Today

- Code: house finch & moving *Paramecium*
- Data structures: vectors & indexing
- Repetition structures:
  - counter control using for

house finch continued

Paramecium movement continued

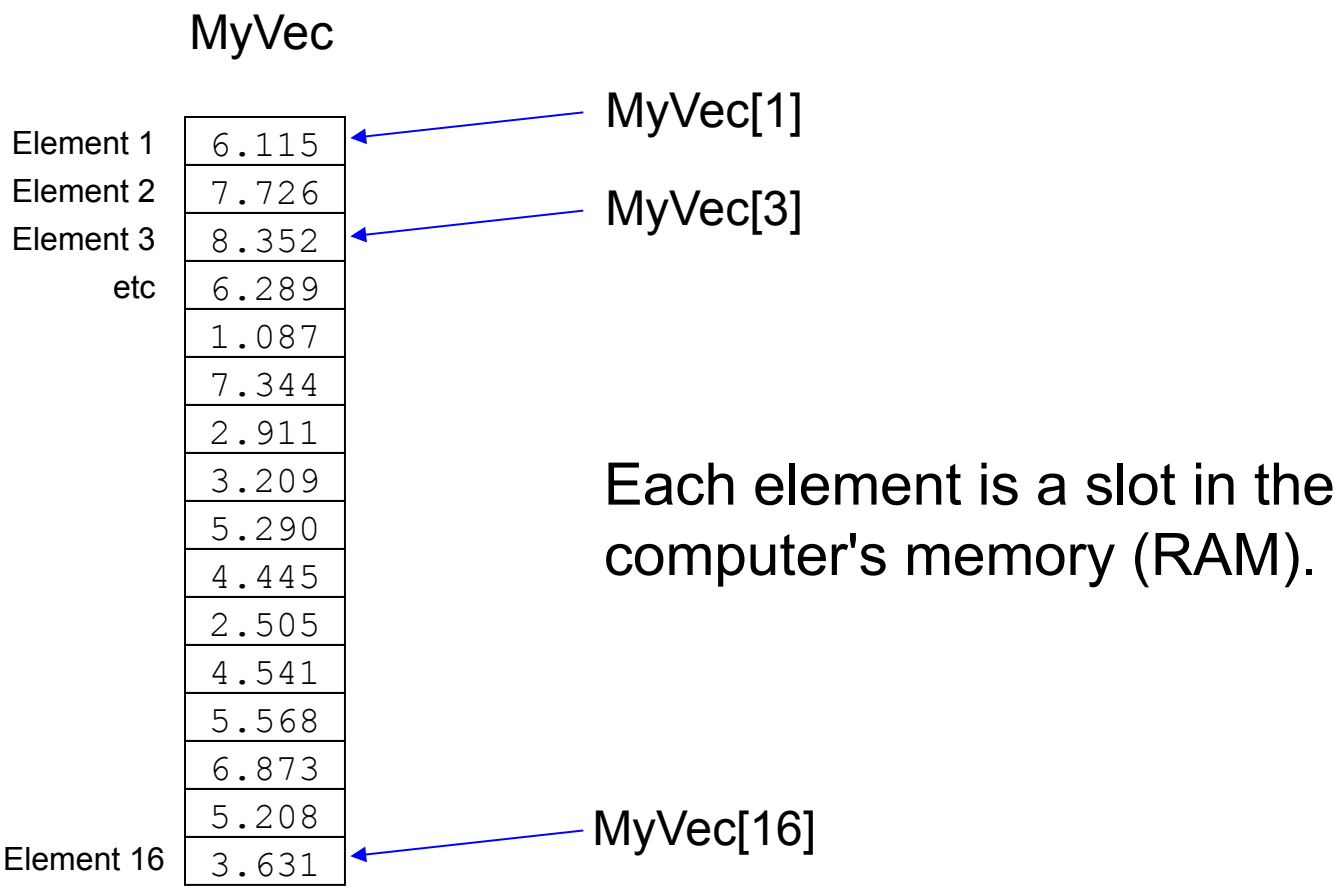
Data structures

- Scalar: a single number
- Vector: multiple numbers, 1 dimension

Data structures

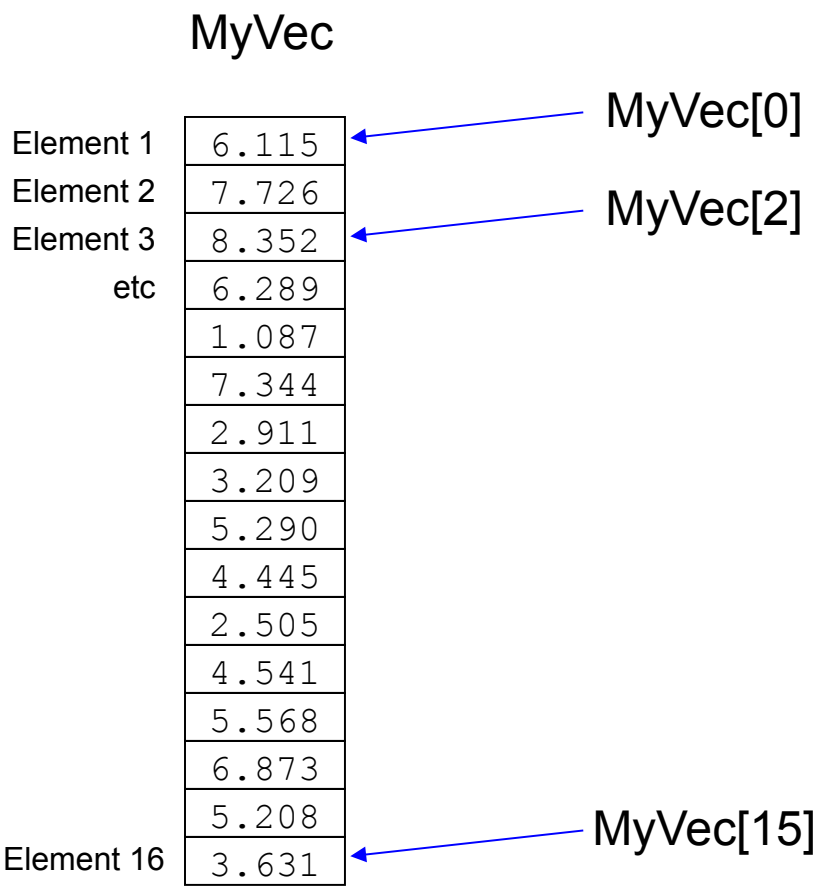
- Scalar: a single number
- Vector: multiple numbers, 1 dimension

Vectors: indexing



R

Vectors: indexing



Python and C use offset indexing

0 is first element

C, Py

R: `for` repetition structure

Most programming languages have a specialized structure for counter-controlled repetition (usually called for)

```
for ( i in 1:n ) {
    expression
}
```

