CYPRUS INTERNATIONAL UNIVERSITY

CYPRUS INTERNATIONAL UNIVERSITY
ENGINEERING FACULTY

Lecture 1A

# Otto The Robot

CMPE223 / ISYE223
ALGORITHMS AND PROGRAMMING
2023 – 2024 Spring

# Otto The Robot

- When working with a computer, the knowledge base is *defined* and *limited*.
  - You may not use any other instructions, actions, or formats.
- The rules, called *syntax*, are already defined.

- We will introduce a robot called *Otto The Robot* to better understand this concept.

# Otto The Robot

- Otto has a limited knowledge base, which is defined by 15 actions.
- Otto can sit, stand, add, subtract, and check some of its systems.

You must define your solutions to Otto's problems within the limitations of this knowledge base.

# Assumptions

- Otto always starts from a sitting position with its hands and arms by his side.
- Otto must always end its actions in a sitting position (this means you have to tell Otto to sit down at the end of your solution).

# Simple Actions

- Action 1: Stand up.

  Otto will stand up to an upright position.

- Action 2: Sit down.

  Otto will sit down. You must ensure the chair is behind Otto before giving this instruction.

- Action 3: Take a step.

  Otto will take one step forward (Otto must be standing).

- Action 4: Turn.

  Otto will turn right 90 degrees (Otto must be standing).

# Example

- Write a set of numbered step-by-step instructions to make Otto walk three steps forward and then return and sit on the chair.
- (Use the six problem-solving steps to develop the solution)

# Example (continued)

## 1. Identify The Problem:

Make Otto stand up, walk three steps forward, turn around, walk to the chair, then turn around and sit down.

## 2. Understand The Problem:

Do you understand what Otto is going to do?

Otto's knowledge base is limited to four actions (Stand up, Sit down, Take a step, and Turn).

# Example (continued)

## 3. Alternative Solutions:

a) Stand up, walk three steps forward, turn right 180 degrees, go back to the chair, and sit down.

b) Stand up, walk three steps forward, turn left 180 degrees, go back to the chair, and sit down.

## 4. Select The Best Solution:

Solution (a) is the best since Otto can only turn towards right.

# Example (continued)

## 5. List Instructions:

1. Stand up
2. Take a step
3. Take a step
4. Take a step
5. Turn
6. Turn
7. Take a step
8. Take a step
9. Take a step
10. Turn
11. Turn
12. Sit down

## 6. Evaluate The Solution:

# Arm Actions

- Action 5: Raise arms.

  Raises arms to a forward position perpendicular to its body.

- Action 6: Lower arms.

  Lowers arms to a position at the side of its body.

# Arithmetic Actions

- Otto has one memory location that may be changed by one of the following instructions:
  - Action 7: Add one.

    Adds one to the memory location.

  - Action 8: Subtract one.

    Subtracts one from the memory location.

- Note that this memory location always starts at **zero** when the execution of the instruction set begins.

# Conditional Actions

- Conditional actions give Otto the ability to make a decision (these actions use its memory location and sensory devices at its fingertips).
- Otto has the ability to make three different types of tests and follow one of the two sets of instructions according to the Yes or No result.
- Note that instructions under each branch can be one or more.

# Conditional Actions

- Otto can determine if its hands are touching something.
- Action 9: Test—Are your hands touching anything?

  Yes, then: (one or more actions for “Yes” answer)

  No, then: (one or more actions for “No” answer)

- Otto can determine if its hands are touching a door because the door has a different sensory return than anything else.
- Action 10: Test—Are your hands touching a door?

  Yes, then: (one or more actions for “Yes” answer)

  No, then: (one or more actions for “No” answer)

- Otto can determine when its memory location is zero.
- Action 11: Test—Is the number in your memory zero?

  Yes, then: (one or more actions for “Yes” answer)

  No, then: (one or more actions for “No” answer)

