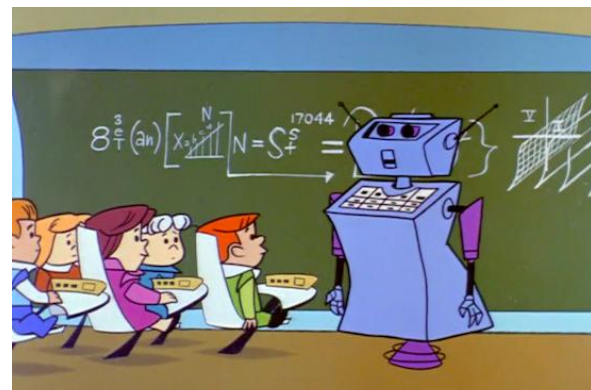


# Designing a Teachable Agent for Novice Programmers

Amit Kumar, Xiaoyi Tian, Zachary Wilkerson (6930)

Project Proposal for CIS 4930/6930

Fall 2021



# Motivation: Novice Programmers

- Novice students in introductory computing class struggle with
  - Development of computational thinking (CT)
  - Language-specific challenges
- Spoken dialogue system might help
- Learning by teaching
- Teachable agents influence self-efficacy, motivation and learning (Jacq et al., 2016)

# Overview: Teachable Agent

- English-based teachable agent: *Nova*
- Students can teach the agent specific programming topics
- Two types of problem:
  - Coding problem
  - Conceptual problem
- *Nova* persona:
  - A young, middle-school-aged female learner
  - Two dialogue styles: neutral and social

# Dialogue Example:

SYSTEM: [Introduction]

SYSTEM: What topic would you like to work on today? **Topics** that I'm interested in are **shown on the screen** (*database displayed through visual interface*). I can also list these topics out loud for you, if you want.

USER: I want to solve a system of 3 equations.

SYSTEM: That problem is a bit too complex for me, as I'm still a beginner! Could we try something easier? For example, you can teach me (*random topic selection from database query*) **how to add 2 numbers**.

*(Quiet for a few seconds as user reads possible topic list)*

USER: Let's try adding two variables.

SYSTEM: Okay! **I'm trying to design a program that stores two numbers, each as a different variable, adds them together, and then outputs their sum. Where should we start?**

USER: First, we should **create a variable** to store one of the values.

SYSTEM: Okay, why do we do that?

USER: Because a variable allows you to store the user input which you can use later on.

SYSTEM: Okay, the first variable is set now. What do we do next?

USER: You can then similarly create another variable to take the second input.

# Dialogue Example:

SYSTEM: Cool! **Why do we need another variable?**

USER: Because you already used the first variable to store the first number and you don't wanna overwrite the value of it.

SYSTEM: Hmmm, interesting. Could you explain a little more about "overwrite"?

USER: Overwrite means the value of a variable is completely erased by another value. You don't wanna erase the first value because you will need to use it later on.

SYSTEM: Great, So I have the two variables now with their inputs. What do we do then?

USER: Well you add them together.

SYSTEM: Cool! How do we add them together?

USER: You can create another variable and store the addition of the other two variables inside it.

SYSTEM: Okay, Great! What should we do next?

USER: **Well that's it! We have solved it now.** If we want we can also output the third variable.

SYSTEM: What is an output?

USER: An output of a program is something you display to the user.

SYSTEM: How do we output a variable?

USER: Make a print statement.

SYSTEM: **Hooray! We did it! Thank you so much for helping me!**

# Project Goals:

Minimal
Create knowledge database (1)
Program conceptual agent via DialogFlow
Test for concept task completion

Basic
Expand knowledge database (2)
Implement visual interface (coding)
Improve system flexibility (more training phrases, responses)
Test for all task completion (+feedback request)

Stretch
Expand knowledge database (3+)
Customized agent
Host system publicly
Implementation on common platforms

**Q&A:**