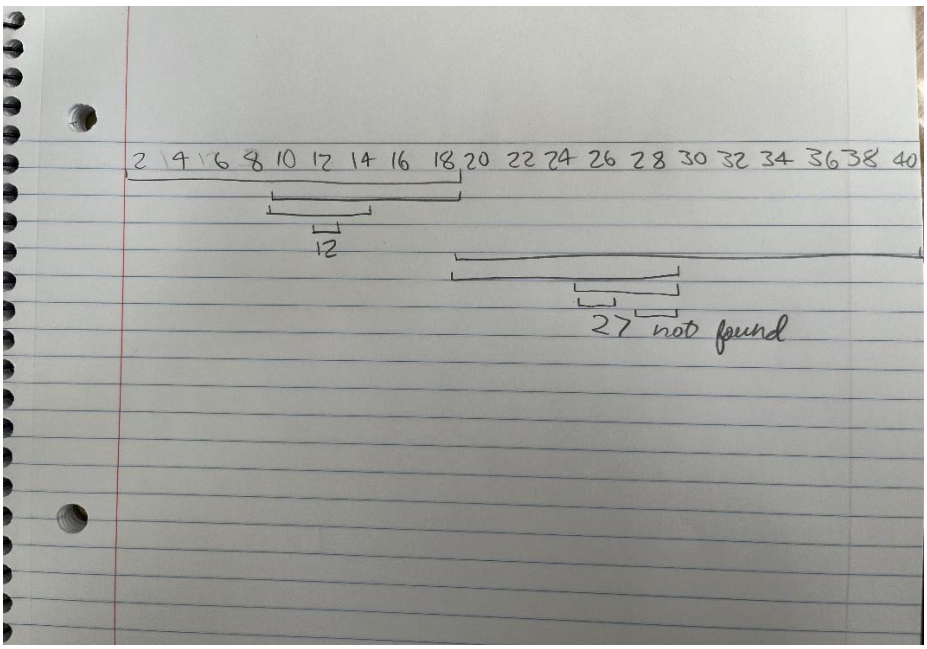

WHILE not found

SET middle

IF target < middle

SET left

SET right

IF target > middle

SET left

SET right

ELSE

RETURN target

In the row of 20 numbers, that are sorted in order you are looking for a specific number. Find the center of the list and determine if the desired number is less than or greater than the current number you are on. If it is greater than the middle number, then find the middle number in that half of the number list. Do the same thing respectively if the number is less than. Continue this process until you find the desired number.

A.I. Pseudocode:

```
Algorithm BinarySearch(A, target)
    Input: A sorted array A, a target value
    Output: Index of the target in array A, or -1 if not found

    left ← 0
    right ← length(A) - 1

    while left ≤ right do
        middle ← floor((left + right) / 2)
        if A[middle] = target then
            return middle
        else if A[middle] < target then
            left ← middle + 1
        else
            right ← middle - 1
    return -1
```

- Provide an analysis as to the pros and cons of the two solutions.

  I think that my code is a little easier to understand and there is less going on. The AI-generated code maybe is a little more fool proof, and someone who knows what's going on may be able to make a better program following the AI code. A con to the AI code it just indexs the array up or down one value after it determines which way it needs to go.

- How can your solution be improved based on what Copilot provided?

  The only suggestions that it gave me was to define the array, and make it just increment the array up or down, not find the middle again.

- How can Copilot's solution be improved based on what you know?

I think that the AI code is good, but for this specific algorithm, it shouldn't just increment the array one at a time and check again to see if it's found the target it needs to re-find the middle.

- Does the pseudocode in Step 3 and Step 4 match the algorithm you performed in Step 1?

  Yes, I think that my algorithm follows it more closely, but I think that the AI algorithm would still work very well for that. That could also be in part because of the prompt that it was given.

Updated code:

```
WHILE not found
SET array
SET middle
    IF target < middle
        SET left
        SET right
    IF target > middle
        SET left
        SET right
    ELSE
        RETURN target
```

Step 1 By Hand: 1 minutes

Step 2 Approach: 18 minutes

Step 3 Pseudocode: 45 minutes

Step 4 Copilot: 10 minutes

Step 5 Compare and Contrast: 20 minutes

Step 6 Update: 5 minutes