WEEK 1

# WELCOME TO SECTION!!!

Section Leader
Eesha Tariq

# Introductions

# A Little About Me

- ★ My name is  Esha Tariq.  I'll be your Section  Leader for Code in Place, and everyone here majoring/will be your section-mates!
- ★ I am working in Software Engineering.
- ★ I am Top 20th Github Contributor
- ★ I enjoy coding in python.

# What About You All?

Go ahead and share:

1. **Your name**
2. **Where you're tuning in from**
3. **One thing you're looking forward to (it doesn't have to be from Code in Place)!**

If you aren't warmed up and comfortable with speaking just yet, that's fine! You can share directly to everyone in the chat or you can message me and I can read out your introduction for you!

Another question: what can I do to make you feel included? Please feel free to private message me in the chat if there's anything I can do to make you more comfortable.

# Breaking the Ice

In breakout rooms:

- Share your names one more time!!!
- Icebreaker Question: What's your dream project?
- If no one wants to share first, the person who is geographically closest to Stanford shares first!

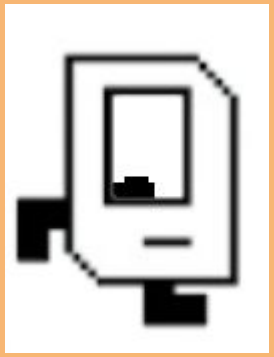
# Lecture Review

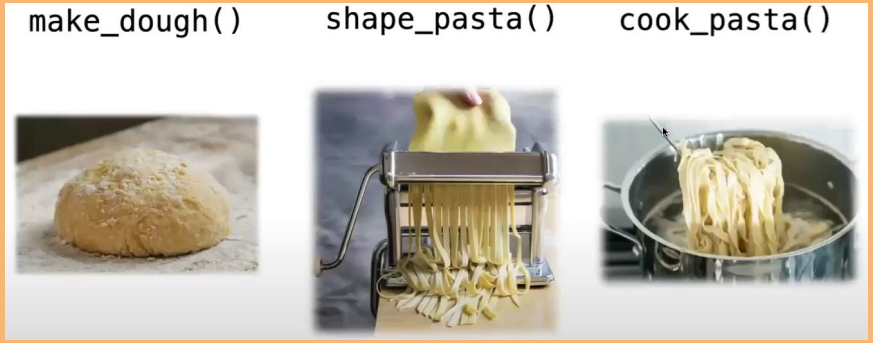
# What We've Learned

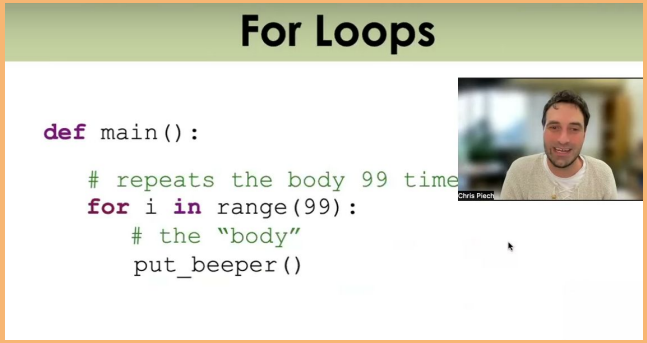
Before we get into our sample problem for today, let's review a bit. We've learned:

- The basics about  Karel, the magnificent and wonderful robot
- Functions, a way of breaking down big problems into smaller chunks
- Control  Flow, loops and conditional statements which guide our programs

This is a **LOT** of content, especially if you are newer to CS!





