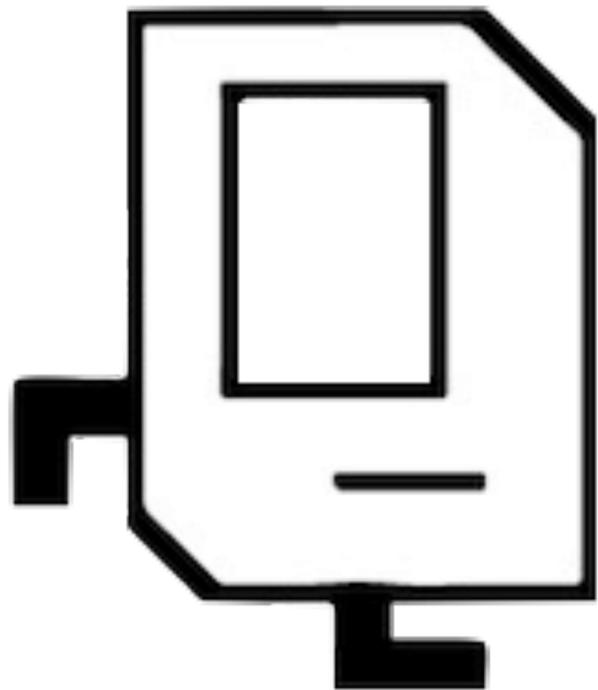


Let's learn!

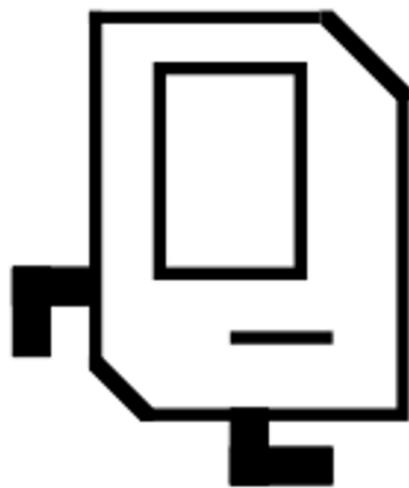


Piech and Sahami, Code in Place



Meet Karel the Robot

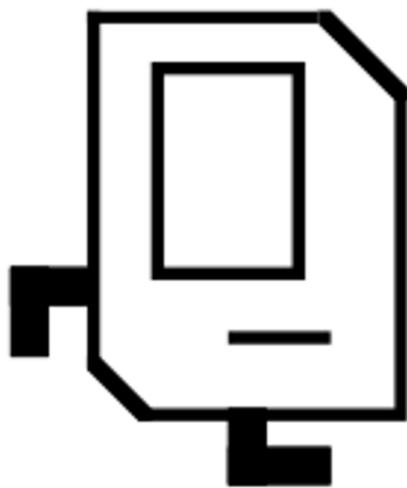
Rich Pattis



Good morning



Karel Speaks Python

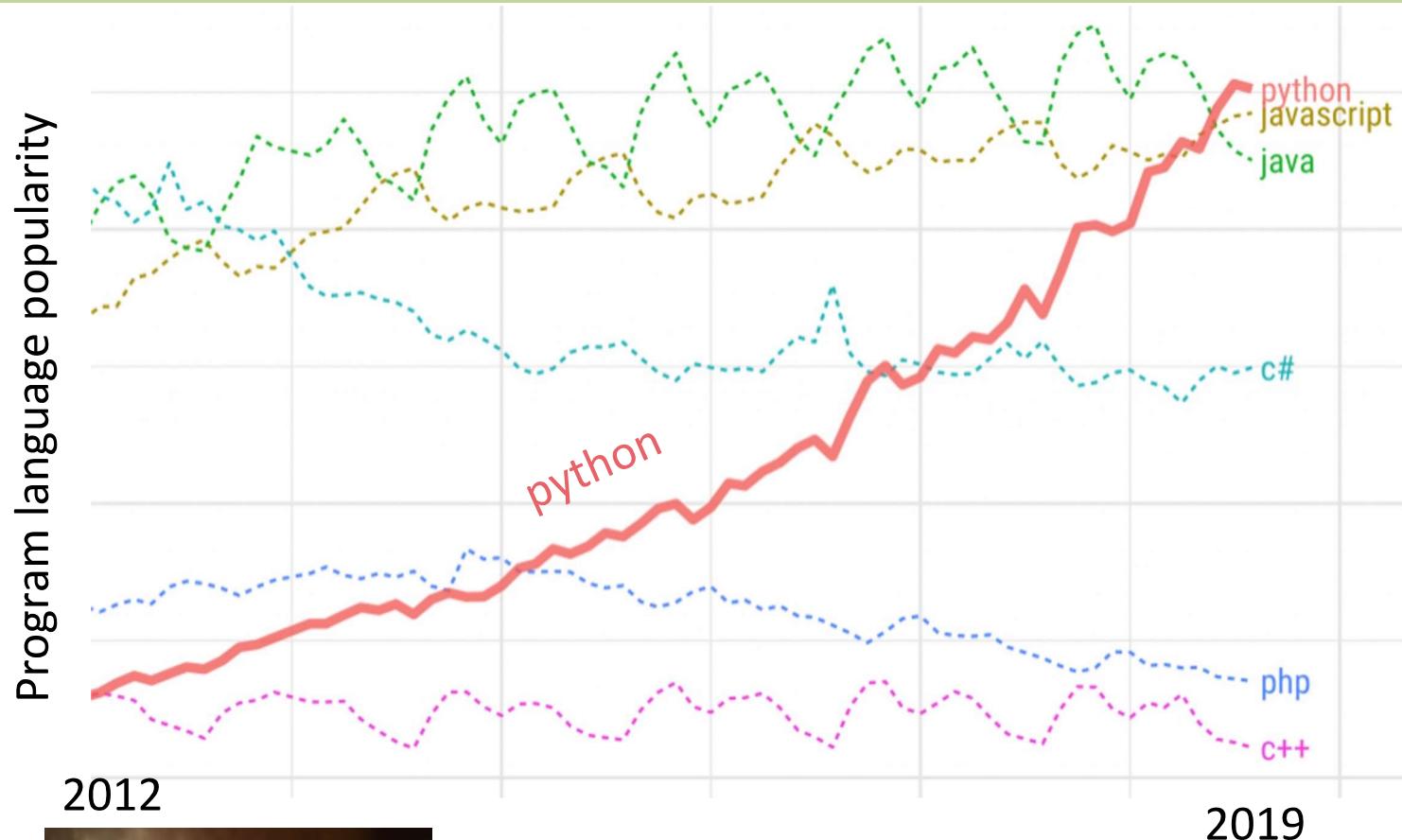


Piech and Sahami, Code in Place

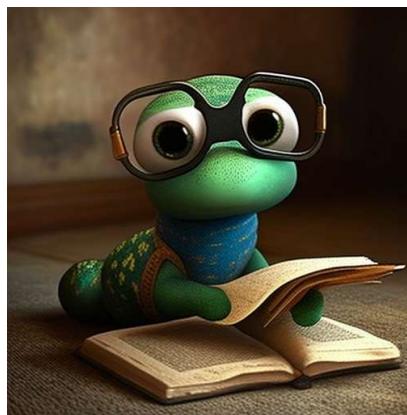


Why Python?

