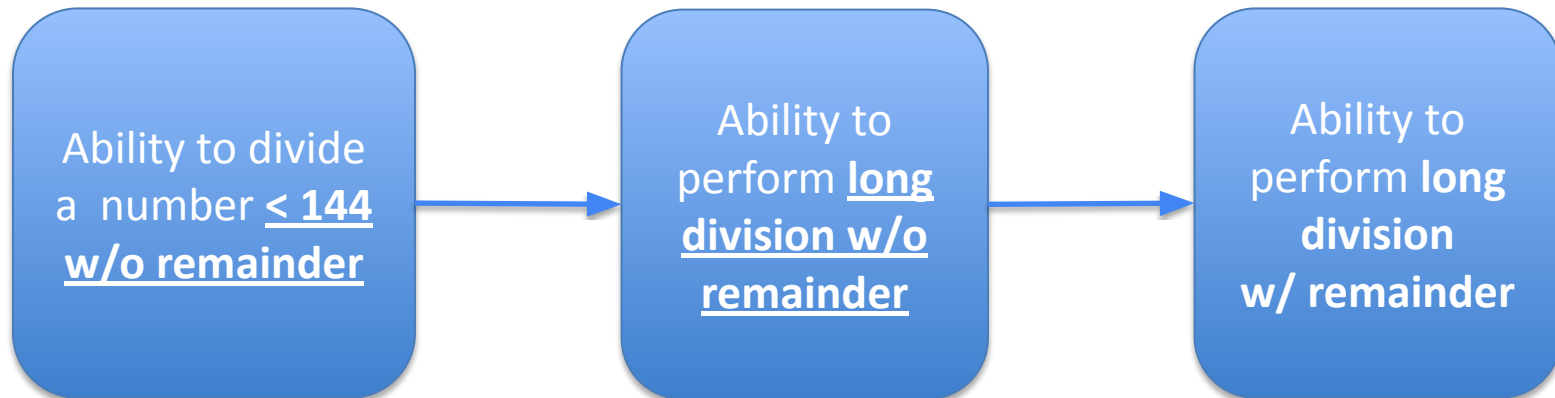
Too coarse-grained – “division” is not one learning goal

Too much content – won't teach this all in one activity

Potentially too easy – division (from products < 13) is memorization, not understanding

Stepwise Refinement – Break it Down!

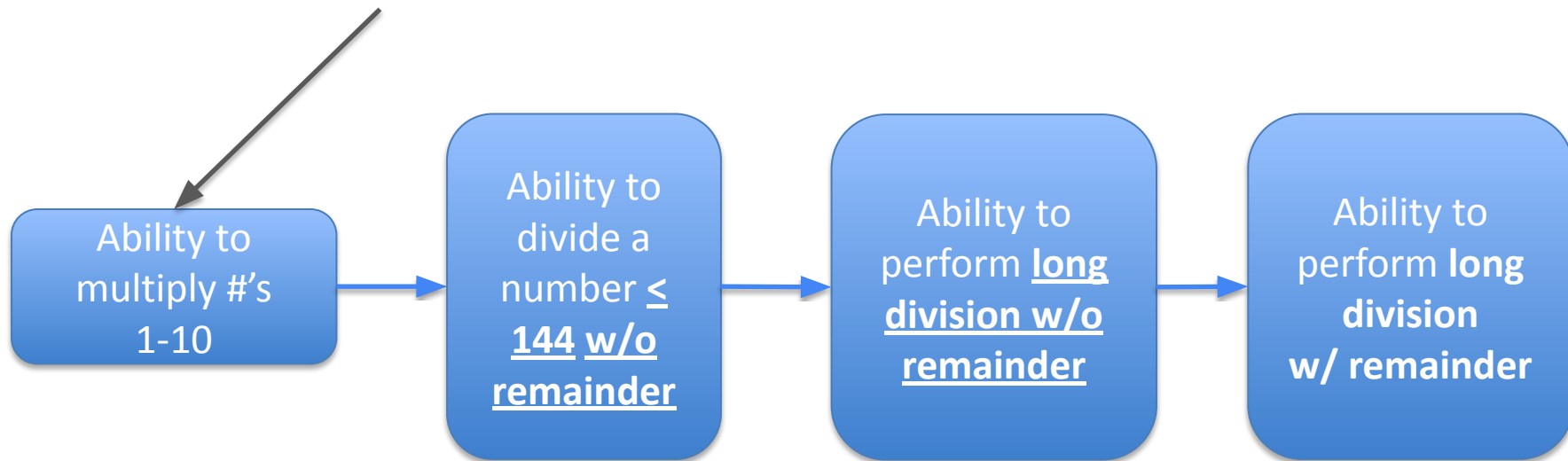
Division



Stepwise Refinement – Break it Down!

