Becoming a better problem solver

Olac Fuentes
Computer Science Department
University of Texas at El Paso

A tale of two mindsets

Fixed Mindset

Abilities are fixed, effort has little impact

Growth Mindset

Abilities can be improved through personal effort

Neuroscience tells us that mental abilities aren't fixed; the brain is continuously rewiring itself in response to stimuli – thus the growth mindset is not just better for your success, it actually reflects our current understanding of how the brain works.

Neuroscience tells us that mental abilities aren't fixed; the brain is continuously rewiring itself in response to stimuli – thus the growth mindset is not just better for your success, it actually reflects our current understanding of how the brain works.

Examples:

- Taxi drivers in London have a significantly larger hippocampus than average people – the area of the brain that deals with spatial reasoning
- IQ scores have been rising over time – the Flynn effect
- Think of the first time you were asked to write a for loop

Growth Mindset

Not only can we get better, we can get better at getting better

Growth Mindset

What it is not:

Anybody can achieve anything if they try really hard!

- In reality, we all have limits, but they are unknown and unknowable, so there's no reason to stop trying to improve

Growth Mindset

What it is not:

Effort is all that matters

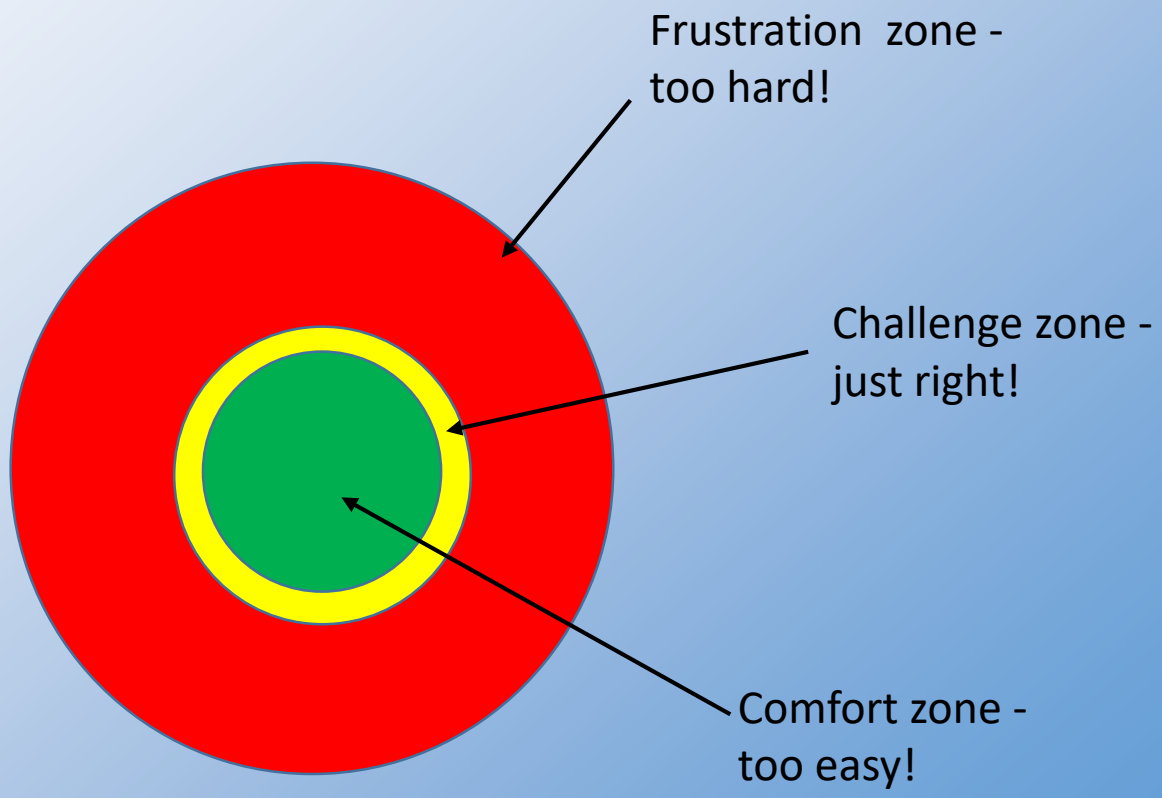
- In reality, effort is crucial. For any of us, the greater the effort, the better the results. But results are what matter.