1



2



<https://stackoverflow.blog/2017/09/06/incredible-growth-python/>



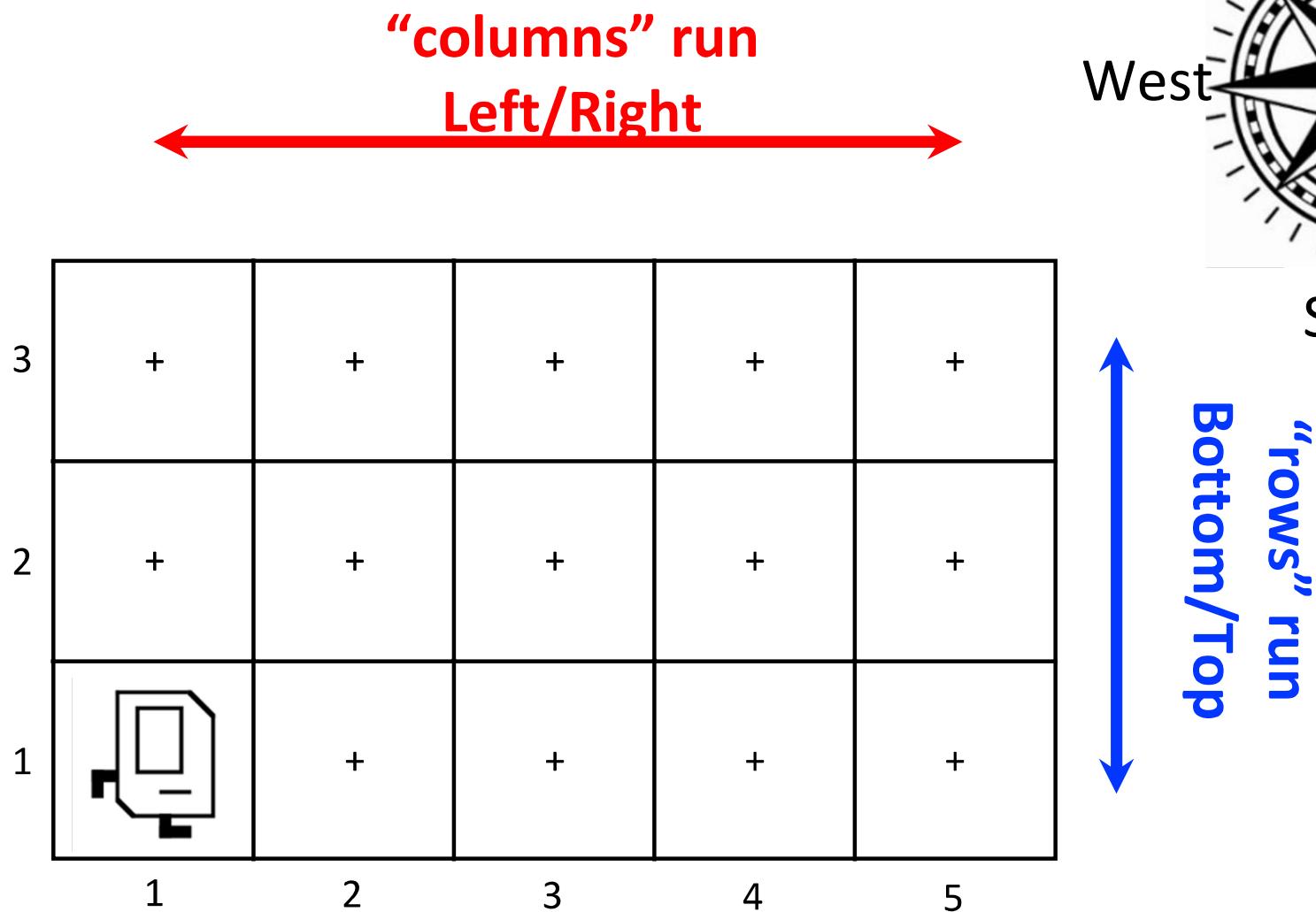
Guido van Rossum



Piech and Sahami, Code in Place



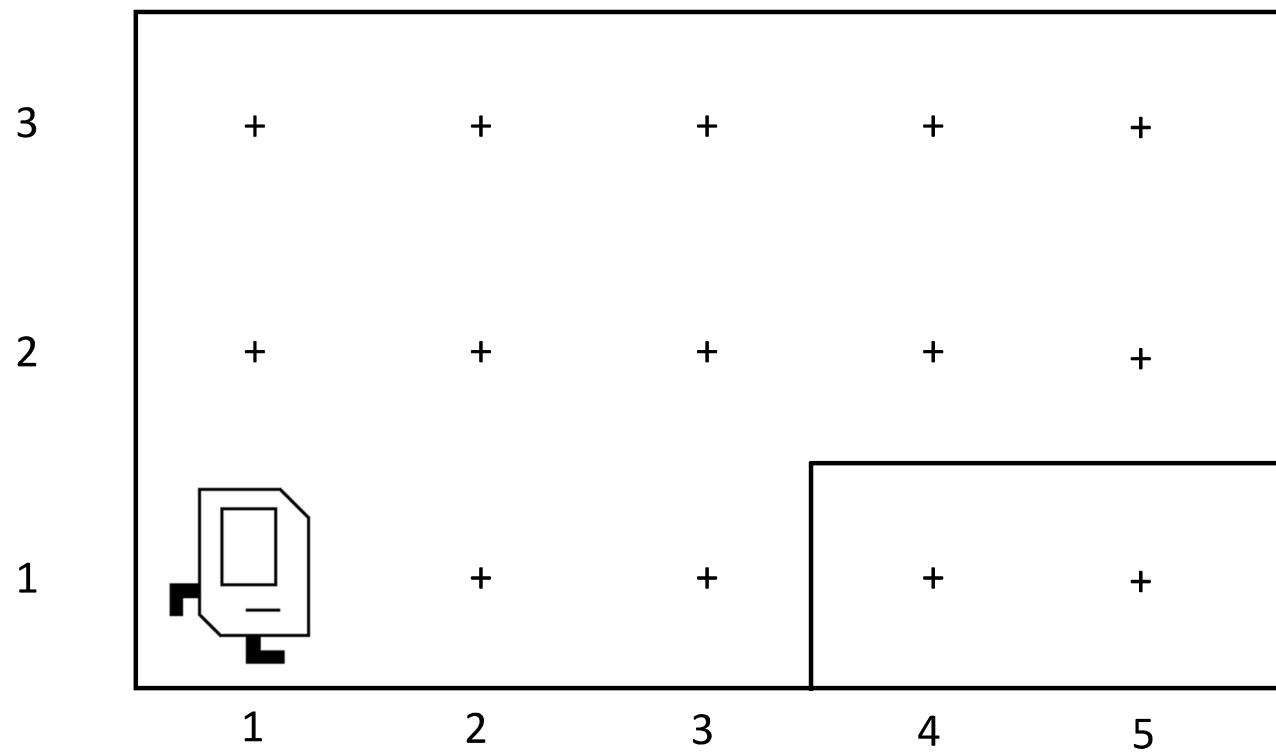
Karel's World



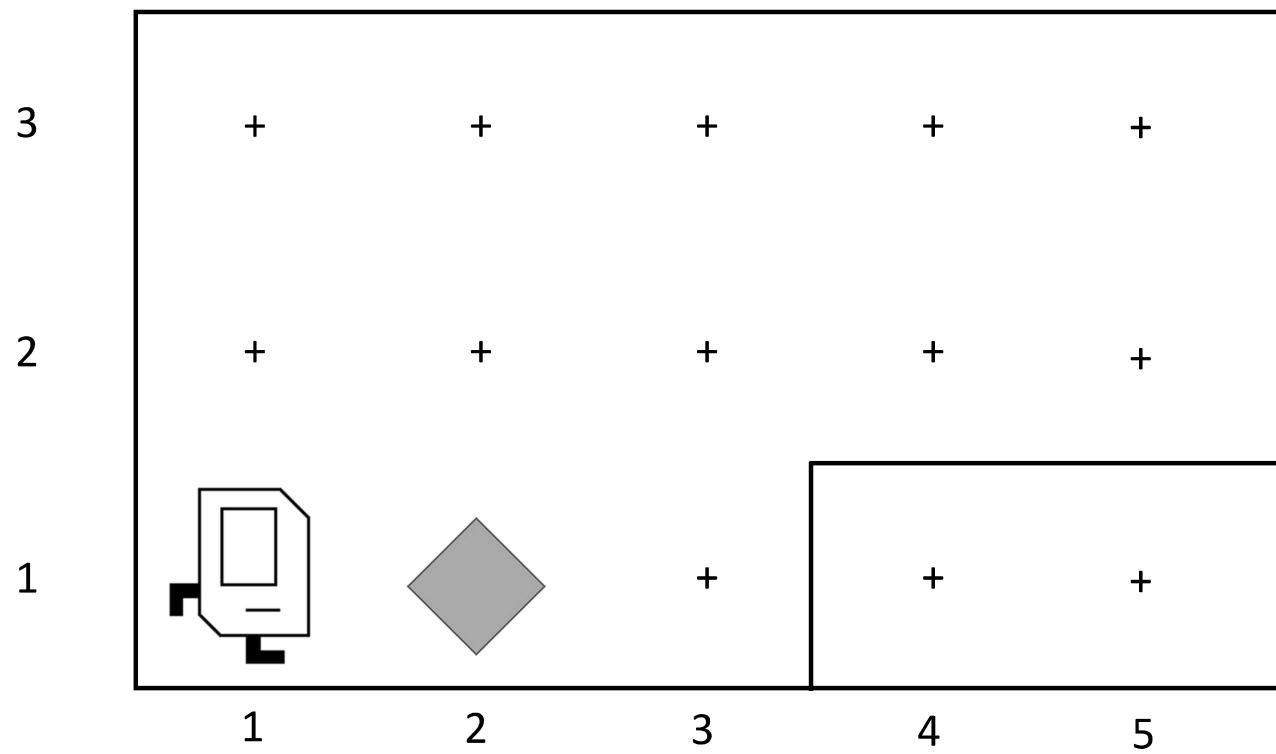
Piech and Sahami, Code in Place



Walls



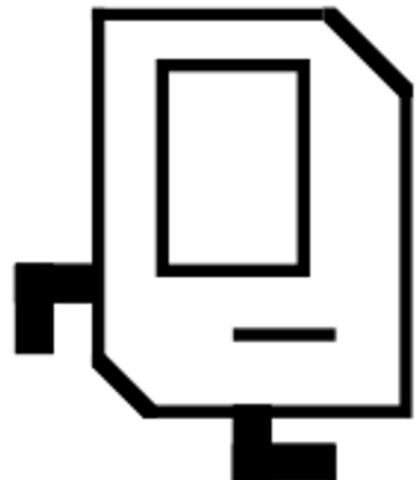
Beepers



Piech and Sahami, Code in Place



Knows Four Commands



move()

turn_left()

