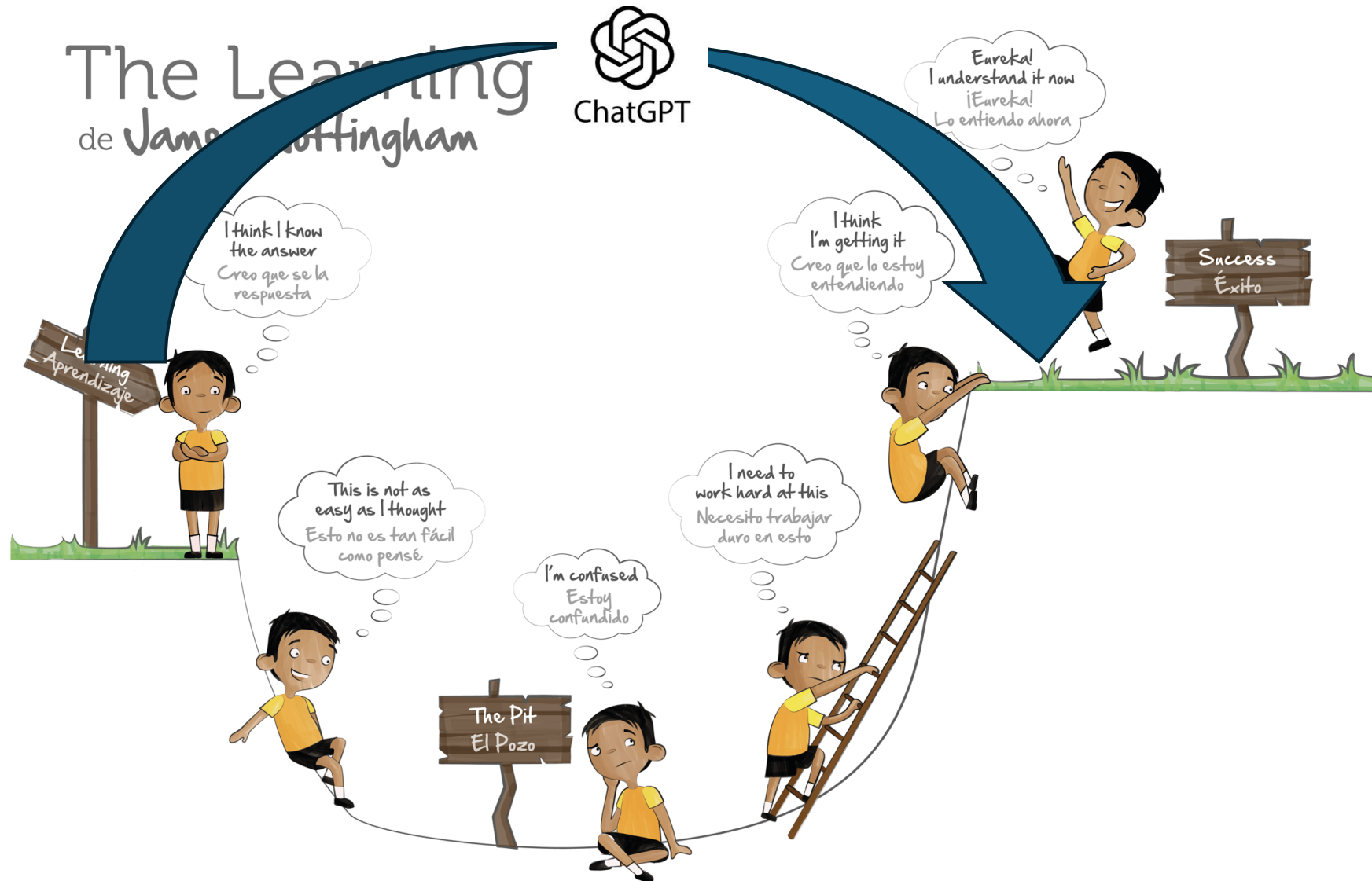
A top-down view of a wooden desk. In the top left is a small potted plant with green grass-like leaves. To its right is a white computer keyboard. Below the keyboard are two black paper clips. In the bottom center is a black pen. To the right of the pen is a black spiral-bound notebook with a white sheet of paper on top. The background is a dark, solid color.