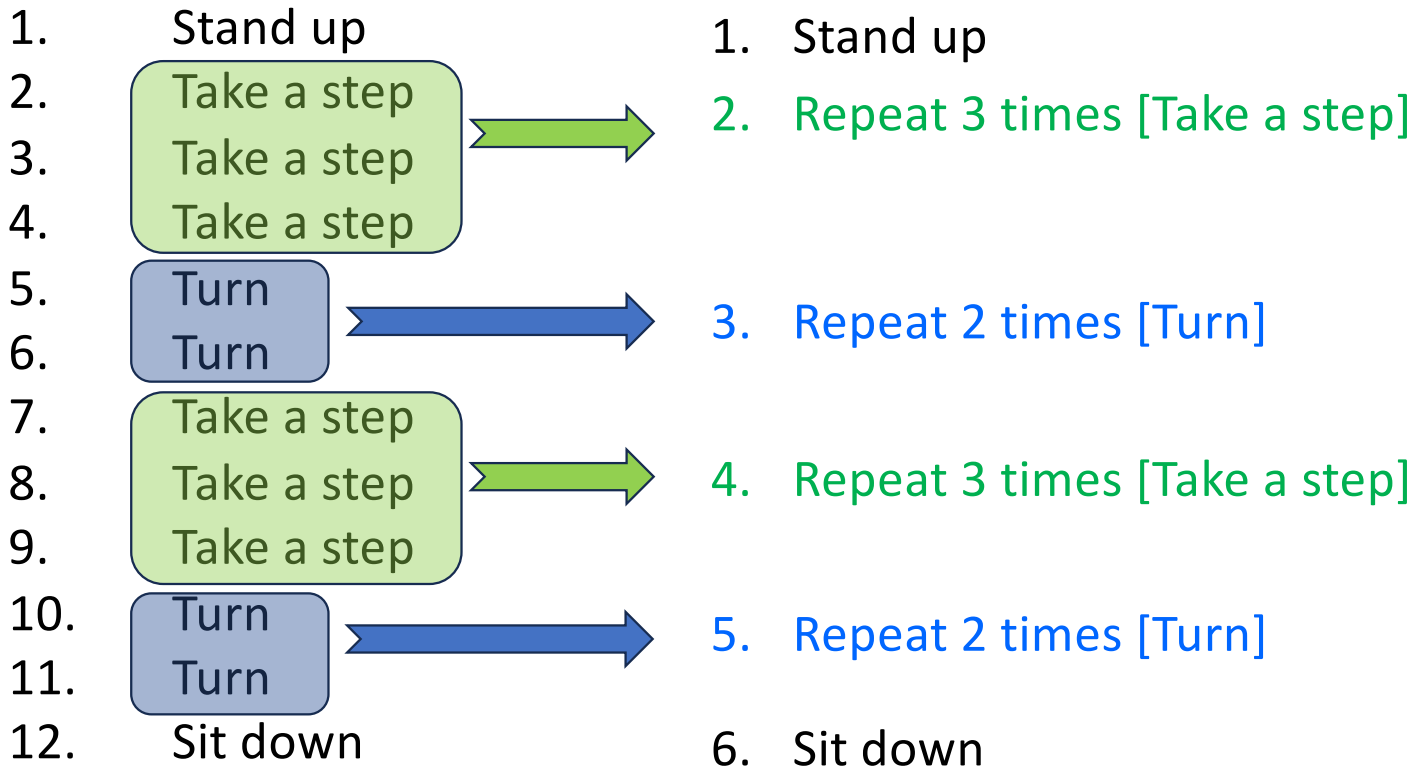
# Repeat Action

- Action 12: Repeat *x* times [set of instructions]

  Repeat instructions inside the brackets *x* number of times (*x* is a known number).

# Example (continued)

- Let's re-write the instructions for the previous example with the new knowledge base.



# What is the Difference?

- We now have two versions of the list of instructions for the same problem.

```
1.  Stand up
2.  Take a step
3.  Take a step
4.  Take a step
5.  Turn
6.  Turn
7.  Take a step
8.  Take a step
9.  Take a step
10. Turn
11. Turn
12. Sit down
```

```
1.  Stand up
2.  Repeat 3 times [Take a step]
3.  Repeat 2 times [Turn]
4.  Repeat 3 times [Take a step]
5.  Repeat 2 times [Turn]
6.  Sit down
```

# What is the Difference?

- The later version has 6 instructions to complete the same task (as compared with the 12 instructions of the previous list.
- From Otto's perspective, nothing has changed.
  - The actual number of actions carried out by Otto is the same.
- From our perspective (the programmer), we have written fewer lines of instructions.

# Transfer Action

- Action 13: Goto *s*.

  Jump to instruction at step *s* and continue to execute instructions in sequential order starting at *s*.

# Other Actions

- Action 14: Open door.

  Opens a door; the door will close after Otto walks through it.

- Action 15: Stop

  Halt execution of instructions.