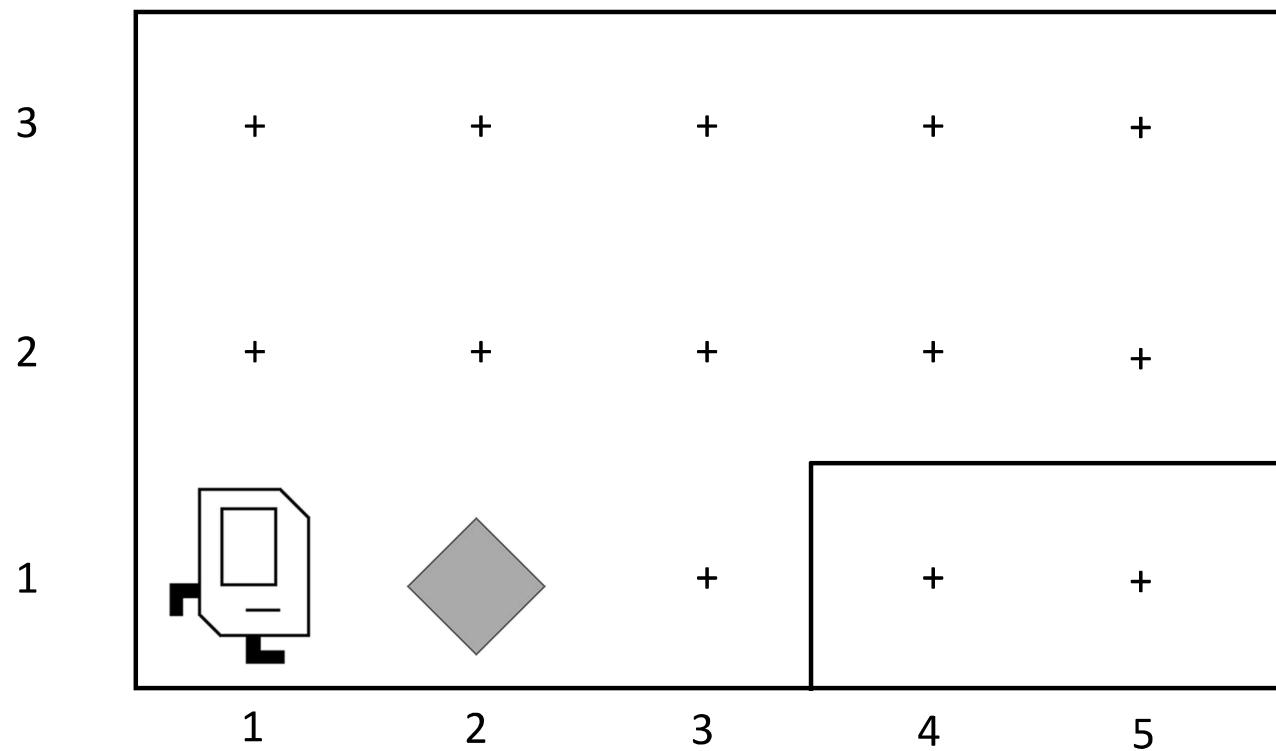
put_beeper()

pick_beeper()

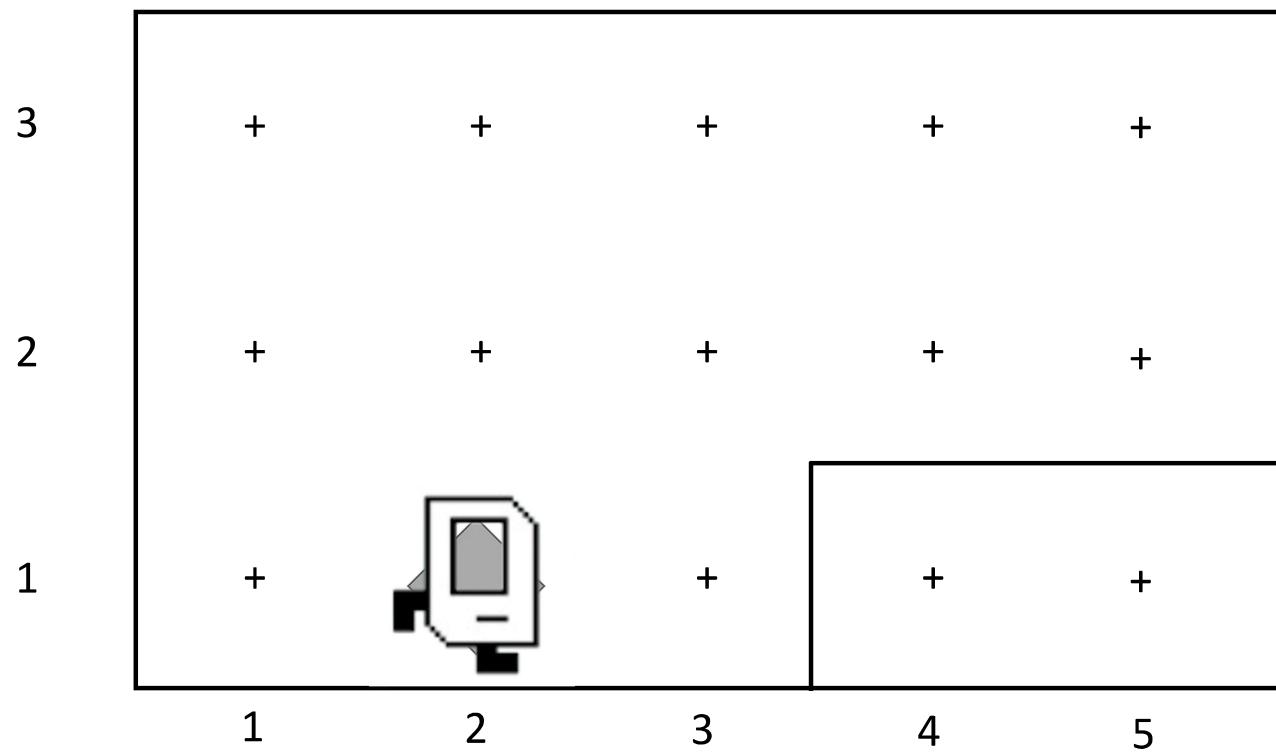


move ()

move()

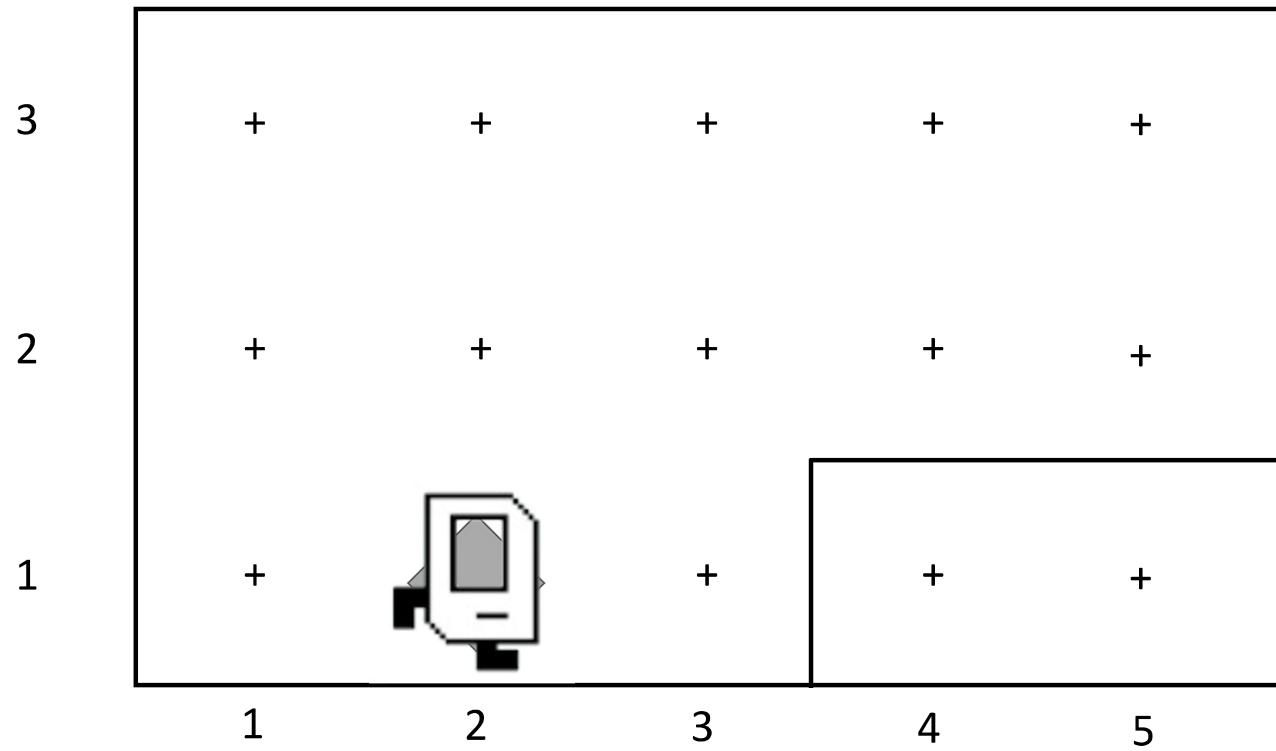


move()