When AI encourages thinking, not answering: Lessons from TRU-Think in classroom

Quan Nguyen, Nisha Puthiyedth
Assistant Professor, Computer Science
Thompson Rivers University



Is learning supposed to feel easy?



Students missing out on “productive struggle”


- Break down the problems into smaller steps
- Attempt
- Debug
- Attempt again

write a function to add two numbers

Here's a simple, clean example in **Python**, plus versions in a few other common languages in case you want them for teaching/demo purposes.

✓ Python

python

 Copy code

```
def add_numbers(a, b):  
    return a + b
```

Usage:

python

 Copy code

```
result = add_numbers(3, 5)  
print(result)    # 8
```



AI that support – not replace – human learning

@tutor write a function to add two numbers



TRU Think

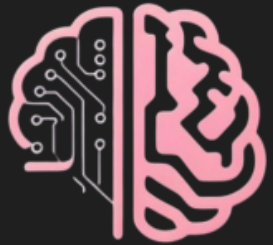
I can't provide a direct solution, but let's break it down together!

Steps to Create the Function

1. **Define the function:** Start with the `def` keyword.
2. **Choose a name:** Think of a descriptive name for your function.
3. **Add parameters:** You need two parameters for the numbers you want to add.
4. **Return the result:** Use the return statement to give back the sum.

Can you try to write out the function definition based on these steps? What do you come up with?





TRU Think

TRUCS

18



An intelligent coding companion for Visual Studio Code, powered by GitHub Copilot

Disable



Uninstall



Auto Update



Beyond answers

TRU-Think is purposefully designed to foster critical thinking and independent problem-solving.


Rather than offering direct solutions, it provides scaffolding through clarifying questions, step-by-step guidance, and analogous worked examples to help students navigate their own learning processes.

TRU-Think reinforces learning through specialized modes like coding practice, exam preparation, and concept revision, challenging students to apply their knowledge in diverse contexts.

Choose a mode to start your TRU Think session:

 **General**


Ideal for general coding questions and tutorials

 **Coding Practice**

Practice coding problems with hints and feedback

 **Concepts Review**

Learn and review core programming concepts

 **Practice Exam**

Simulate exam questions

Pilot in COMP 1110

Introduction to Programming with Python

- Fall 2025
- 60 students
- AI usage is part of the exercise during the seminar hour to help them review concepts, practice coding questions

Exercise X: Learning with GenAI — TRU Think

In this exercise, you will interact with **TRU Think**, an extension in VS Code built on top of GitHub Copilot. The goal is to practice learning through **concept review**, **coding practice**, or **exam simulation**, while engaging critically with the AI tutor.

Follow the steps below:

1. **Open VS Code.**
2. **Open Copilot** by clicking on the Copilot icon in the left sidebar.
3. **Activate TRU Think** by typing `@tutor` in the chat box and pressing **Enter**.
4. **Select a Mode:**
 - Concept Review
 - Coding Practice
 - Practice Exam
5. TRU Think will ask you which topic or concept you want to study. You can choose any of this week's learning outcomes, such as:
 1. [Learning Outcome 1]
 2. [Learning Outcome 2]
6. **Engagement Requirement:** You must exchange at least **20 messages** with TRU Think (20 prompts from you, 20 responses from the extension).
7. When finished, click the **Export Assignment** button in the bottom status bar and select **Export as PDF**.
8. **Submit your exported PDF** together with your assignment for this week on Moodle.

Example of cognitive engagement

i would like to review
repetition structures,
starting with the while loop

Great choice! Repetition structures are fundamental in programming.
Here are some related concepts you might also want to explore:

...

Let's start with the **while loop** as you suggested.

High-level overview:

A while loop repeatedly executes a block of code as long as a given condition is true.

Analogy

Think of it like checking if your phone battery is above 10%. As long as it is, you keep using your phone. Once it drops below 10%, you stop.

