Active Learning: more fun, better learning

Diane Horton
University of Toronto

New Educators Wednesday Roundtable
SIGCSE 2020

Active Learning

Anything that gets students to really think.

Many benefits. My top 3:

- You find out what they know and don't know and can adapt your teaching
- *They* find out what they know and don't know when you are there to help
- Better attention and understanding when you are lecturing

It’s doable!

You don’t have to redesign your whole course, make videos, etc.
Flipped/inverted teaching is just one approach

You can start small
If you like it, you can add more

Don’t need TA support in the classroom (at least up to 150 students)

It’s works with classes of any size

Can be done well even in ill-suited classrooms


© University of Toronto
Academic + Campus Events


QD
2018

And it works!

Freeman et al, PNAS 2014

- Meta-study of 224 studies across STEM & engineering
- Only includes randomized controlled studies

“These results indicate that average examination scores improved by about 6% in active learning sections and that students in classes with traditional lecturing were 1.5 times more likely to fail than were students in classes with active learning.”

A little background

Consider this schema:

$$\text{One}(\text{this}, \underline{\text{that}}, \text{other})$$

$$\text{Two}(\text{tweedledee}, \underline{\text{tweedledum}})$$

$$\text{One}[\text{this}] \subseteq \text{Two}[\text{tweedledee}]$$

One

this	that	other
1	88	10
8	17	20
5	52	20
4	89	30
etc.		

Two

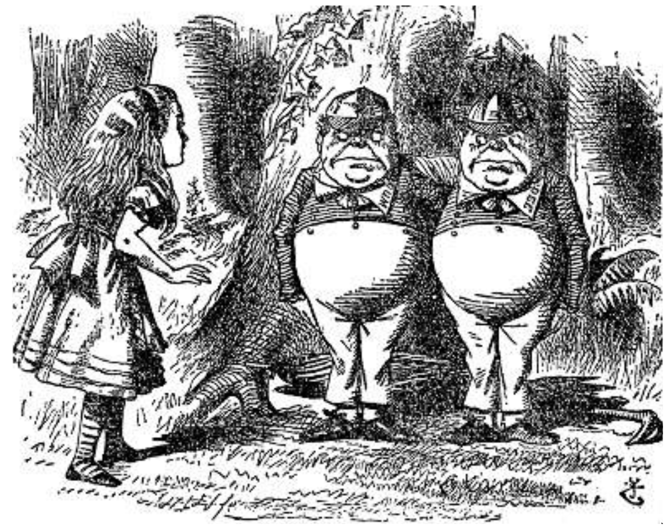
tweedledee	tweedledum
5	A
4	B
1	C
8	D
etc.	

Consider this schema:

One(this, <u>that</u>, other)

Two(tweedledee, <u>tweedledum</u>)

One[this] $\subseteq$ Two[tweedledee]



Suppose relation *One* has 20 tuples. What do we know about the number of tuples in *Two*? Circle the one statement below that is the strongest thing we can be certain of.

(a) The number of tuples in relation *Two* must be >= 20.

(b) The number of tuples in relation *Two* must be <= 20.

(c) The number of tuples in relation *Two* must be 20.

(d) The number of tuples in relation *Two* must be >= 1.

(e) The number of tuples in relation *Two* must be >= 0.

Goals

That question is designed to

- Confirm understanding of keys and foreign keys
- Deepen understanding through explaining to each other
- Get students back into the material at the start of a lecture
- Cause a bit of a stir

Think-pair-share

Format

- Students think about a question individually
- They discuss in pairs or small groups
- One or more groups share their answers with the class

Benefits

- Students more comfortable discussing after they’ve had time to think first
- Discussion quality improved
- Students feel safer sharing with the class because it is a group answer

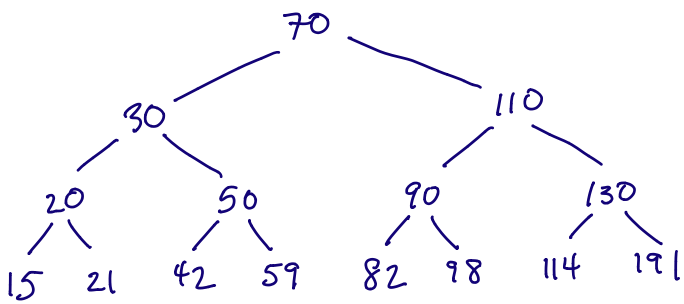
Think-pair-share

Good for

- Checking understanding of prerequisite material from previous class or a reading (a warm-up to the next material)
- Checking understanding at any point in class
- Identifying and clearing up misconceptions
- Building community

CSC148 - Deletion from a Binary Search Tree

1. Suppose you have to delete a value from the BST below. What would be an extremely easy value to delete?



2. Suppose instead you have to delete 70. Ug. One strategy is to leave most of the tree structure as is, but to find another value in the tree that can go where the 70 is. (We can delete that value later.)