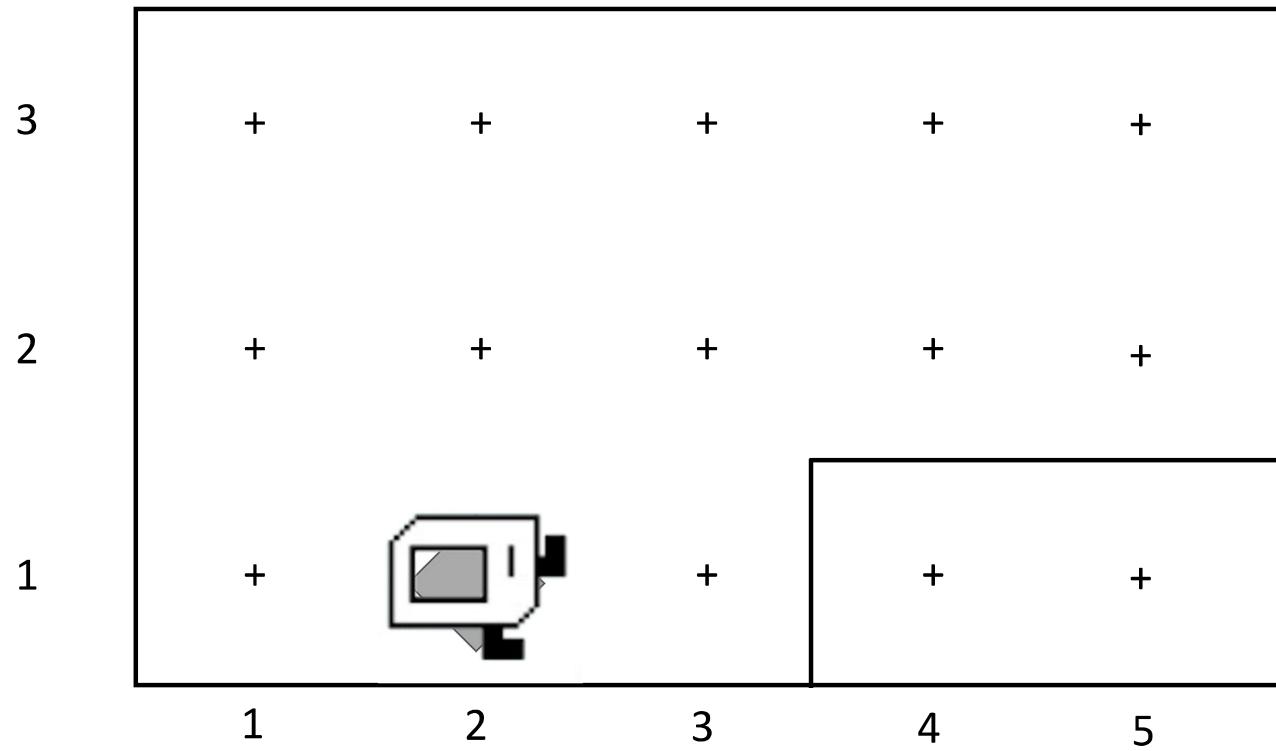


turn_left()

`turn_left()`

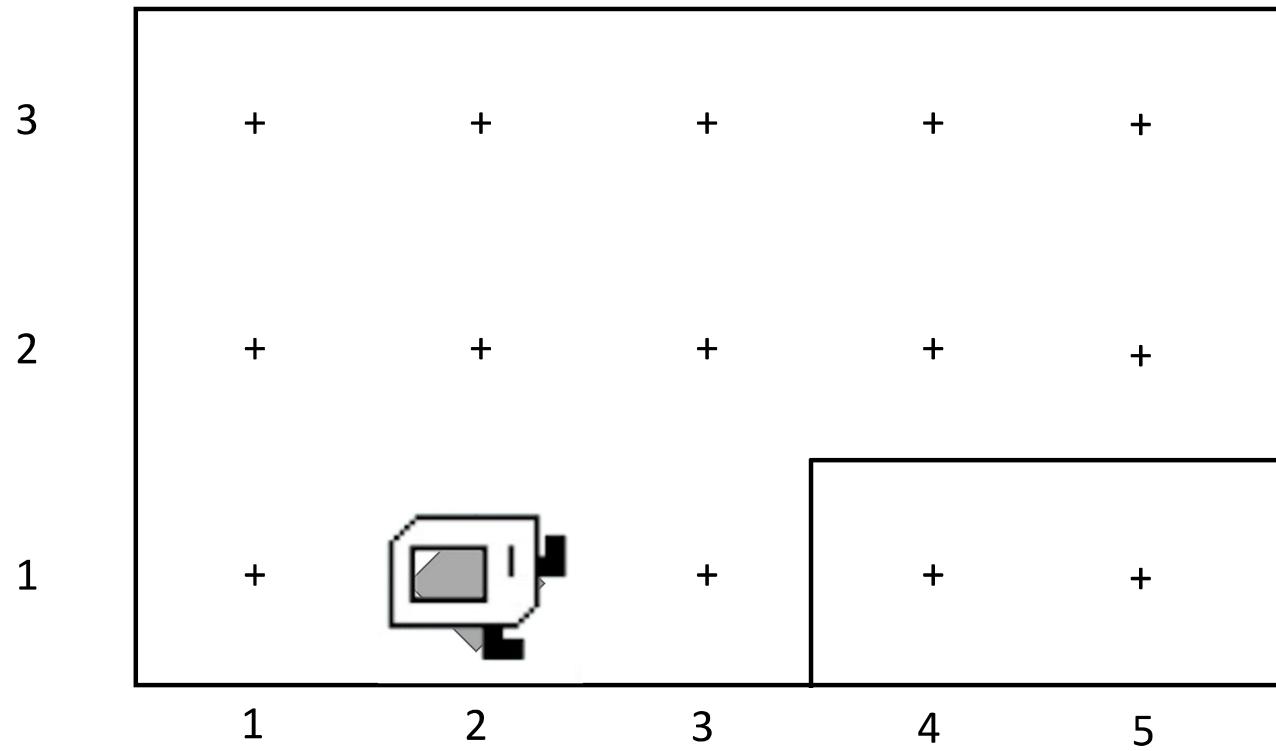


`turn_left()`

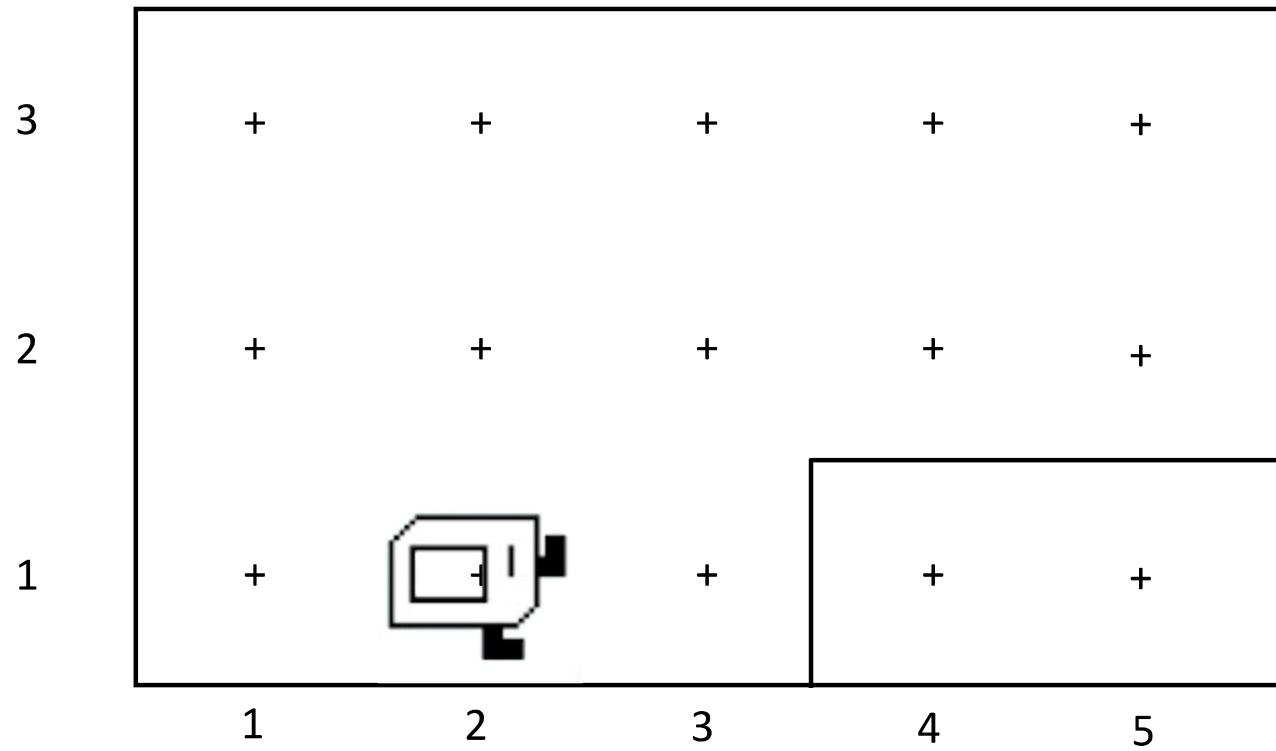


pick_beeper()