Step 1:

Can you describe, in your own words, how a while loop works and when you might use it? Try to write a simple pseudocode example (not actual code yet).

Example of cognitive engagement

a loop that repeats as long as it's true

Step 1:

Can you describe, in your own words, how a while loop works and when you might use it? Try to write a simple pseudocode example (not actual code yet).

Great start! You've captured the essence: a while loop keeps repeating as long as its condition is true.

Feedback:

- You understand that the loop depends on a condition.
- It's important to specify what changes inside the loop so the condition eventually becomes false, or else the loop could run forever.

Next step:

Can you try to write a simple pseudocode example of a while loop? For instance, imagine you want to count from 1 to 5. How would you express that in pseudocode? Give it a try!

Example of cognitive engagement

```
num = 1
```

```
while num < 6:  
    print (num)  
    num= num + 1
```

Can you try to write a simple pseudocode example of a while loop? For instance, imagine you want to count from 1 to 5. How would you express that in pseudocode? Give it a try!

Great job! Your pseudocode correctly demonstrates how a while loop works to count from 1 to 5.

Feedback

...

Next step:

Can you explain what would happen if you forgot to increment `num` inside the loop? Why is that increment important?

Questions about learning strategy

How is Python good for lab future work?

how biometrics worth for comp courses?

which is the best advices to study for a quiz, midterm or a final?

What do you think is good to study the night before a midterm?

the best mode that can International students in Canada can handle the emotion on winter time?

All right you can assist on that but in your opinion TRU is ranked overall all Canada?