Multiplication

Lower Anchor Point(s)
(what learners already know)



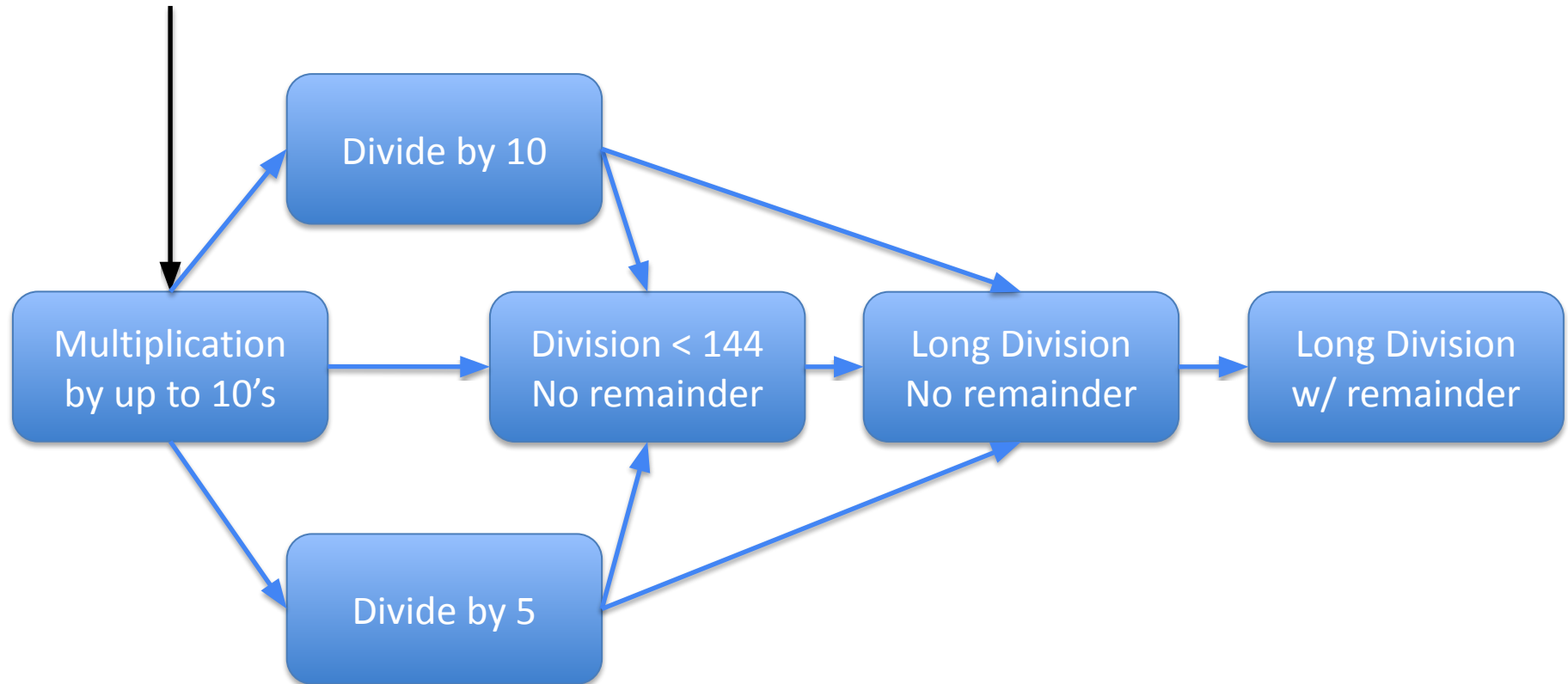
What are some other lower anchor points?

(similar to ability to multiply #'s 1-10 or
in between that and division a # < 144 w/o remainder?)

Stepwise Refinement – Break it Down!