`turn_left()`

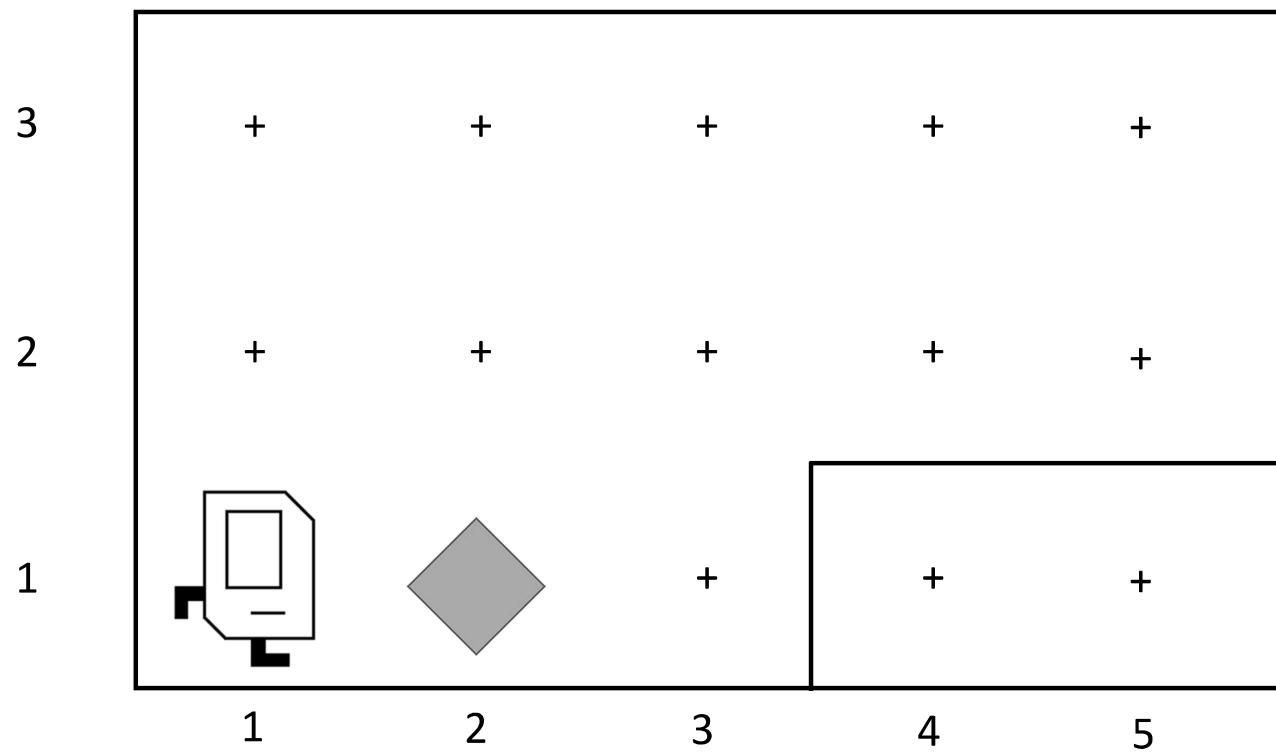


`turn_left()`



Make Sense?

