

Rust Programming

Henry Oehlrich

April 14, 2023

Roadmap

- ▶ Introduction
- ▶ Memory, the stack and the heap
- ▶ Variable lifetimes and scope
- ▶ References
- ▶ Safety

Rust Concepts

- ▶ Enums
- ▶ Structs
- ▶ Traits
- ▶ Lifetimes
- ▶ Generics
- ▶ Primitives
- ▶ **References and borrowing**

Memory

Memory is for temporary storage of program data at execution

```
fn main() {  
    let x: i32 = 10;  
    let s1: &str = "I'm a string literal";  
}
```

The Stack

- ▶ Fast way to store and retrieve data
- ▶ Last in first out
- ▶ Must know the size of the data



The Heap

- ▶ Slower to store and retrieve data
- ▶ Need not know the size of the data
- ▶ Able to resize, copy, and clone on the fly



Memory

address	value
...	...
0x7fffac86e908	00000000
0x7fffac86e909	00000000
0x7fffac86e90a	00000111
0x7fffac86e90b	11100111
...	...

output:

```
let x: u32 = 2
println!("x ad
```

```
x addr: 0x7fff
```