Multiplication

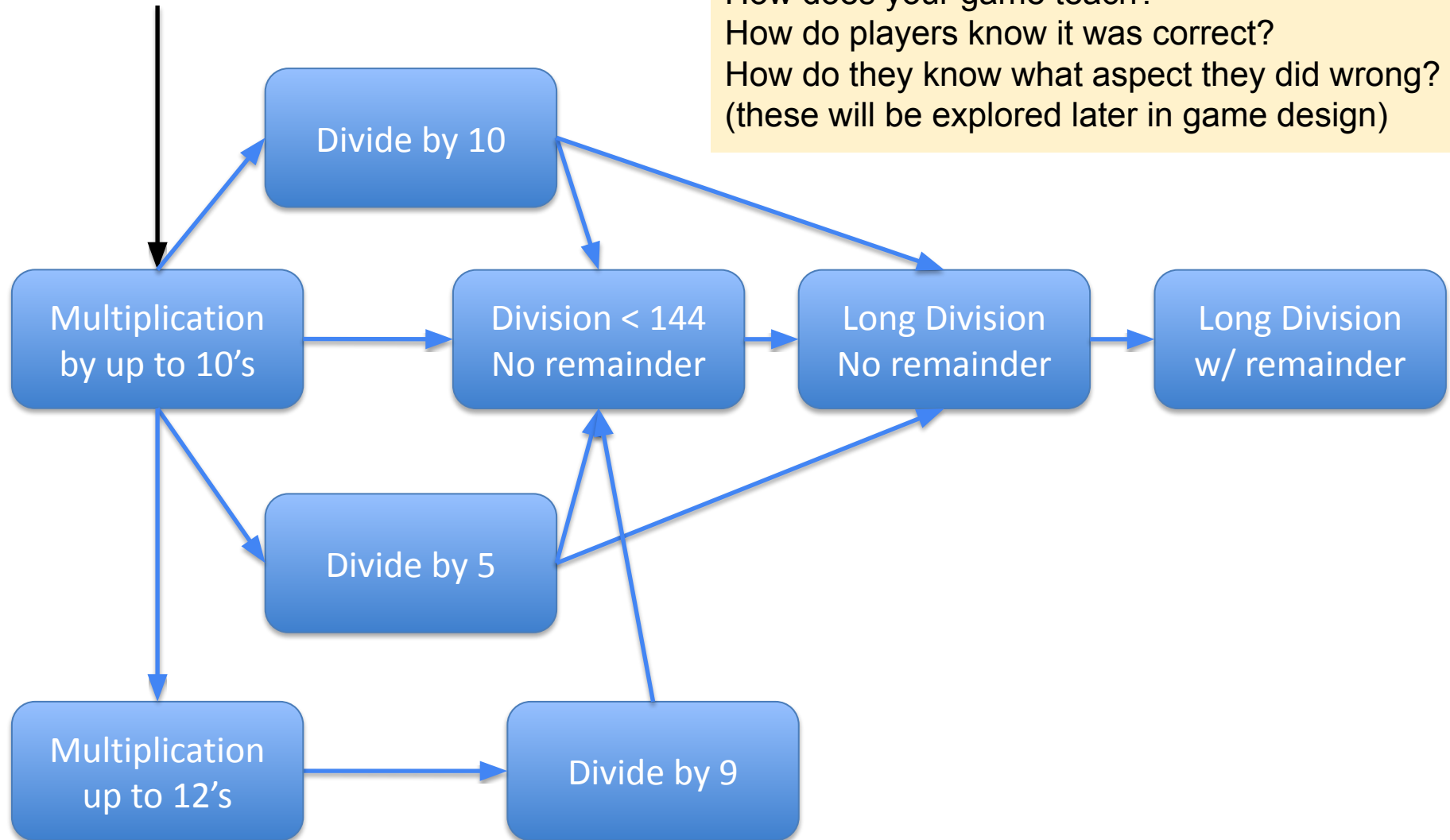
Lower Anchor Point



Stepwise Refinement – Break it Down!

