



## Practice 4: Swap

Algorithmic logic to swap values between two temporary variables  
Programming Course in Snap! and Pseudocode

### The Swap Problem

Imagine you have two glasses: one with **orange juice** and another with **milk**. If you want to pour the juice into the milk glass and the milk into the juice glass... how would you do it without mixing them? You need a **third empty glass** (auxiliary) to store one of the liquids while you move the other. In programming, we do exactly the same with variables.

### Objectives of this practice

- Understand the concept of "auxiliary" or temporary variables.
- Learn the correct logical sequence to avoid data loss.
- Differentiate between assigning a value and moving a value.

### The "Third Glass" Logic

To swap the values of two variables (A and B), we cannot simply do  $A = B$  and then  $B = A$ , because performing the first step would erase the original value of A.

### Algorithmic Steps

1. **Save:** We store the value of A in a new variable called `temporary`.
2. **Move:** We pass the value of B to A.
3. **Recover:** We pass the value stored in `temporary` to B.

## THE CHALLENGE: The Variables Magic Trick

Create a program that asks the user for two numbers and then "magically" swaps them before displaying them again.

### Initial setup:

- Create three variables: number1, number2, and auxiliary.
- Ask for the two initial values using **Sensing** blocks.

### Sequence in Snap! (Variables Category):

1. `set [auxiliary] to (number1)`
2. `set [number1] to (number2)`
3. `set [number2] to (auxiliar)`

## Solution Pseudocode

```
Algorithm MagicSwap
  Define num1, num2, aux As Integer
  Write "Enter the first number:"
  Read num1
  Write "Enter the second number:"
  Read num2
  // The swap begins
  aux <- num1
  num1 <- num2
  num2 <- aux
  Write "Abracadabra! Now the first number is ", num1, " and the second is ", num2
EndAlgorithm
```

*Remember: Without the "aux" variable, one of the two values would be lost forever during the process.*

This document is published under license  
 Creative Commons Attribution 4.0 International (CC BY 4.0)