

# References & Pointers

Jonas Hemmett

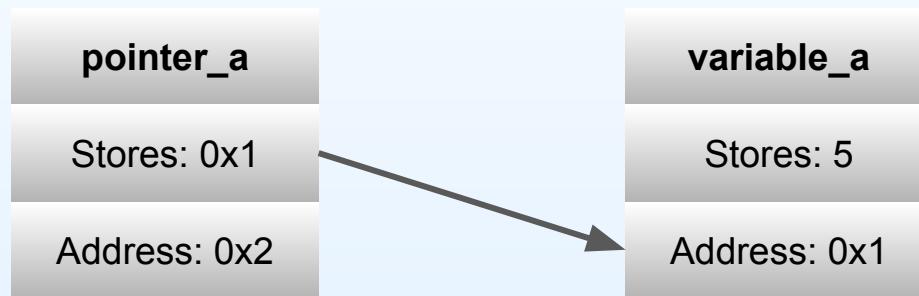
# Variable

<b>variable_a</b>
Stores: 5
Address: 0x1

Before we can look at references and pointers, we first have to look at variables. A variable stores a value at a named location. This value could be an integer, a boolean, etc.

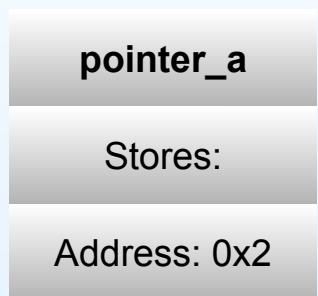
# Pointers

A pointer is variable which can store another variables address as its value.



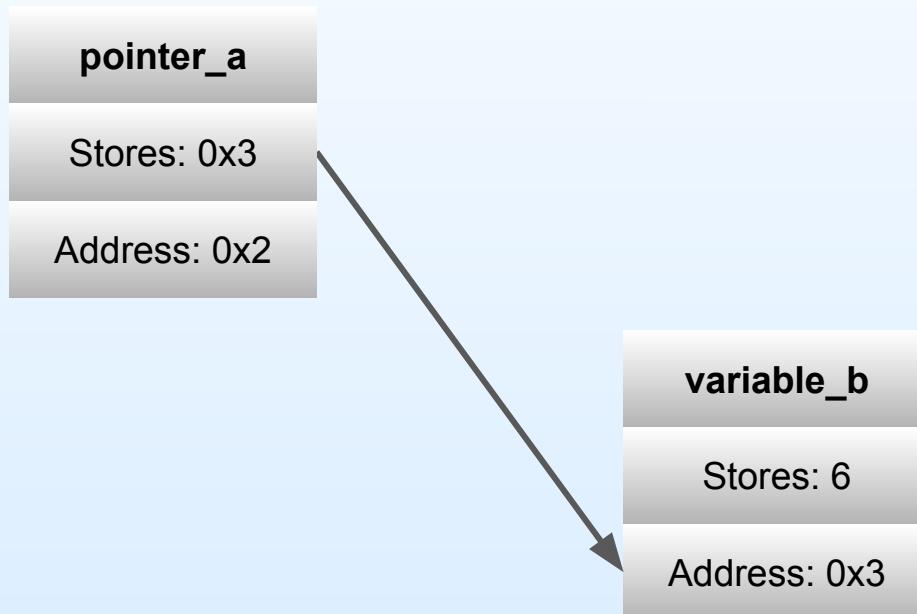
# Pointers

The address it points to can also be null.



# Pointers

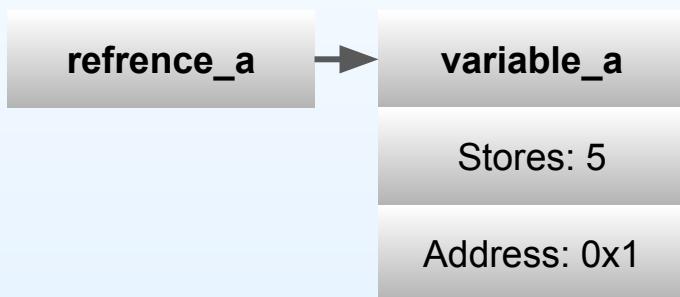
Or resigned to point to another variable.



# Example Pointer Code

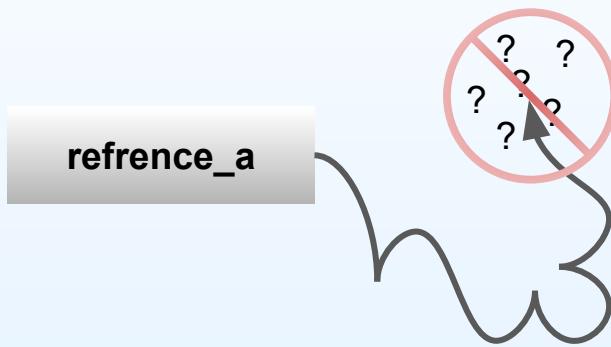
```
int variable_a = 5;  
  
// Creates a pointer named pointer_a. It is denoted using “*”  
int* pointer_a;  
  
// Sets point to point at variable_a  
pointer_a = &variable_a;  
  
// Prints point’s value, which is variable_a’s address (0x1)  
std::cout << pointer_a << std::endl;  
  
// Dereferences and prints pointer_a’s value, which is variable_a’s value (5)  
std::cout << *pointer_a << std::endl;
```

# References



A reference acts as an alias, allowing for an additional name to be given to an already existing variable.

# References



It is important to note that a reference must be initialized to another variable and the variable a reference refers to can't be reassigned.

# Example Reference Code

```
int variable_a = 5;  
  
// Creates a reference named refrence_a, which serves as an alias for variable_a.  
It is denoted using “&”  
int& refrence_a = variable_a;  
  
// Prints point's value, which is var's value (5)  
std::cout << refrence_a << std::endl;
```

# References & Pointers

Both references and pointers provide ways for data to be accessed easily and coded more efficiently. Pointers work better for things that require changing memory address, like linked lists and trees as they can have null values or have their values resigned. But if these features are not needed, then using references will probably work better.

# Works Cited

GeeksforGeeks. “Pointers vs References in C++.” GeeksforGeeks, May 2017, [geeksforgeeks.org/cpp/pointers-vs-references-cpp](https://geeksforgeeks.org/cpp/pointers-vs-references-cpp). Accessed 31 Aug. 2025.