Multiplication

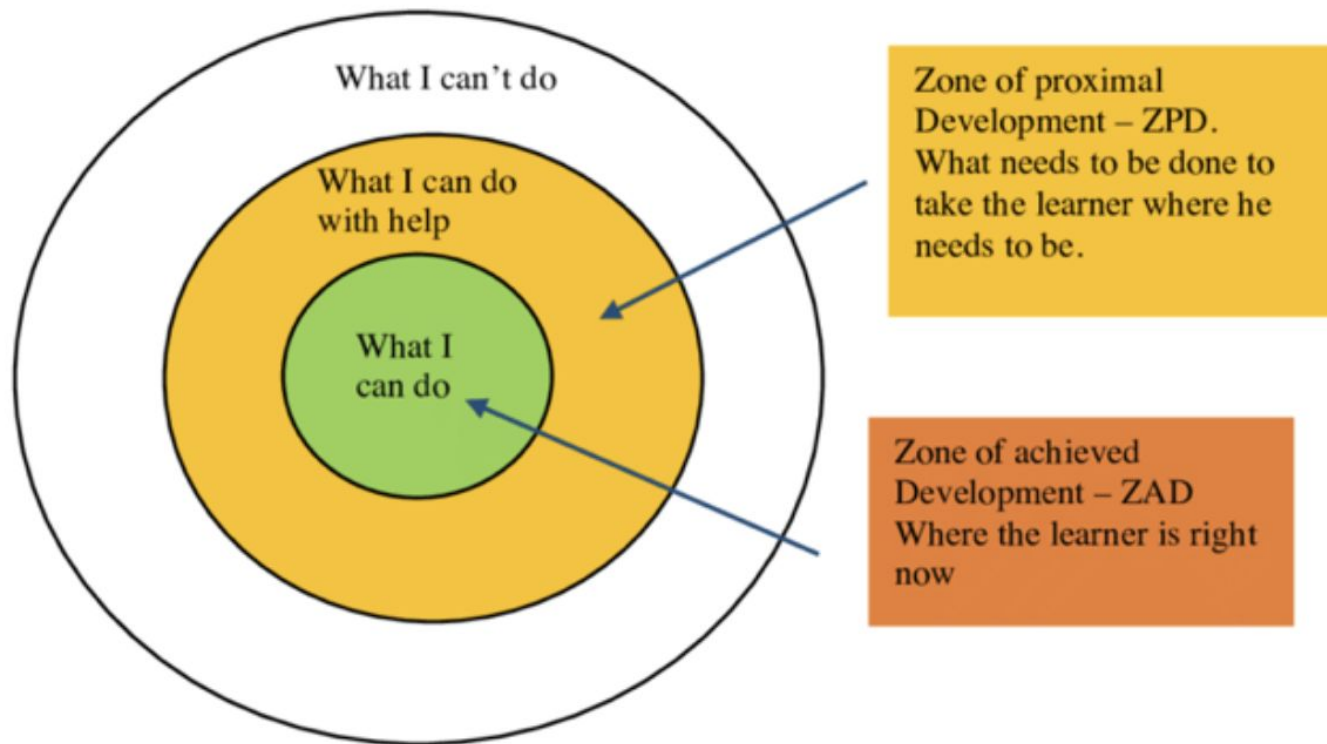
Lower Anchor Point



Practice creating a LT

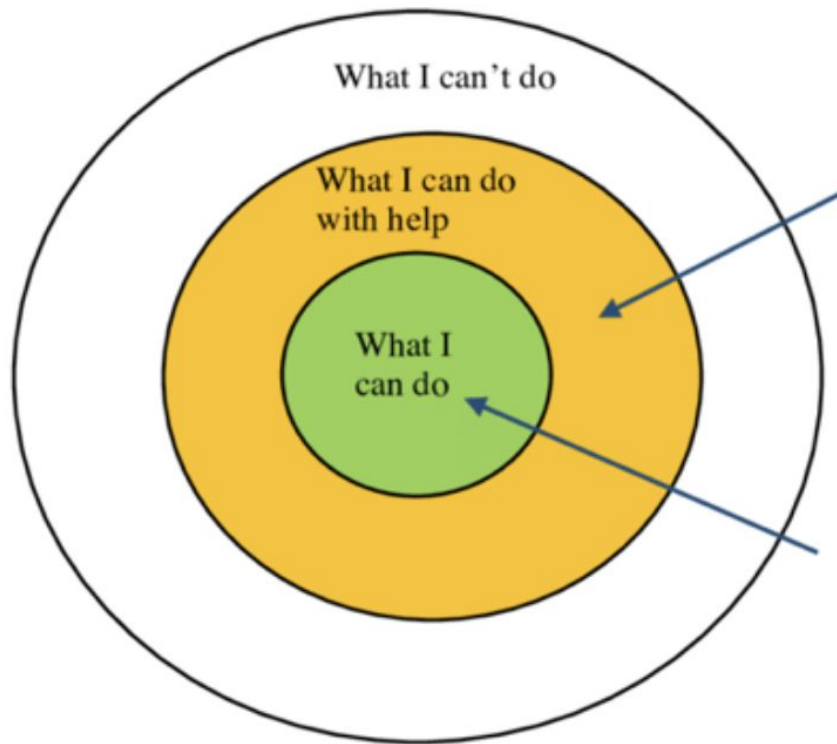
- Try to get together with someone you think *might* be your project partner (groups of 2-3)
 - Those who find their partners move towards back
 - Those still without a partner, mill around towards front
 - Repeat until everyone has someone they *might* work with – no partnerships formed now are set in stone
- Pick a topic that *could* become your real topic but doesn't have to
- Go through the process of defining LGs and creating a LT for your chosen topic (using paper and pen; doesn't need to look polished)