# Let's review and refresh these concepts a bit!

# Karel Overview

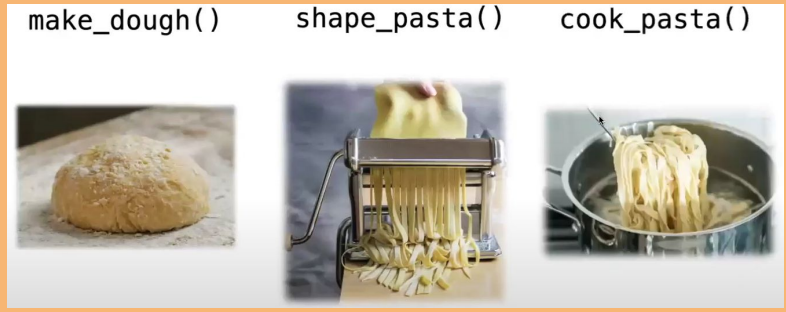

- Karel is a small, but mighty robot!!!
- It has a few basic commands including: **`move()`**, **`turn_left()`**, **`pick_beeper()`**, and **`put_beeper()`**
- On its own, Karel has limited functionality, but with the help of our code, we can make great things happen!

# Functions Overview

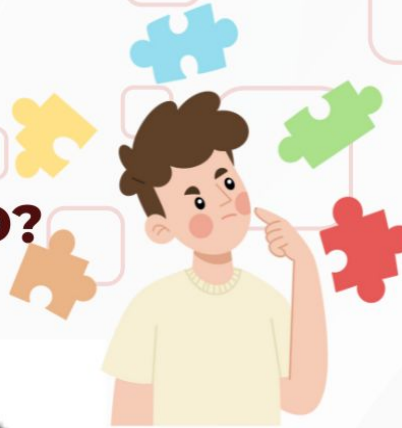
When you think of functions, recall Chris and Mehran's analogy to **making pasta**. Each function has a specific purpose which breaks down a larger problem into smaller chunks—just like steps in making pasta from scratch!

To make a function, you need to define it using the def keyword. Afterwords, write the code you want the function to run. Make sure the code is indented below the function name like so:

```
def function_name():
    # Function code goes here!!!
```



# WHICH CONTROL FLOW SHOULD BE USED?



1. Move Until Wall:
2. Obstacle Check:
3. Jump Over a Hurdle:
4. Place 5 Beepers:

# Control Flow Overview

**For-loop (definite loop):**

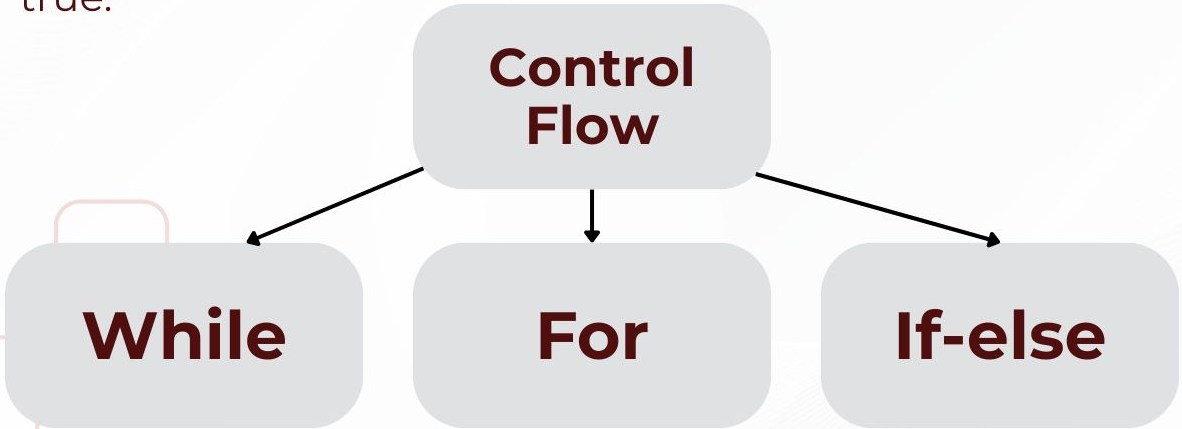
- Performs some block of code, a specific amount of times.

**While-loop (indefinite loop):**

- Continuously perform a block of code until what's being tested is evaluated to false.

**If-statement:**

- Tests for truth. Performs a block of code only when evaluated to true.



# For-Loop

An example for-loop that you may see and use with  Karel:

```
def turn_right():
    for i in range(3):
        turn_left()
```

This loop is also called a *definite loop* because we know where it ends, when i reaches 3. (Be careful to remember that i begins at 0 when we start our loop!!!)

