Ethan Wong

S4OP

I had a bit of trouble understanding the concept of recursion after watching Lecture 12 and decided to research the topic more online. I found a great example that clarified my conclusions:

- Think of recursion in terms of compound interest. With compound interest, the money earned is based off of what has accumulated in the account, not the initial investment.

For example, investing $100 with an interest rate of 10% compounded once a year would entail:

Years passed	0	1	2	3	4
Money in Account	$100	$110	$121	$133.10	$146.41

Seeing how the amount of money gained each year wasn't a flat $10 really put the concept of recursion into perspective for me. The initial condition would be the initial investment of $100 when zero years have passed since the investment. The recurrence relation would be expressed by multiplying the amount of money from the previous year by 1.10 to get the amount for next year.

While I was studying the textbook, I came across an exercise that stated

- Explain how to solve a recurrence relation by iteration.

When I saw this, it made me think of iteration from a computer science standpoint using loops. I know for loops / while loops aren't really used in math, but here's how I would solve a recursive relation with the logic of a while loop

declaration of initial condition (i.e. apples = 5)
while (stopping condition is not met):
    perform some operation on initial condition
    print (result of previous line)

By using this logic, I would have the initial condition and all the terms after it. I could analyze this sequence of terms to find a pattern and solve the recursive relation. I realize this is pretty much what is expected, but I wanted to take a computer-science based approach because that's what I'm more comfortable with.