Zone of Proximal Development



zone of proximal development diagram

Zone of Proximal Development



Moving target:

- Not so hard that the learner becomes frustrated or cannot learn from it
- Not so easy that it is covering only skills they have already mastered

Zone of Proximal Development

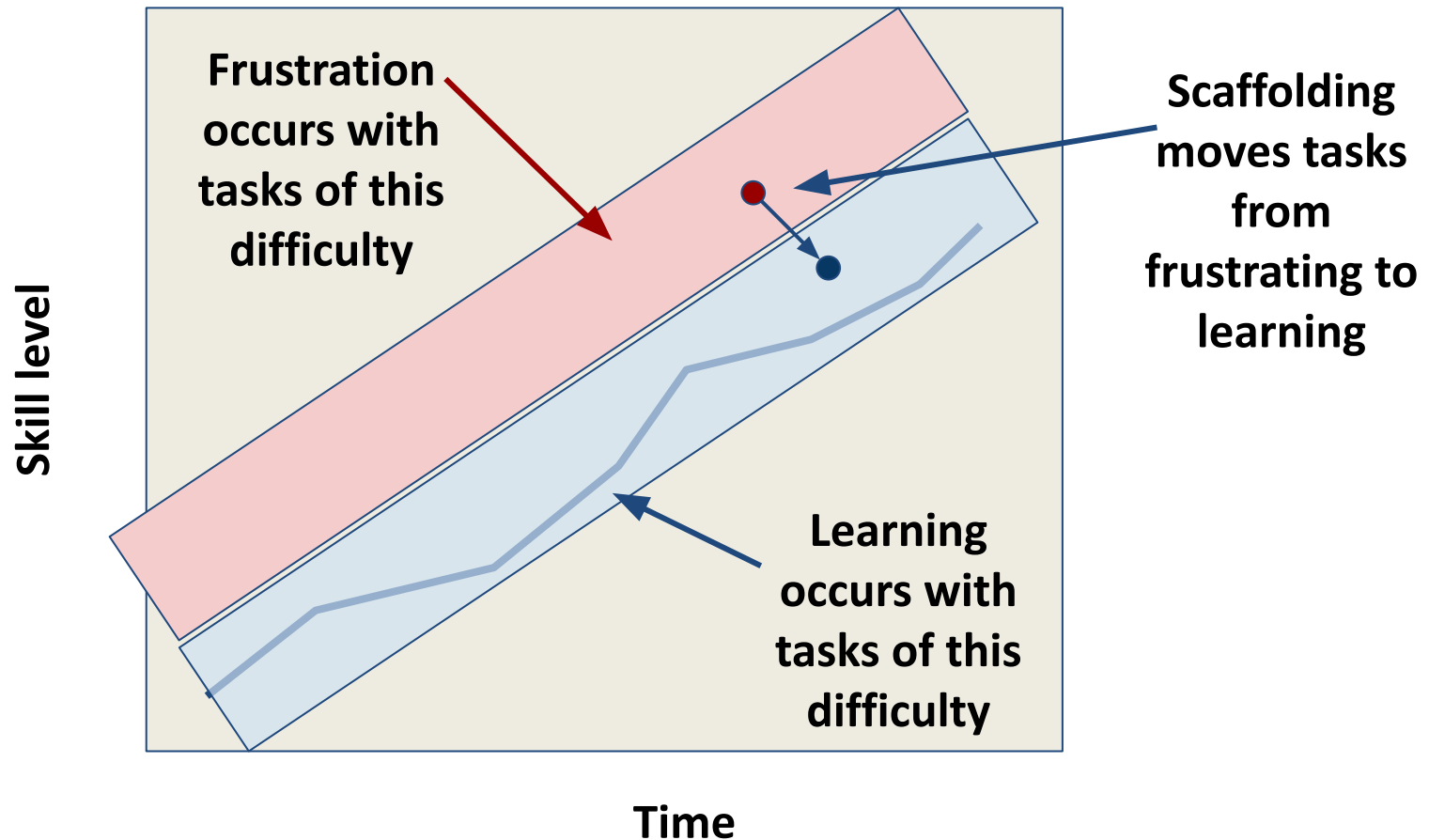
- The Goldilocks view of problem challenge
 - Too hard – learner gets frustrated, learns little because material is out of reach
 - Too easy – learner gets bored and isn't being given material that will teach
 - Just right – learner learns
 - This is impossible if you have a wide range of learners!
You (the designer of the learning activity) will need to decide *who* you want to target the most
- Challenge:
 - As student learns, that zone changes!