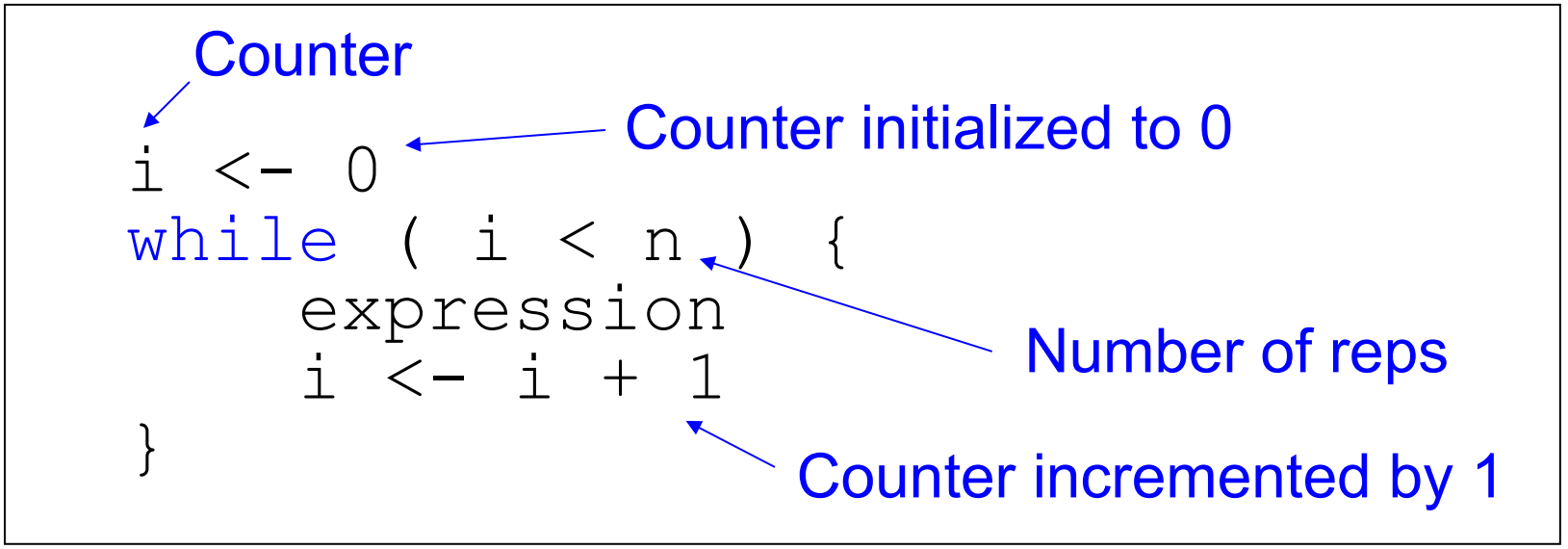
R: `for` repetition structure

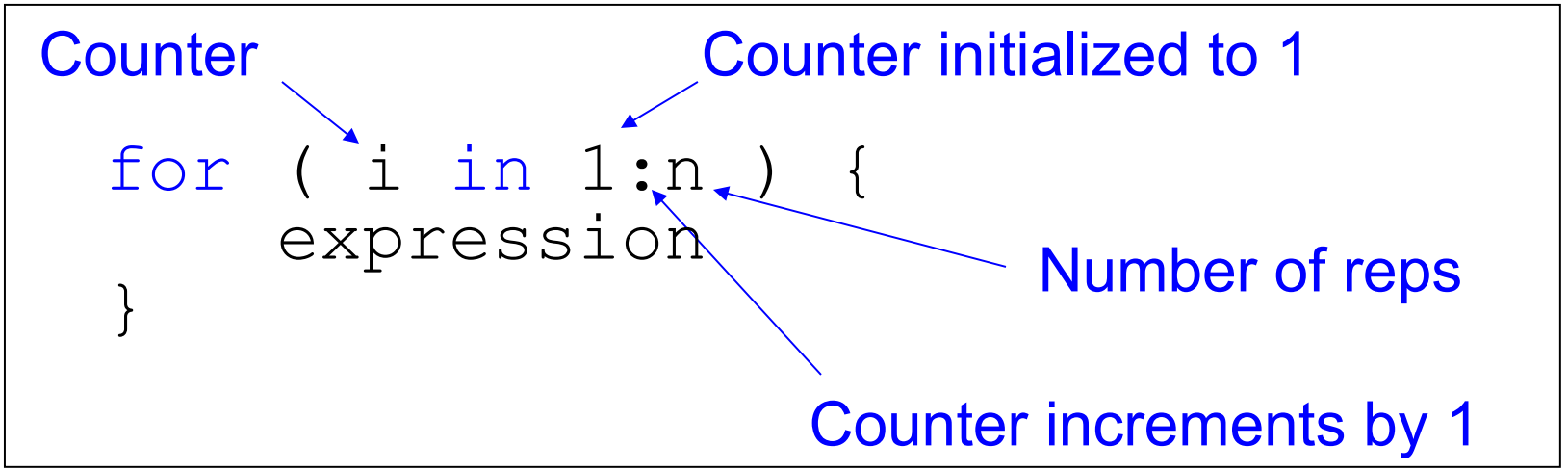
Example

```
for ( i in 1:10 ) {
    j <- i * 2
    print(j)
}
```

What does this do?

The 4 components of counter control using `while` or `for`





R

R: `for` repetition structure

Correct

```
for ( i in 1:n ) {
    expression
}
```

Incorrect

```
i <- 1
for ( i in 1:n ) {
    expression
    i <- i + 1
}
```