

## Pseudo Code Exercise

### Goal:

The goal of this exercise is to practice understanding that although a computer has the benefit of being extremely fast compared to the speed of humans, humans are able to use certain context clues and common sense to assume information. A computer on the other hand only uses the information that is given to it by the programmer and nothing is ever assumed.

### What is Pseudo Code?

In computer science, pseudocode is a plain language description of the steps in an algorithm or another system. In other words it is an informal way of describing the programming language that does not require any strict rules. This is a great first step before even typing your first line of code as it allows the programmer to think about the logic of their code and ensure that it makes sense in English before attempting to translate it to a programming language.

### Task:

Think of a simple task. Write out step by step instructions in pseudo code that a computer would use to carry out this task. Be mindful that you must include each step of the process, even the ones that seem obvious, or else the computer will not perform it. For example if I want to give instructions on how to draw a square on a piece of paper I can't just simply say "draw four equal lines and make them connect." How would the computer know how long a line is? How would it know that all of a square's angles are required to be 90 degrees?

Here is an example:

1. Fold the paper once vertically then once horizontally
2. Unfold the paper
3. Locate the center of the paper by finding the intersection of both folds done in Step 1
4. Touch the center of the paper using the side of the pencil with a tip
5. Draw a four inch line to the right then stop but don't lift the pencil off the page
6. Rotate the paper exactly 90 degrees to the left then repeat step 4
7. Repeat step 5 two more times then lift the pencil off the paper

**Key Takeaways:**

Step 1 and 2 are included to establish a baseline so that if this code was executed a large number of times, each time would start from the same place. Since Step 1 and 2 are clear and established we can refer to it in Step 3 or any other step. This allows for less redundancy in our code and produces cleaner code. Note that step 4 seems very obvious but this step is very important to not allow errors. If that step isn't included the computer may not know where to start their square and may start drawing their line at the edge of the paper. This might lead the line to run off the page which would cause an error and an incomplete square. These special cases are called edge cases and must be taken into consideration when designing your code. Step 5 and 6 includes very specific measurements to ensure the code executes the same way each time. In Step 6 and 7 we are able to refer back to previous steps since they were clearly defined.

**Now let's have some fun!**

Test your knowledge by creating your own pseudocode. Once you're confident that you've mastered it, find a partner and see if they are able to get the same result you were expecting.