# While-Loop

An example while-loop that you may see and use with  Karel:

```
def move_to_wall():
    while front_is_clear():
        move()
```

This loop is also called an *indefinite loop* because it will run until the associated condition becomes false, which may be never! Who knows? You will, hopefully.  Be careful so you don't get stuck in an infinite loop while using this!

# If-Statement

An example if-statement that you may see and use with Karel:

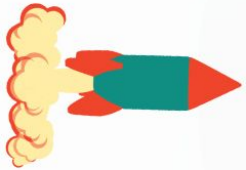
```
def safe_move():
    if front_is_clear():
        move()
```

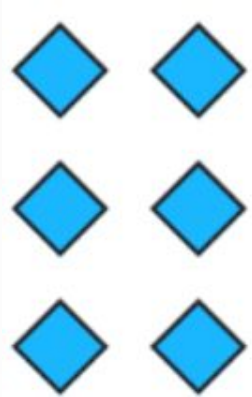
An if-statement runs code inside of it when the associated statement is evaluated to true. We will get into more complex statements later on in the course to add much more flexibility to our if-statements!

# Any Questions?

**HAVE YOU EVER WONDERED IF YOU COULD BUILD A HOSPITAL IN KAREL WHERE THE ROBOT ACTS LIKE A NURSE?**

Hospital

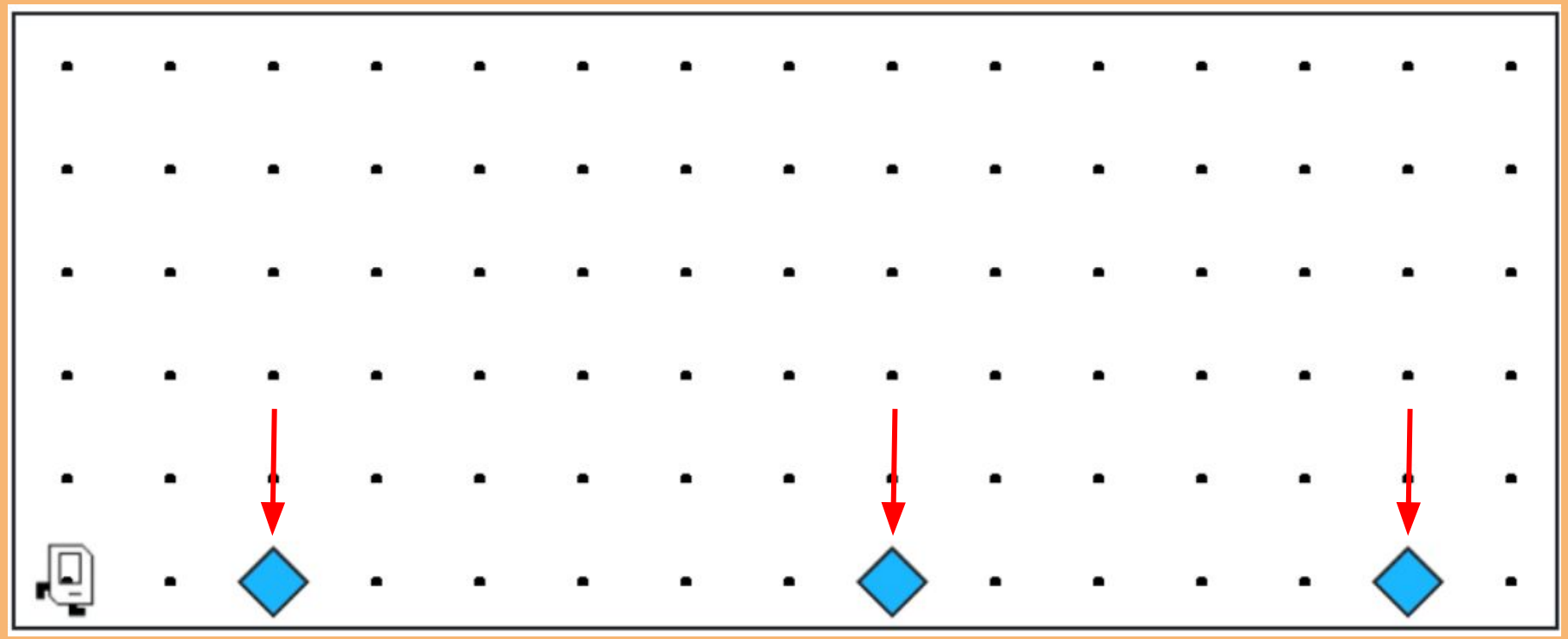




Hospital Karel

# Section Problem: Hospital Karel

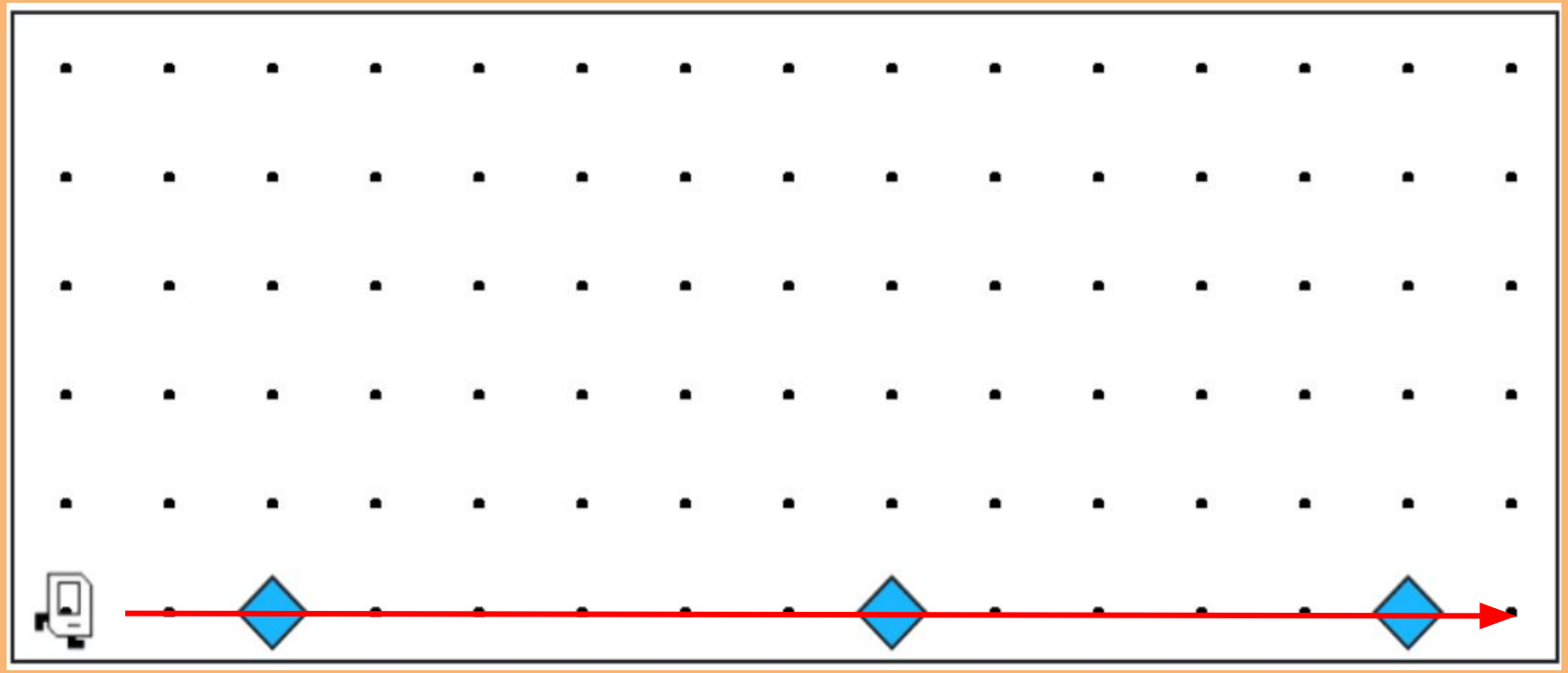
Each beeper in the figure represents a pile of supplies.



