## COMP6991 23T3

Slices & Lifetimes

# Ownership & Borrowing

...recapped



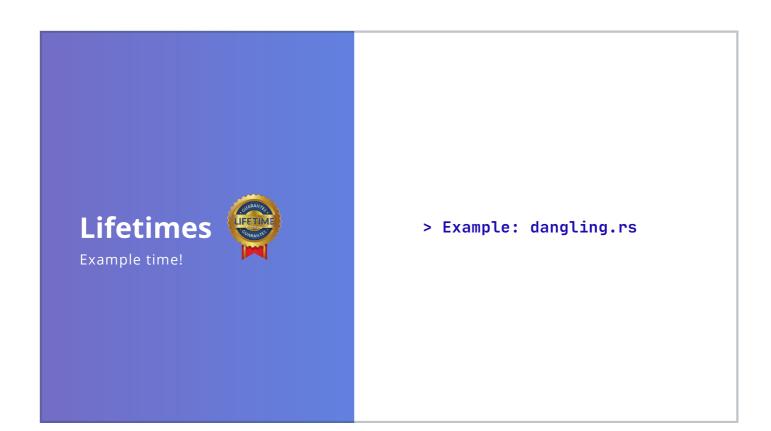
Туре	Requirements	Access
т	Exactly one owner	Read & Write
&Т	Only shared &T borrows can coexist	
&mut T	No other borrows can coexist	Read & Write



	Туре	Layout	Access
	[T; N]	Contiguous, exact length	Owned (or Copy)
	Vec <t></t>	Contiguous, dynamic length	Owned
Slices	&[T]	Shared borrow of a contiguous subsequence	Read only borrow
	&mut [T]	Exclusive borrow of a contiguous subsequence (cannot extend nor shrink)	Read write borrow

### Slices

Туре	Layout	Access
String	Contiguous, dynamic length	Owned
&str	Shared borrow of a contiguous subsequence	
&mut str	Exclusive borrow of a contiguous subsequence (cannot extend nor shrink)	Read write borrow





# Annotations on structs & enums

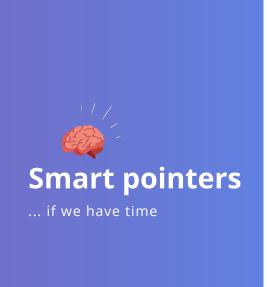
Example time!

> Example: struct\_lifetime.rs



- > What type is a string
  literal? e.g. "foo"
- > What is the lifetime of that literal?
- > What about borrowing a
  global variable?

# Eliding lifetimes > Example: elision.rs Example time!



Туре	Location	Borrowing	Limitations
т	Stack	Owned	Must have a fixed size known at compile-time
Box <t></t>	Неар	Owned	Peformance, memory usage
Rc <t></t>	Неар	Shared without lifetimes!	Read-only, performance, memory usage, reference cycles
RefCell <t></t>	Stack	Owned, allowing dynamic borrowck	Incorrect borrowing causes panic at runtime