Back to Computer Science

- Computer science is about problem solving through programming.
- Programming is a skill, like playing video games, writing science fiction, or playing a musical instrument.
- The main goal of a computer science education is to develop your problem-solving skills.
- The proven best way to improve your skills is through **deliberate practice**.

What is deliberate practice?

Purposeful and systematic practice at just the right level of difficulty



Elements of deliberate practice

Elements of deliberate practice

1. **Motivation**
   - You must be willing to work hard and challenge yourself
   - Commitment in terms of time and effort
   - Unfortunately, deliberate practice is not always fun

Elements of deliberate practice

1. **Motivation**
   - You must be willing to work hard and challenge yourself
   - Commitment in terms of time and effort
   - Unfortunately, deliberate practice is not always fun

   Example: tracing a long program, line by line, until you understand exactly what each instruction in the code does.

   It requires motivation, it's not fun, but you can't improve if you don't master this skill.

Elements of deliberate practice

1. **Motivation**
2. **Design**
   - Each practice session must have a clear, concrete, measurable and challenging but realistic objective.

Elements of deliberate practice

1. **Motivation**
2. **Design**
   - Each practice session must have a clear, concrete, measurable, and challenging but realistic objective.

   Bad objectives:
   - "get better at recursion"
   - "understand binary search trees"

Elements of deliberate practice

1. **Motivation**
2. **Design**
   - Each practice session must have a clear, concrete, measurable and challenging but realistic objective.

   Bad objectives:
   - "get better at recursion"
   - "understand binary search trees"

   Better objectives:
   - "be able to solve any of the example problems on disjoint set forests in at most four minutes without looking at my notes"
   - "be able to write from memory all definitions regarding graph representations".

Elements of deliberate practice

1. **Motivation**
2. **Design**
3. **Immediate feedback.**
   - Without feedback it is impossible to evaluate your performance
   - Must be immediate to allow you to determine the next steps during your practice session
   - This is easier in programming than in most other areas (a program compiles, runs, and produces correct results)
   - Take advantage of resources offered by the University: instructor, TA, instructional assistant and peer-leaders

Elements of deliberate practice

1. **Motivation**
2. **Design**
3. **Immediate feedback**
4. **Repetition**
   - You must repeatedly practice the same or similar tasks, progressively moving towards more difficult objectives
   - Frequent, short sessions are more effective than long less frequent ones

Elements of deliberate practice

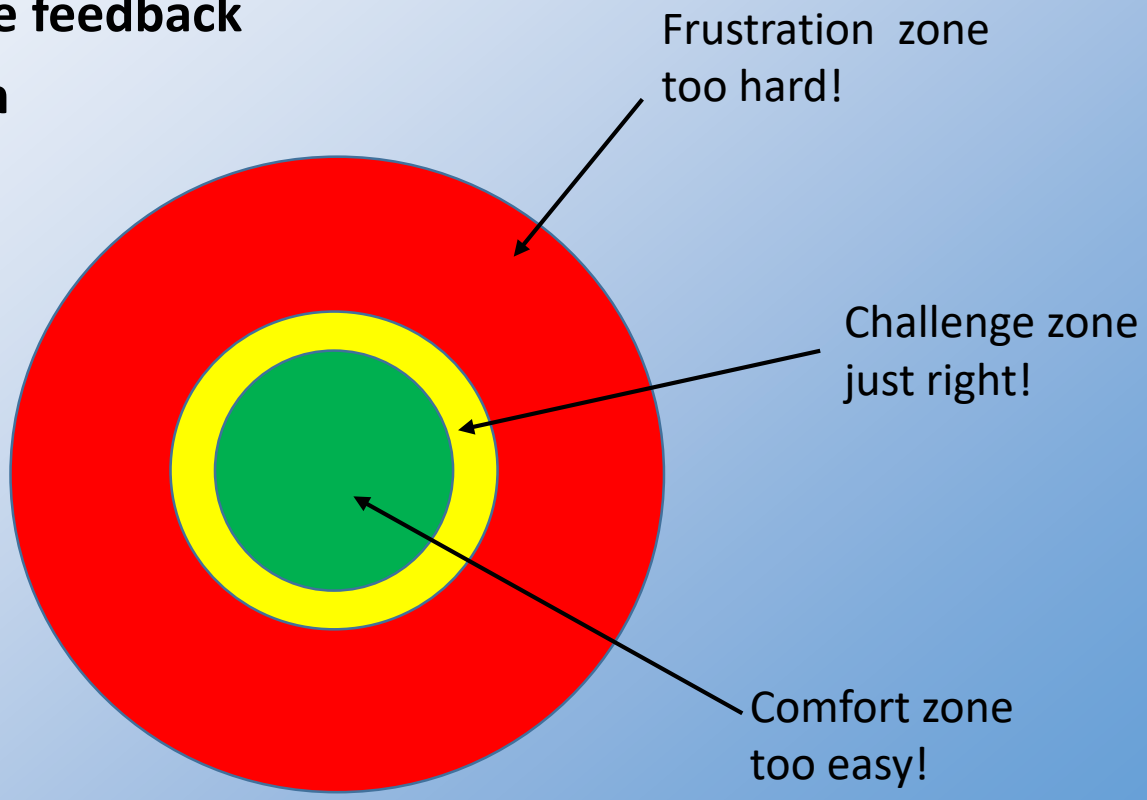
1. **Motivation**
2. **Design**
3. **Immediate feedback**
4. **Repetition**