   Could 110 go there? ______ Could 20 go there? ______ Could 98 go there? ______

   Exactly which values can go there?

3. Write down the inorder traversal for the above (unchanged) tree.

Next, we will develop method `delete`. Its API is below:


```
class BinarySearchTree:
    """Binary Search Tree class."""
```

Goals

That question is designed for students to

- Realize that deleting from a leaf is easy, elsewhere is harder
- Figure out that you can pick
  - the greatest value from the left subtree or
  - the least value from the right subtree
- See the relationship to an in-order traversal
- Be ready to implement the delete method

15 20 21 30 42 50 59 ~~70~~ 82 90 98 110 114 130 191

Worksheet

Format

- Students work [alone or in groups] to solve a problem or series of problems

Benefits

- You can provide necessary reference material (definitions, an API, etc.)
- Blank spaces can induce writing

Worksheet

Good for

- Applying a definition to confirm understanding
- Practicing problem-solving
- Discovering a technique
- Anticipating a concept
- Inferring an implication
- Realizing a connection between concepts
- etc.

Example: Relational algebra worksheet

A series of increasingly challenging queries to write.
Students encounter seemingly impossible queries, for example:

Given only σ, π, and X, find the maximum grade in a table.

Repeat until solved:

- Students try, and get stuck.
- We regroup to discuss what happened.
- We make a discovery.

Ultimately, they discover the technique for a category of problem they've never seen!

Student number	Grade
51823	75
87128	69
14010	84
66432	72

Goals

That process is designed so that students will

- Deeply understand the relational model and “think relationally”
- Discover the needed technique themselves
- Remember the technique
- Develop general strategies for query writing

- In addition to mastering the fundamental relational algebra operations.

Live Action Demos

Format

- Varied! Specific to the content.

Benefits

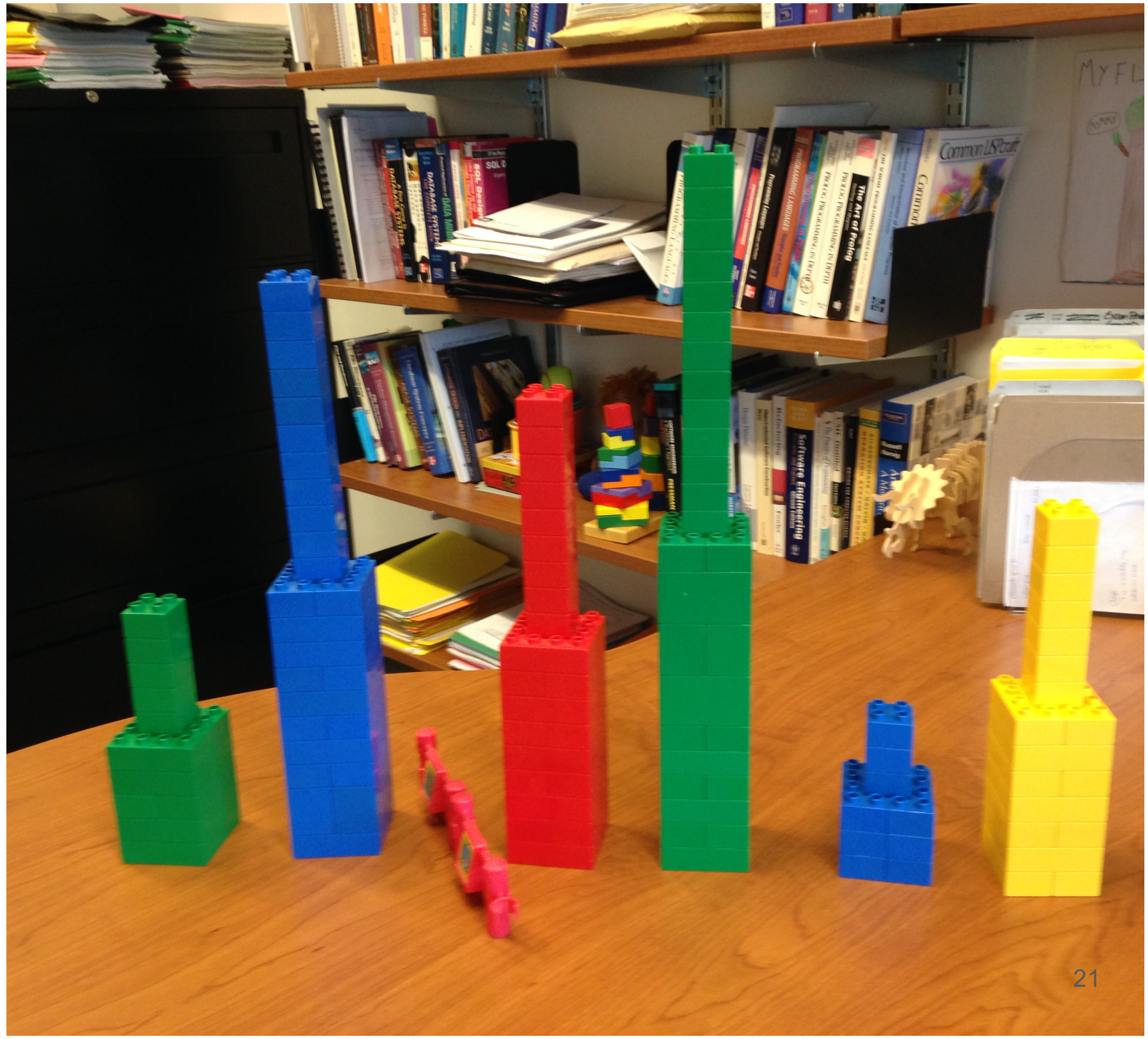
- Memorable.
- Change the energy in the room by getting people up and moving.