# Context

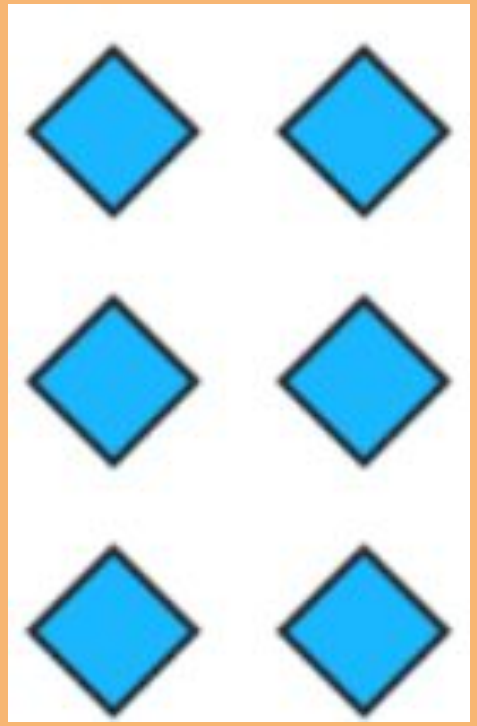
Countries around the world are dispatching hospital-building robots to make sure anyone who gets sick can be treated. They have decided to enlist  Karel robots. Your job is to program those robots.

Karel's job is to walk along the row and build a new hospital in the places marked by each beeper.

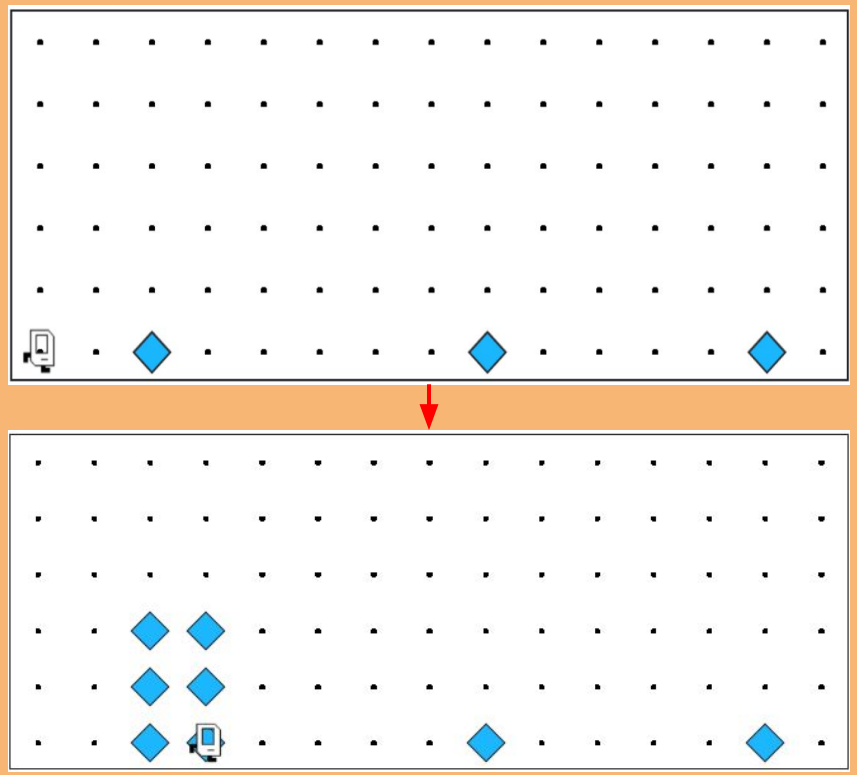


Karel's job is to walk along the row and build a new hospital in the places marked by each beeper.

