A screenshot of a computer

AI-generated content may be incorrect.

3. Write a short paragraph about your approach and why is solves the problem.

My approach is to replace the for loop with a while loop, and for that to happen, I would need a count variable to keep track of how many inputs are being entered. At the same time, I would require another variable to keep track of the numbers within the resized array to ensure that previous indexes aren’t overwritten. Resizing by doubling the length of the array is a simple solution. Additionally, to properly list out the numbers within the array, I’d have to iterate through the count variable instead of the current size of the array otherwise it would print out the defaults (0) for the rest of the “empty” indexes.

A computer screen with white text

AI-generated content may be incorrect.

5. Define what an Object is in terms of the Object Oriented Programming paradigm. Feel free to use AI like this prompt I put into CoPilot. What is an object in object oriented programming. How does a Python List embody what CoPilot to you?

Objects are instances of a class, a blueprint of code, that contain defined data and functions. They are used to interact with other elements within projects to manipulate data and produce a desired result. I have no idea how a Python list embodies what CoPilot to me (???).

I am going to assume the question is: How does a python list embody an object and not an array? Python utilizes built-in functions to allocate data within arrays. Therefore, it is not strictly an array, but a bundle of code and function to create lists AND relocate data in memory. It is more than an array because it is not simply an order of data. The list would have to then be resized manually if it was a proper array without build-in functions to do the work.

6. How does using a linked list solve some of the same problems we are trying to overcome when using dynamic memeory?

Linked List allows for a noncontiguous allocation of data in memory. This makes it easier to store data as you don't have to worry about the exact location of the data, but only when and where to point a node to another node. Linked Lists allow for the easy implementation of dynamic sizing, since data can be stored anywhere in memory. It sacrifices organization and speed for simpler implementation.