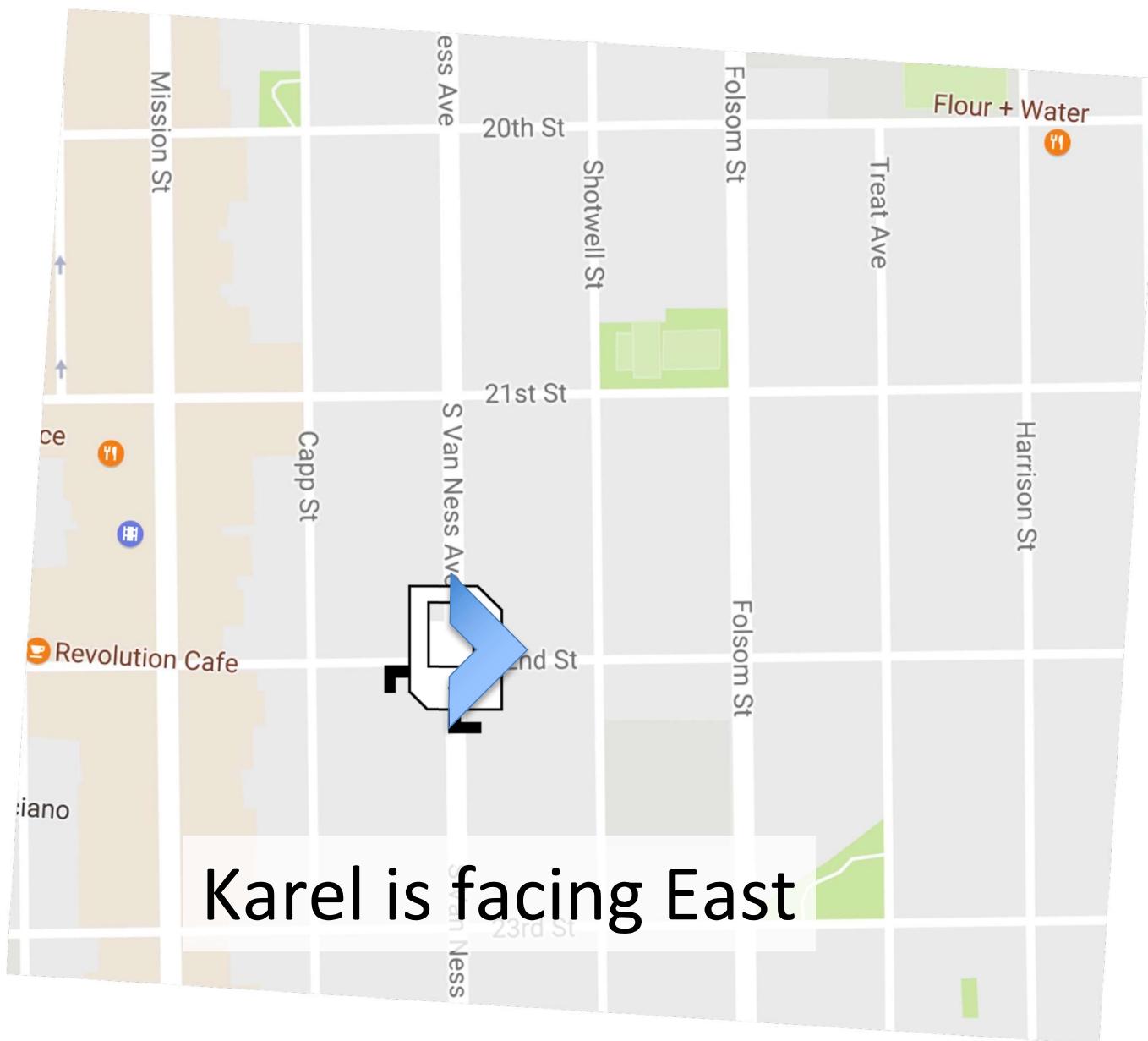
Bird's Eye View



Piech and Sahami, Code in Place



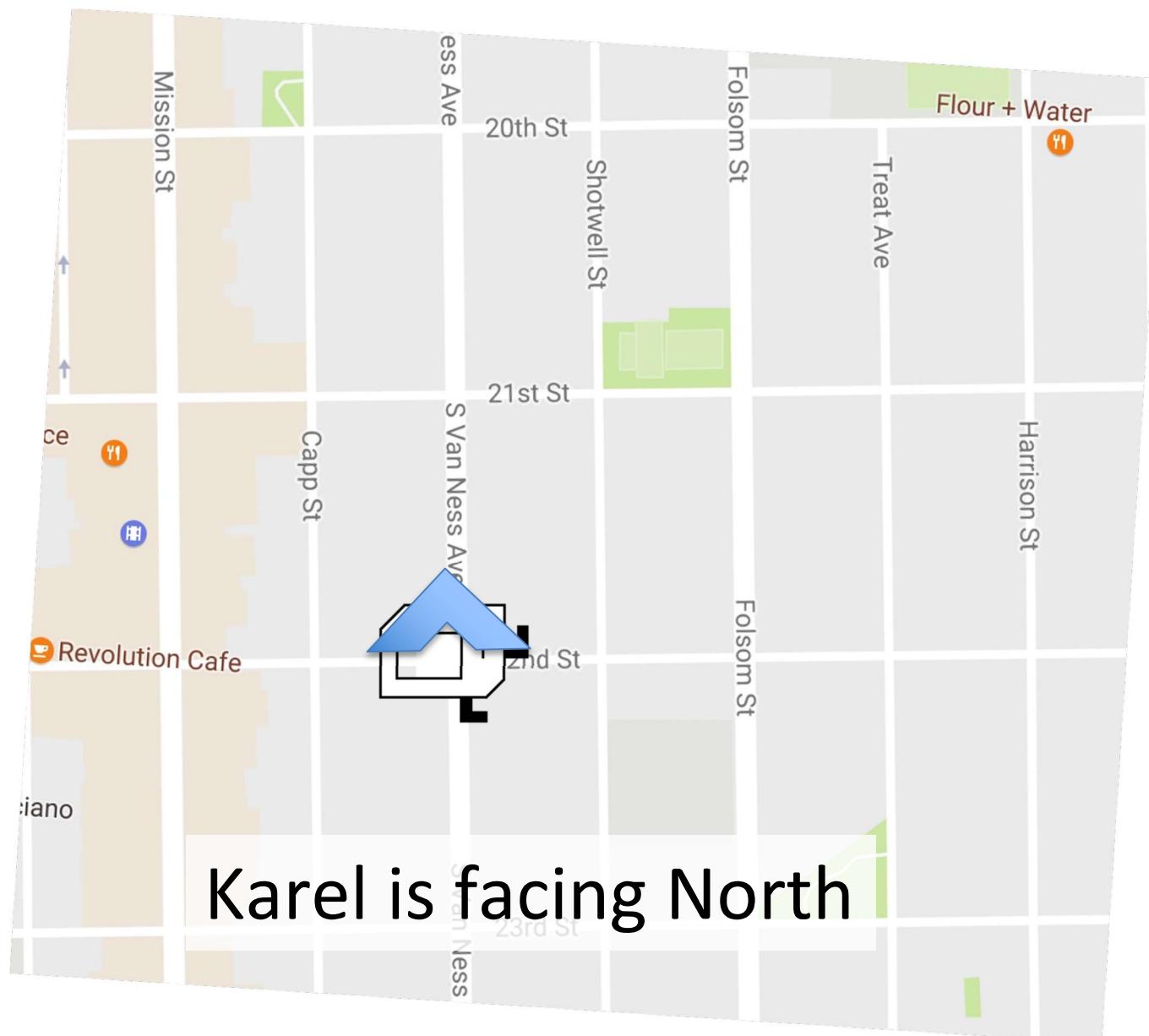
Bird's Eye View



Piech and Sahami, Code in Place



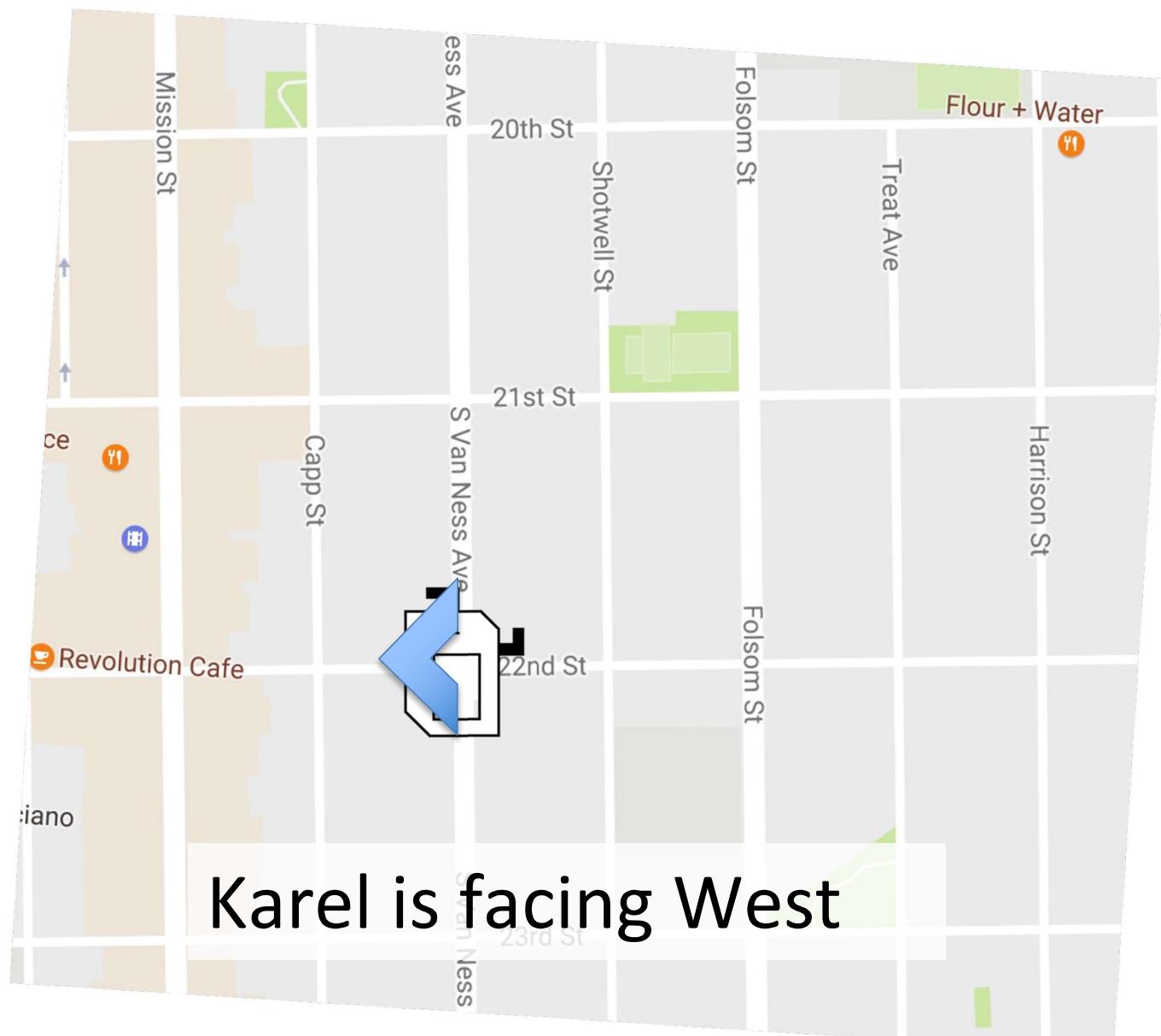
Turn Left



Piech and Sahami, Code in Place



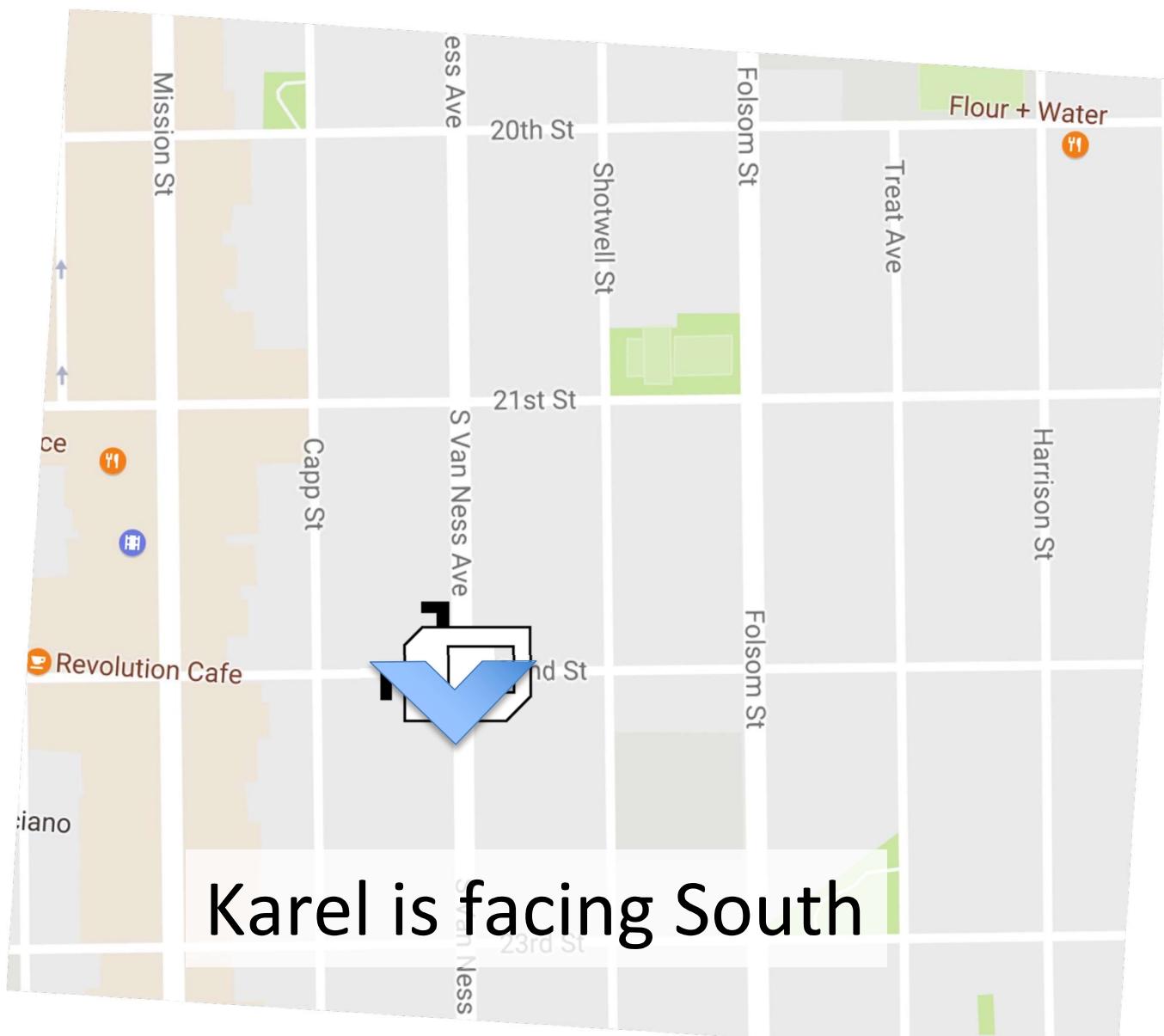
Turn Left



Piech and Sahami, Code in Place



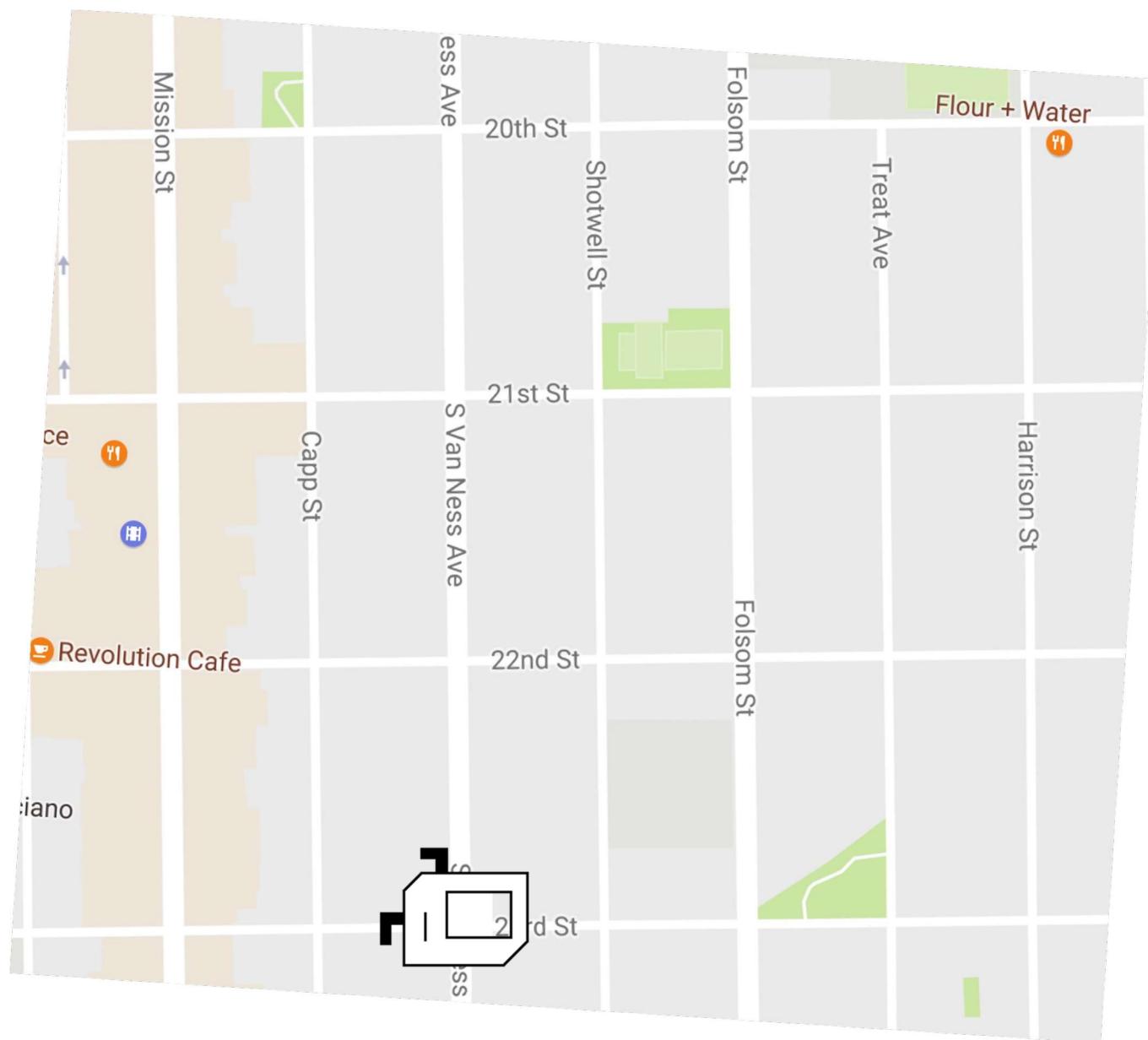
Turn Left



Piech and Sahami, Code in Place



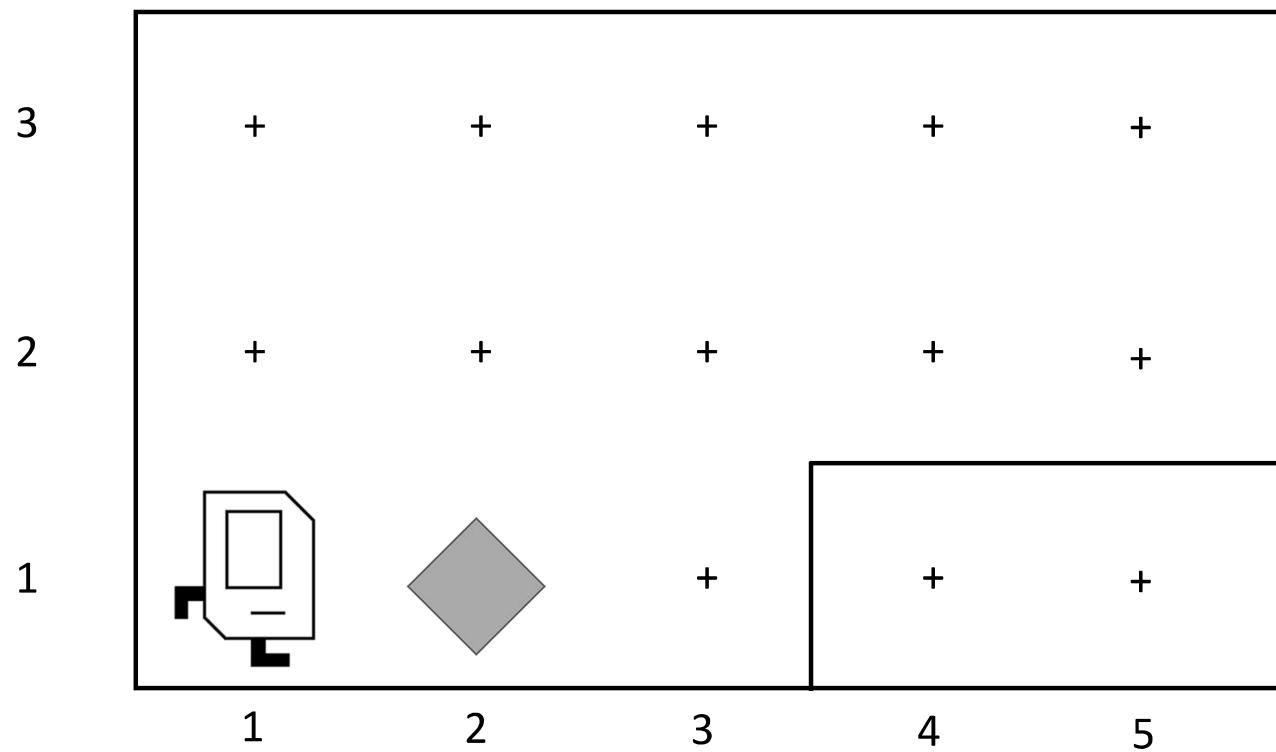
Move



Piech and Sahami, Code in Place



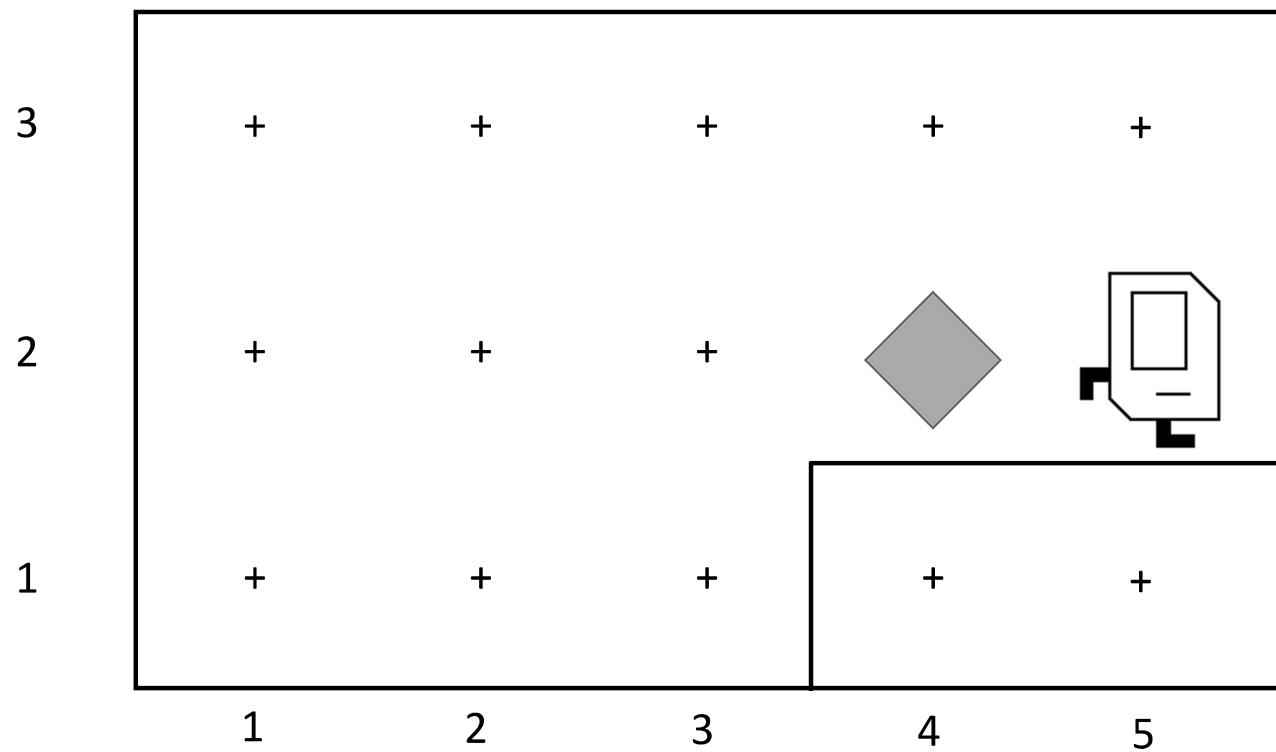
First Challenge

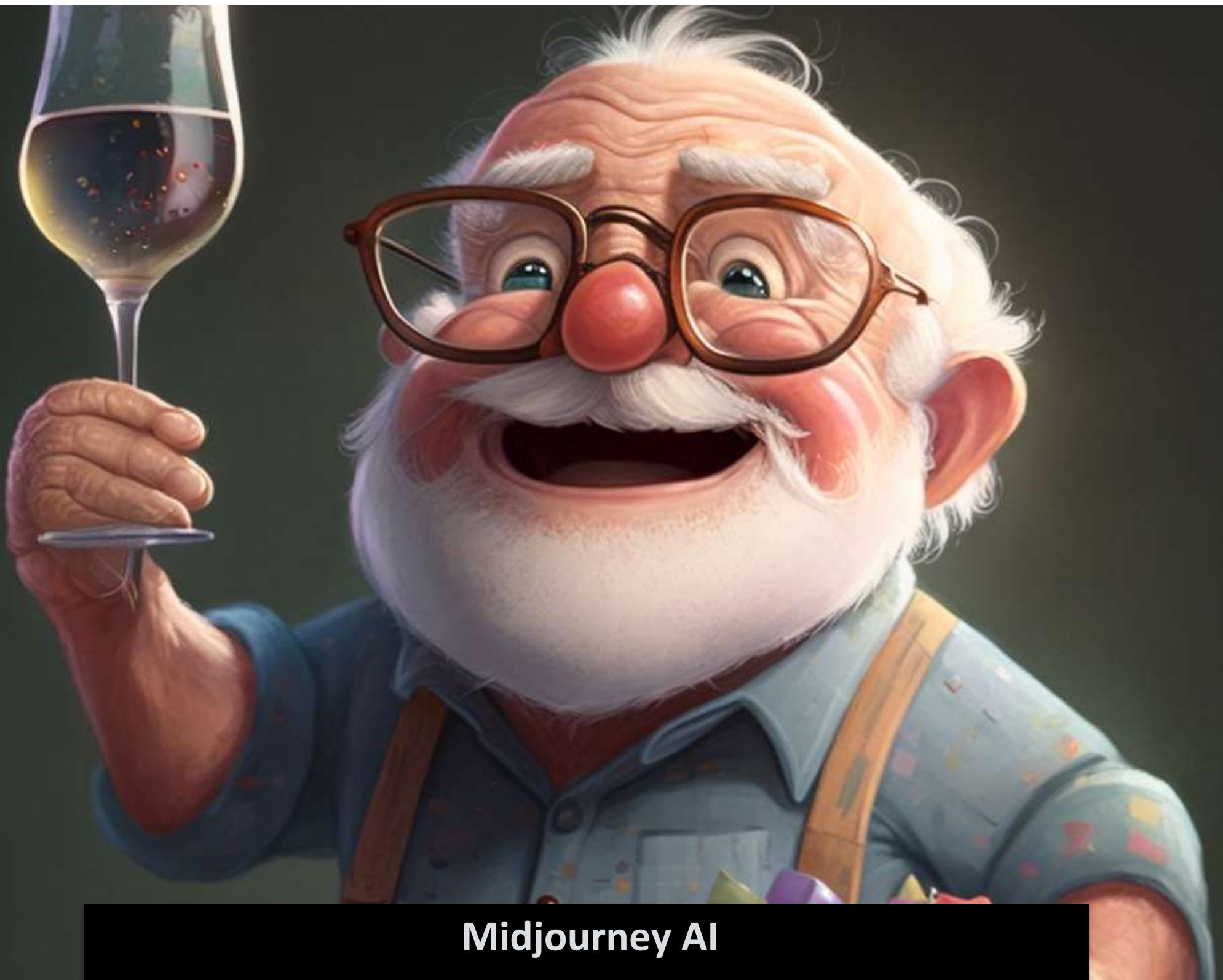


Piech and Sahami, Code in Place



First Challenge





Midjourney AI

Learn By Doing





The Python IDE
for Professional
Developers

[DOWNLOAD](#)

Full-fledged Professional or Free Community

Piech and Sahami, Code in Place



Function Definition

```
def name():
    function statements
```

This adds a new
command to Karel's
vocabulary



Anatomy of a Program

Import Packages

Program



Anatomy of a Program

Import Packages



Anatomy of a Program

Import Packages

main function

helper functions

start program



Anatomy of a Program

Import Packages

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

helper functions

start program