right now am with functions and is kind of hard

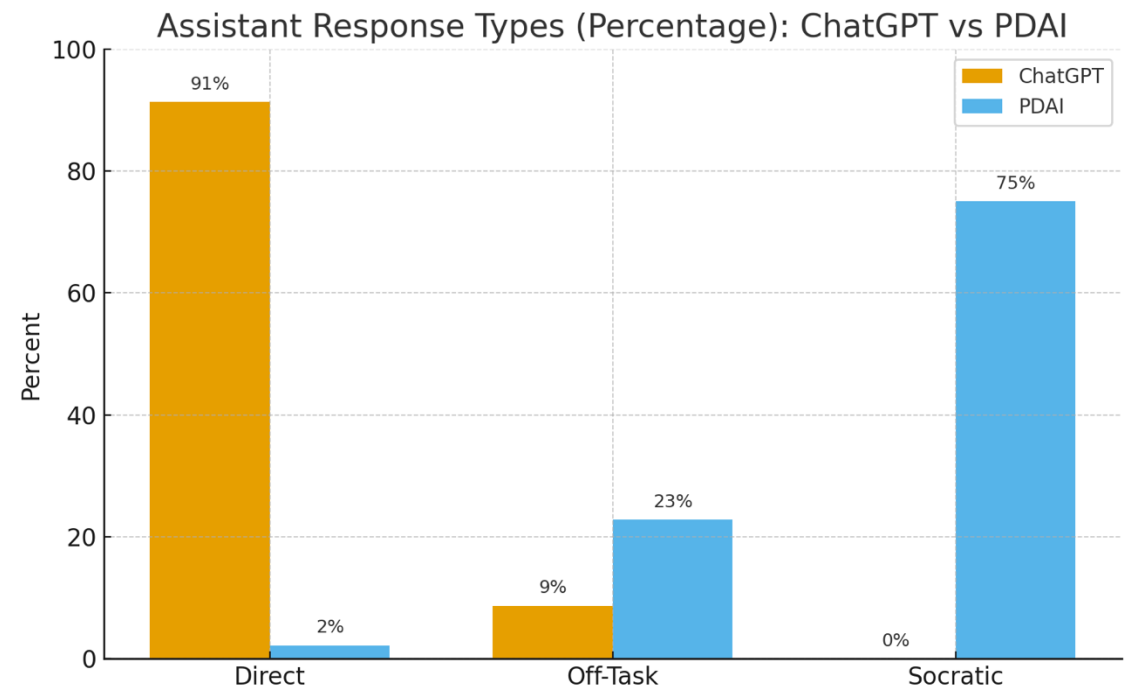
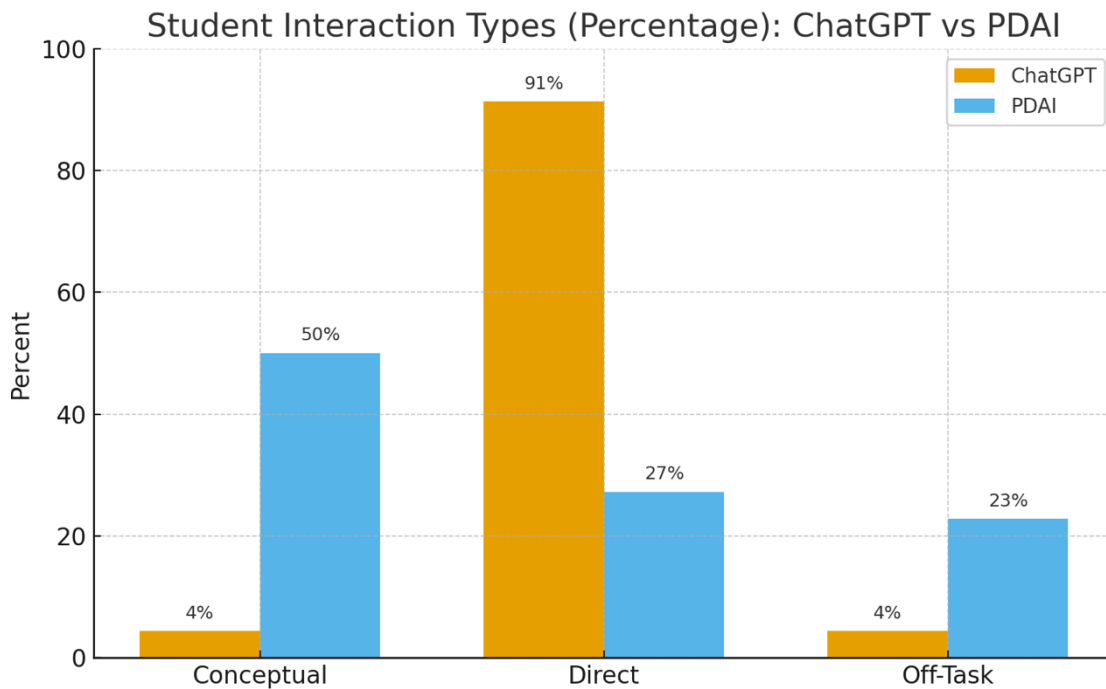
Chemistry is my major I love also so what will be a good connection with python program?

Give me an example of python program with shows balance, name, int and other stuff.

How can you collaborate soccer with science but like computer science?

A pilot experiment

- 10 students use chatGPT
- 10 students use TRU-Think
- Solving the same problem
- 45 mins pre-during-post test



PDAI (TRU-Think) generated 5.4 times more interactions than chatGPT

Lessons learned

- Competing with chatGPT is difficult (they are addictive like sugar!)
- AI needs to be interwoven with assignments, lab activities
- Tools should be accessible and equitable (free, easy-to-use)
- Students are capable of using AI responsibly, but they need our help! Don't just say "don't eat sugar, it's bad for you"



The bigger picture


- Everyone is using AI, but how we know if it's effective?
- We need to collect data on how students engage with AI to make decisions on AI adoption
- Otherwise, we are just walking blind and make assumptions!



Future development

- Open up to other disciplines
- Moodle integration
- Multi-modal (ability to process images, audios, videos, figures, tables, plots)
- Group chat features


@tutor when is the office hour

 TRU Think

Current Mode: Moodle

The office hour for ADSC 3610 is scheduled for Thursdays from 11:30 AM to 12:30 PM at OM 1232. You need to book your timeslot in advance.

@tutor when is assignment 2 due

 TRU Think

Current Mode: Moodle

Assignment 2 for ASTR 1150 is due on **June 4, 2025 by 5 pm**. Make sure to submit it before the deadline to avoid late penalties.