In this class you will be doing A LOT of deliberate practice!

- In-class exercises and quizzes
- Common model of leaving all learning to the day(s) before exams does not work!

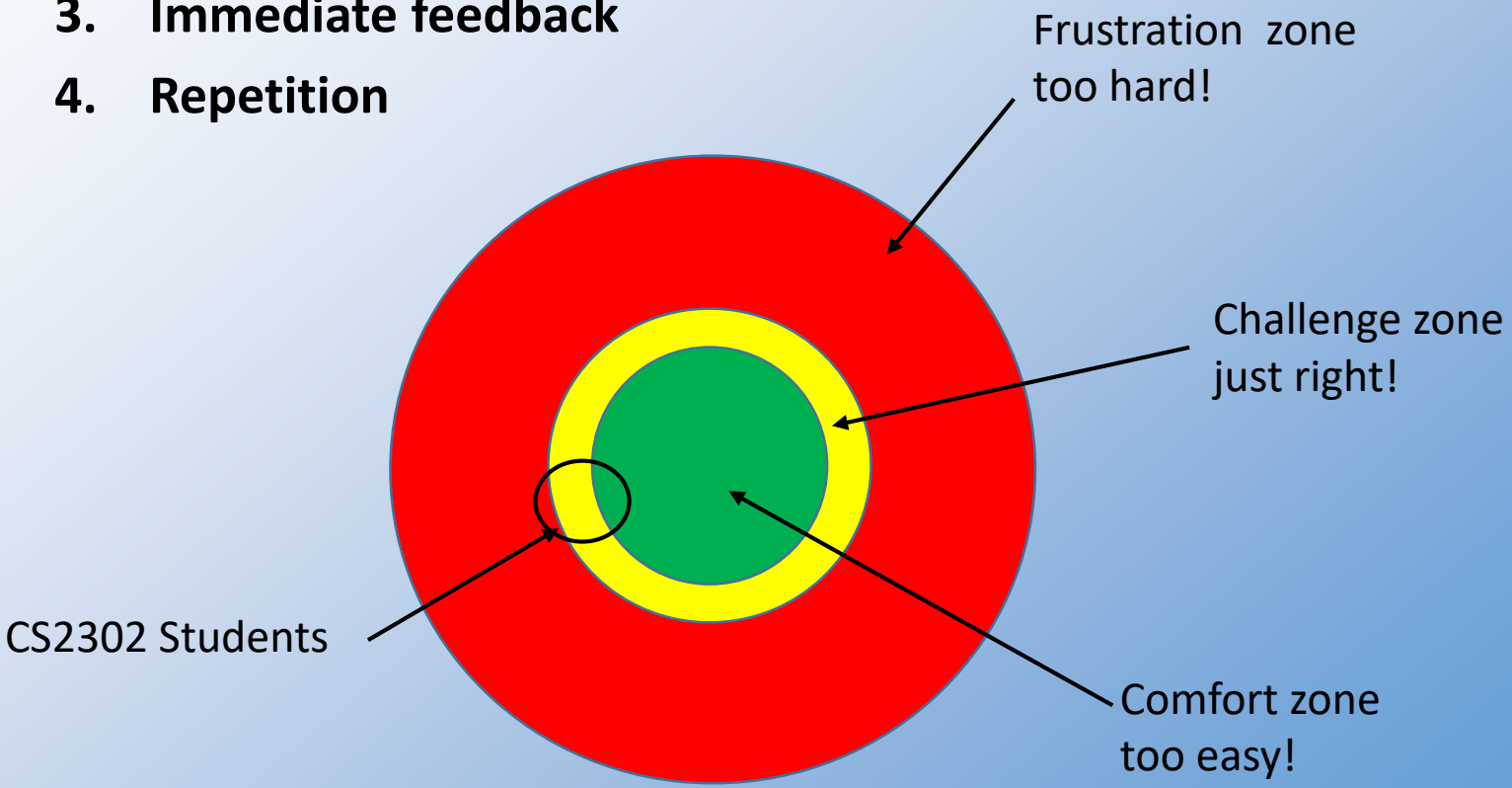
Elements of deliberate practice

1. **Motivation**
2. **Design**
3. **Immediate feedback**
4. **Repetition**



Elements of deliberate practice

1. **Motivation**
2. **Design**
3. **Immediate feedback**
4. **Repetition**



This class is designed so exams, quizzes and labs should fall into the challenge zone for students who successfully completed CS2401 and MATH2300

If you think most questions fall into your comfort zone:

- Try to solve the extra credit exercises (when provided)
- Invent you own extensions to the exercises
- Ask for more work!

If you think most questions fall into the challenge zone:

- You're in the right place
- Enjoy the challenge and learn!

This class is designed so exams, quizzes and labs should fall into the challenge zone for students who successfully completed CS2401 and MATH2300

If you think most questions fall into the frustration zone:

- Realize that with hard work this situation is temporary
- Resist the temptation to just look up the answers or ask a more advanced friend for help
- Do not waste your time looking for “better” explanations online
- **Almost always the problem has to do with background material**
- Practice with background material until it stops being challenging and becomes easy
- Then the current material should go from frustrating to challenging

Usually, students who pass CS2401 and MATH2300 with A or B start the semester in the challenge zone

Some students who pass CS2401 with a C and students who took a break from school often start the semester in the frustration zone – if that is your case, talk to us, we can help!

Regardless of where you start, with **deliberate practice**, by the end of the semester most problems, questions and exercises will be in your comfort zone (a few will still be challenging, but that's OK!)

This class should be challenging, not frustrating

Our goals:

- Convince you of the power of a growth mindset
- Help you develop your problem-solving skills through deliberate practice
