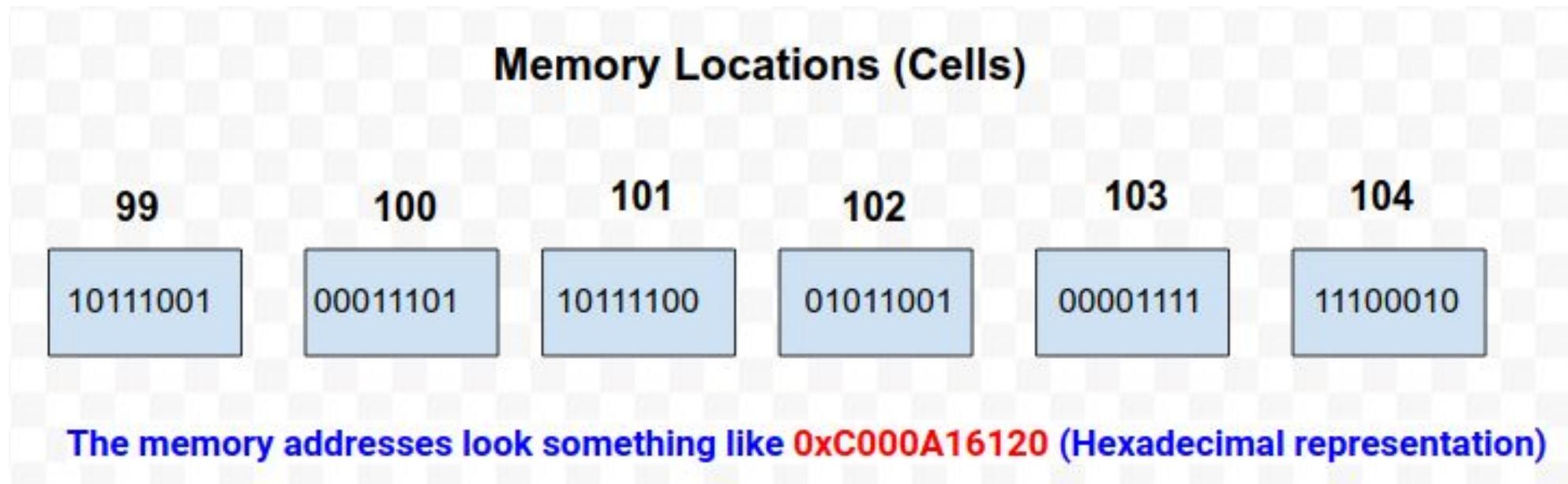


Pointers - 1

- **The Computer Memory (RAM)** can be thought of as a sequence of boxes or cells, placed one after another in a line. **Each cell is labeled with a unique number**, which increments sequentially; this number is the address of the cell or its **memory location**.



- Each cell holds a single value. **Everything the CPU does is fetching and storing values into memory cells.**

Pointers - 2

- **What is a variable?**

A variable is just a convenient, alphanumeric nickname or label for a memory location.

When we write `var a int = 5` we create a label called **a** for a memory location where the value **5** of type **int** will be stored.

In a nutshell: memory is just a series of numbered cells, and variables are just nicknames for memory locations assigned by the compiler.

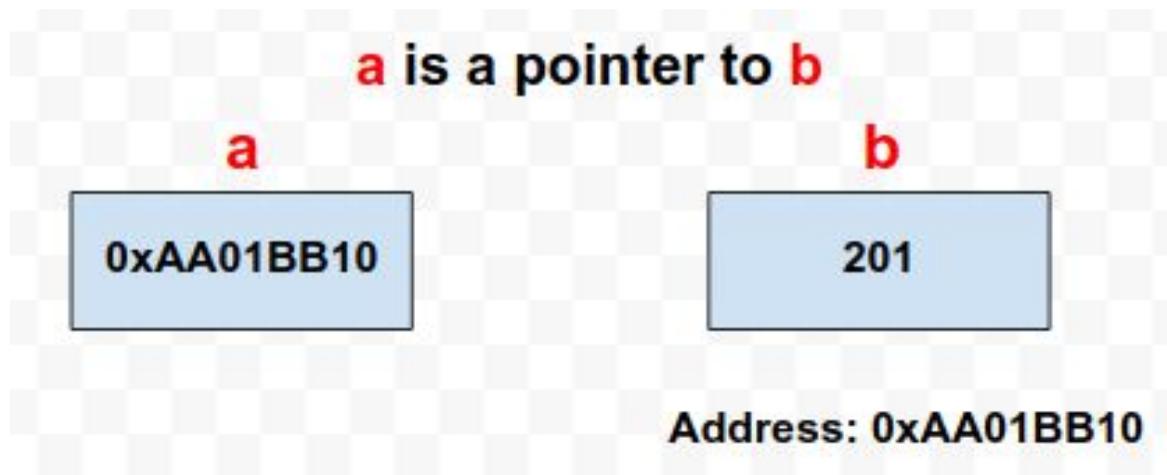
- **What is a pointer?**

A pointer is a variable that stores the memory address of another variable.

The pointer points to memory address of a variable, just as a variable represents the memory address of a value.

A pointer value is the address of a variable or `nil` if it hasn't been initialized yet.

Pointers - 3



- In the above image, variable **b** has value **201** and is stored at memory address **0xAA01BB10**. The variable **a** holds the address of **b**. **Now a is said to point to b or a is a pointer to b.**
- **Pointers have the power to mutate or change the data they are pointing to.**
- Unlike C, **Go has no pointer arithmetic.**