Sometimes great, sometimes not.

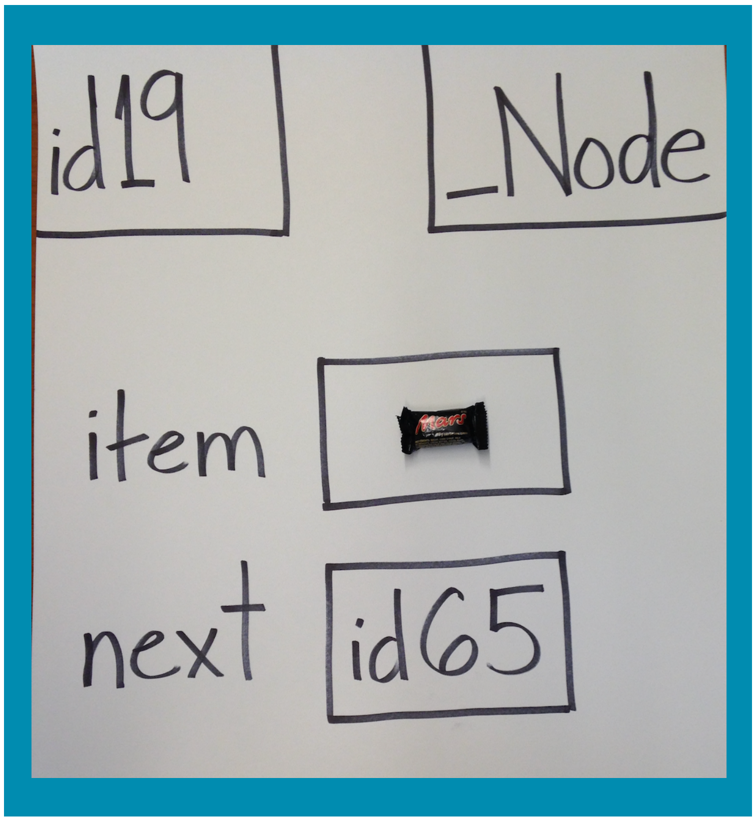
Example: Sorting humans

Playful!
But the participants don’t get to see the big picture.
Takes quite a bit lot of time.

We choose instead to sort objects.