- Make you feel challenged, not bored or frustrated
- 100% passing rate

... but we need your help!

Obstacles to successful deliberate practice:

- Searching for shortcuts
  - Not reading code, hoping text explanation will be enough to understand
  - Skipping mathematical formulas describing a problem or algorithm, hoping text will be enough to understand
  - Skipping pseudocode description of an algorithm, hoping example traces will make it clear
  - Searching for online resources (usually videos) to understand a topic presented in the textbook or course slides
- Focusing on the wrong thing
  - Read lots of problem solutions, without practicing how to solve problems
- Wrong self-assessment
  - Underestimating or overestimating your skill level, which leads to practicing things that are too easy or too hard
- Not searching for feedback
  - Your teaching staff is here to help you, but they need to see your work to figure out what you need

Quiz given to students before a Data Structures exam:

Answer True or False

1. I know all the relevant concepts and definitions regarding the subject.
2. Regarding programming, I can trace a method that solves a problem in the area; I understand exactly what each line in the code does.
3. When presented with a solution to a particular problem, I completely understand it and could replicate it without looking at my notes.
4. When presented with a new problem, I can solve it.

Answers:

1. True 74%, False 26%
2. True 63%, False 37%
3. True 54%, False 46%
4. True 82%, False 18%

Do you see anything wrong with these answers?

- Students overestimate their skill level
- A student answering *True* to question 4 must have answered *True* to the 3 previous questions

Pair Exercise

Given what we discussed about mindset and skill development, discuss possible answers to hypothetical students who say the following:

1. “I’m having a hard time understanding how to use 2D arrays”
2. “I’m just not good at math”
3. “I’ve read the book 5 times and still can’t solve any of the problems”
4. “I understand the concepts, but don’t do well on exams”
5. “I don’t even know where to begin to solve the problems assigned in class”
6. “I just don’t understand the Python syntax”

Questions?

Comments?