Anatomy of a Program

Import Packages

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

```
def turn_right():
    turn_left()
    turn_left()
    turn_left()
```

start program



Anatomy of a Program

Import Packages

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```



Anatomy of a Program

```
from karel.stanfordkarel import *
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

```
def turn_right():
    turn_left()
    turn_left()
    turn_left()
```

```
if __name__ == "__main__":
    run_karel_program()
```



Anatomy of a Program

```
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()

def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```



Anatomy of a Program

```
from karel.stanfordkarel import *
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

}

This piece of the program's **source code** is called a **function**.

```
def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```



Anatomy of a Program

```
from karel.stanfordkarel import *
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

This line of code gives the **name** of the function
(here, the name is: **main**)

```
def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```



Anatomy of a Program

```
from karel.stanfordkarel import *
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

```
def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

This line of code gives the *name* of
the function
(here, the name is: **turn_right**)



Anatomy of a Program

```
from karel.stanfordkarel import *
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

```
def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```

This is called a **code block**
(Note the indenting)



Anatomy of a Program

```
from karel.stanfordkarel import *
```

```
def main():
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_right()
    move()
    put_beeper()
    move()
```

This is called a **code block**
(Note the indenting)

```
def turn_right():
    turn_left()
    turn_left()
    turn_left()

if __name__ == "__main__":
    run_karel_program()
```



Anatomy of a Program

```
from karel.stanfordkarel import *
```

```
def main():
```

```
    move()
```

```
    pick_beeper()
```

```
    move()
```

```
    turn_left()
```

```
    move()
```

```
    turn_right()
```

```
    move()
```

```
    put_beeper()
```

```
    move()
```

This is called a **code block**

(Note the indenting)

```
def turn_right():
```

```
    turn_left()
```

```
    turn_left()
```

```
    turn_left()
```

```
if __name__ == "__main__":
    run_karel_program()
```