Example: Linked list algorithms

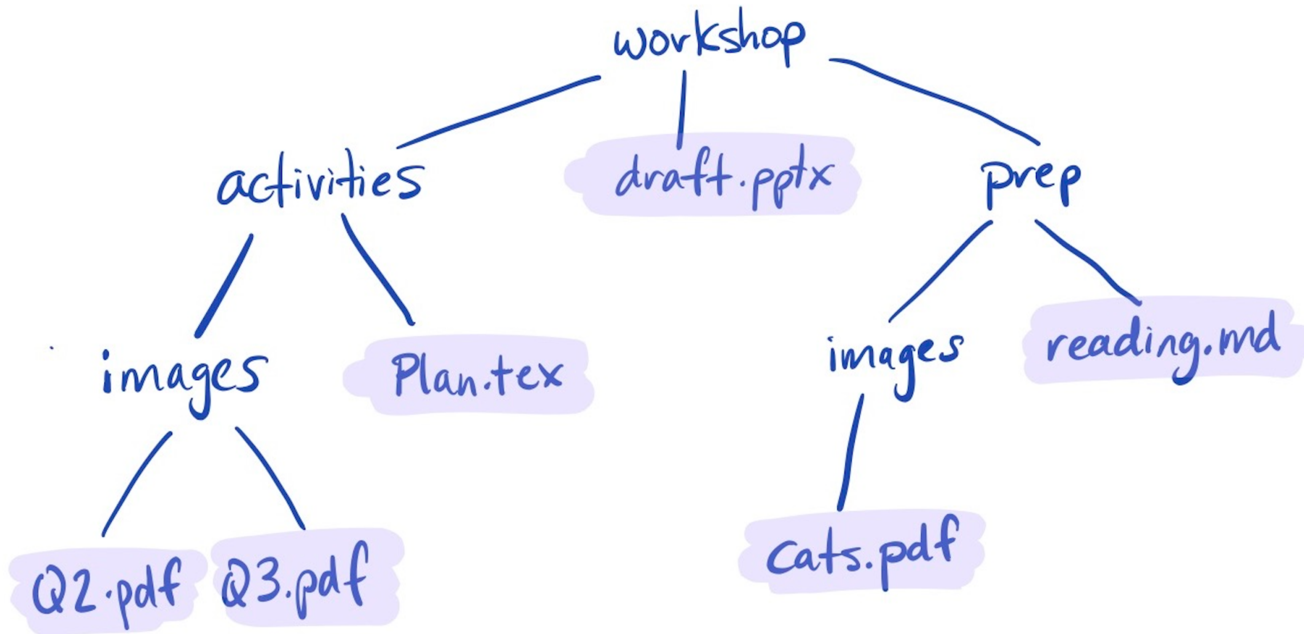


Individual students represent nodes.
Each holds up a drawing of a Node object.

Algorithms are acted out.

Little pay-off for the time consumed.

Example: Acting out tree algorithms



Wolfman, *Making Lemonade: Exploring the Bright Side of Large Lecture Classes, SIGCSE 2002.*

Instructions to students

When asked to join the tree:

- Stand up. You are the root of your own tree.
- If any people are still seated, pick one and ask them to join the tree as your "**right child**". Point your right hand towards them.
- If any people are still seated, pick one and ask them to join the tree as your "**left child**". Point your left hand towards them.

A question for the root: How tall is this tree?

That is, how many people are on the longest path through the tree?

Why this live demo works well

It helps students understand the algorithm

There is no easy way to eyeball it – must apply the algorithm.

Everyone has to apply the algorithm.

It help students get “recursive” thinking: “How can this child help?”

Works well in a large class.

You can have fun with the puzzling moments

It also gets *everyone* standing.

Designing an active learning activity

Identify one thing in your course that students find tricky.

Think about

- What exactly do you want them to be able to do?
- Why do they find it hard?
- Can you lead them to discover it themselves?
- Is there a hard kernel at the core that you should focus on?
- Would breaking it into sub-steps help?
- Is there a more basic concept that is holding them back?

Design an activity that they can do that will be revealing to them and you.

Wrapping up

Making it manageable

You don’t need to flip or semi-flip a course
You don’t need to make a bunch of videos
If you do make videos, high production values may not matter
You can use someone else’s materials
You can get at deep understanding with a quiz on your LME
You don’t need a special room
You probably don’t need TAs in the room to help
Activities can be very low-tech

You can start small and expand over time

Meta study: Freeman et al, *Active learning increases student performance in science, engineering, and mathematics*. PNAS 111 (23), 2014.

UofT study: Horton et al, *Comparing Outcomes in Inverted and Traditional CS1,* ITiCSE 2014.

Student perceptions: Deslauriers et al, *Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom*, PNAS 116 (39), 2019.

Student identities: Henning et al, *Hidden Identities Shape Student Perceptions of Active Learning Environments*, Frontiers in Education, November 2019.

Lemonade paper: Wolfman, *Making Lemonade: Exploring the Bright Side of Large Lecture Classes, SIGCSE 2002.*

Scenarios cards: *Microaggressions: The Game!* and *Teaching Practices Game*. csteachingtips.org/cards

Making it happen

Rewards & what it will take to attain them

- What might you gain by using active learning in your teaching?
- What resources will you need?
- Who will you have to convince?

Risks & how to manage them

- What are the challenges in implementing active learning in your environment?
- What might go wrong?
- What can you do to mitigate those risks?