Scaffolding

- Temporary
- Removable
- Helps with the activity
- Specific learning goal



Zone of Proximal Development / Scaffolding Relationship



Scratch Activity



Decomp by Sequence: Spirit's Eve Balloon Launch

by [ScratchEncore](#)





Instructions

Complete the code so that the air balloon and the balloon float

Purpose: Modify an animation t
of events with triggers and acti

Notes and Credits

Scratch Encore via UChicago S
of Maryland - College Park

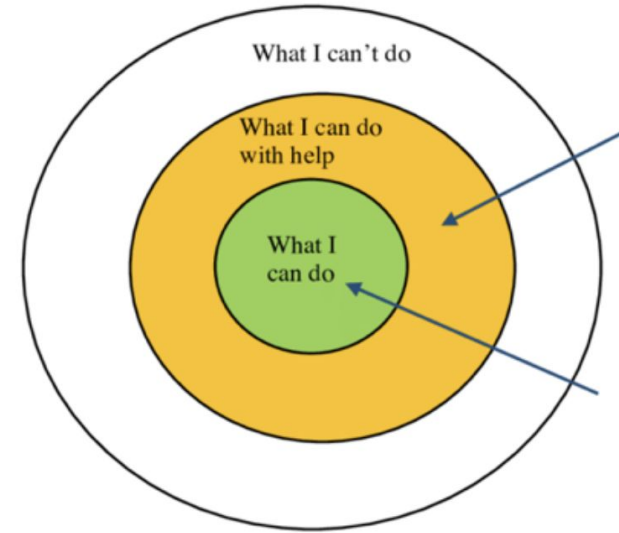
Stardew Valley artwork used wi
ConcernedApe. Use does not i
ConcernedApe.

 2  1  22  85 © Oc

<https://bit.ly/ScratchEncoreDecomp>

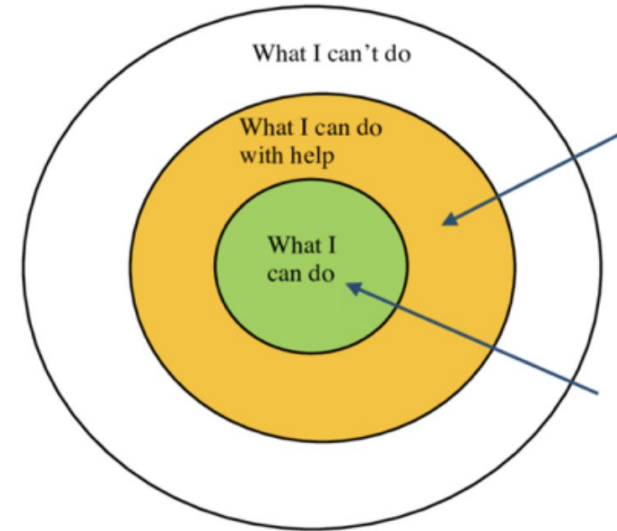
Scratch Activity Reflection

- What was the scaffolding in this activity?
- What did the scaffolding do or in what ways did it help you learn?



Scratch Activity Reflection

This CS topic being taught in this activity is decomposition by sequence, which means “thinking about a project as a sequence of actions and what starts and stops each action”. This is a very important problem decomposition technique that programmers use to break down problems that involve multiple entities interacting together.



Do you think completing this Scratch activity is sufficient for you to go ahead and write a Python project using decomposition techniques?

If not, what knowledge/skill is missing?