

CS631-02

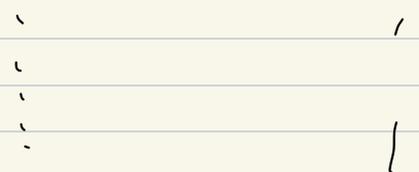
Rust Basics

2026-02-03

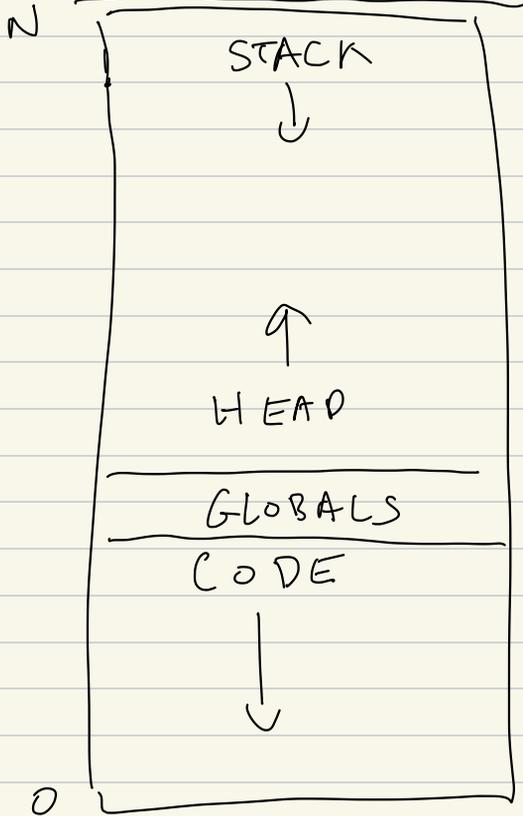
Lab 01



Memory



program process
↓
executing program



compiled
C code
Rust code
Assembled
assemble
RISC-V

C globals

```
int x = 3;  
int y = 4;
```

```
→ main() {  
    int z = 5;  
    z = foo(z);
```

}

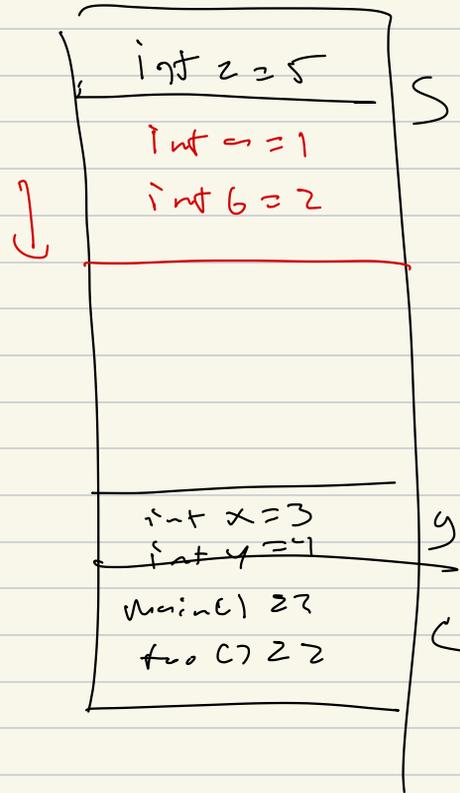
```
int foo(int x) {
```

```
    int a = 1;  
    int b = 2;
```

```
    return a + b + x;
```

```
}
```

STACK



C heap

```
char *str_p;
```

```
str_p = (char *) malloc(sizeof(char) * 32);
```

```
str_p[0] ... str_p[31]
```

```
free(str_p)
```

```
let s = String::new()
```

~~str~~