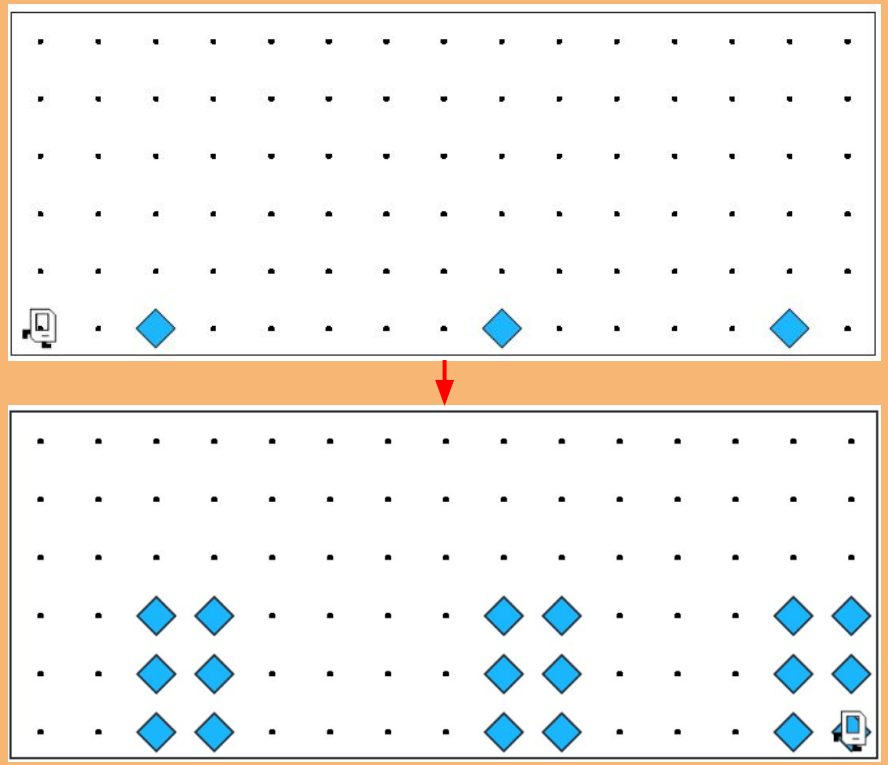
Hospitals look like this: a 3x2 rectangle of beepers!



The new hospital should have their corner at the point at which the pile of supplies was left.



At the end of the run, Karel should be at the end of the row having created a set of hospitals. For the initial conditions shown, the result would look like this:



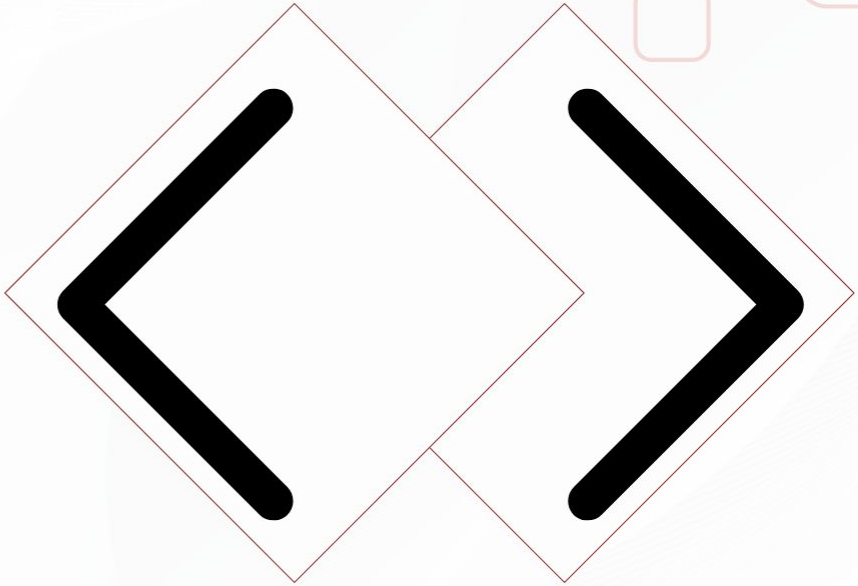
# Notes to Keep in Mind

- Karel starts facing east at (1, 1) with an infinite number of beepers in its beeper bag.
- The beepers indicating the positions at which hospitals should be built will be spaced so that there is room to build the hospitals without overlapping or hitting walls.
- There will be no supplies left on the last column.
- Karel should not run into a wall if it builds a hospital that extends into that final corner.

# Questions Before We Begin?

# Let's get to work!

# THANK YOU



See you in next session