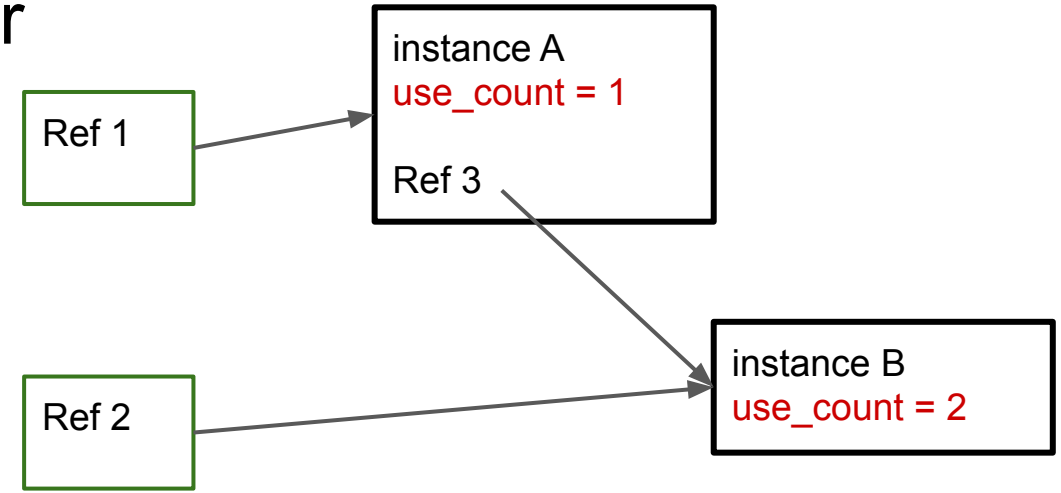
Reference Counting

Godmar Back

Reference Counting

Idea: each object keeps a count of the number of references (or pointers) to it



Count is incremented when a reference is added, decremented when a reference is removed.
If the count reaches zero, the object is freed immediately (no lag)

Manual Reference Counting

Programmer is responsible for updating reference counts

- Example: Microsoft COM

Must follow established set of rules, e.g.

- Call AddRef() when establishing a new reference
- Call Release() when releasing a reference

Still error-prone, but rules have the advantage that a program's correctness can be asserted **locally** (without requiring global context)

Reference Counting using Smart Pointers

Some languages (e.g., C++, Rust) provide language or library support for automatic reference counting when references are created, assigned, copied, and/or overwritten.

C++: std::shared_ptr

Rust: std::rc::Rc

Common to these schemes is that they cannot handle cycles in the object reachability graph