# Complete The Example

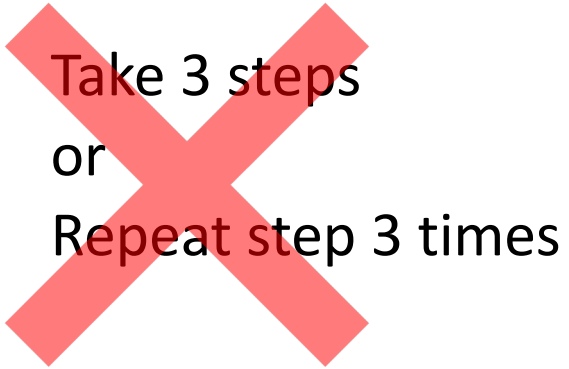
- Since we learned a new instruction (Stop), we must complete the previous example by adding that instruction to the end to make it complete.

```
1.  Stand up
2.  Take a step
3.  Take a step
4.  Take a step
5.  Turn
6.  Turn
7.  Take a step
8.  Take a step
9.  Take a step
10. Turn
11. Turn
12. Sit down
13. Stop
```

```
1.  Stand up
2.  Repeat 3 times [Take a step]
3.  Repeat 2 times [Turn]
4.  Repeat 3 times [Take a step]
5.  Repeat 2 times [Turn]
6.  Sit down
7.  Stop
```

# Otto's Knowledge Base

- There are 15 actions in Otto's knowledge base (We must use ONLY these actions/instructions and no other ones).
- Each instruction must be written in the **proper form** (*syntax*)
- For example:

# OTTO THE ROBOT - INSTRUCTIONS

1. Stand up
2. Sit down
3. Take a step (One step forward – Otto must be standing)
4. Turn (turn right 90 degrees – Otto must be standing)
5. Raise arms (both arms to a forward position perpendicular to its body)
6. Lower arms (both arms down to side position)
7. Add one (adds one to the number in its memory)
8. Subtract one (subtracts one from the number in its memory)
9. Test: Are your hands touching anything?
   - Yes:
   - No:
10. Test: Are your hands touching a door?
    - Yes:
    - No:

# OTTO THE ROBOT – INSTRUCTIONS (continued)

11. Test: Is the number in your memory zero?
    Yes:
    No:
12. Repeat X times [set of instructions] (Repeat instructions inside the brackets X number of times – X is a known number).
13. Goto s (Jump to instruction at steps s – s is a known number - and continue to execute instructions in sequential order starting at s)
14. Open Door (Note that door will close after Otto walks through it)
15. Stop (halt the execution of instructions).

**Assumptions:**

A. Otto will start in a seated position.
B. Otto will end in a seated position in the same chair.

# Exercise 1

Write instructions for Otto the Robot to stand up, go up to the wall, and come back to sit on the initial chair. There are 10 steps to the Wall.

```
1.  Stand up
2.  Repeat 10 times [Take a step]
3.  Repeat 2 times [Turn]
4.  Repeat 10 times [Take a step]
5.  Repeat 2 times [Turn]
6.  Sit down
7.  Stop
```

# Exercise 2

Write necessary instructions for Otto the Robot to stand up, go and find the wall, touch the wall and come back to sit down to the initial chair.

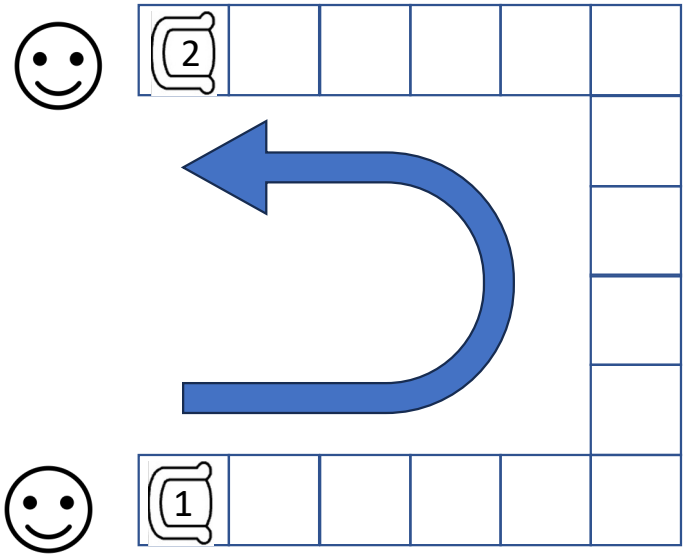
1) Stand up
2) Raise arms
3) Test: Are your hands touching anything?
   Yes: Repeat 2 times [turn]
   No: Take a step
      Goto 3
4) Test: Are your hands touching anything?
   Yes: Repeat 2 times[turn]
   No: Take a step
      Goto 4

5) Lower arms
6) Sit down
7) Stop

# Exercise 3

Write necessary instructions for Otto the Robot to stand up from chair 1 and finish in a seated position in chair 2.

```
1.  Stand up
2.  Repeat 5 times [Take a step]
3.  Repeat 3 times [Turn]
4.  Repeat 5 times [Take a step]
5.  Repeat 3 times [Turn]
6.  Repeat 5 times [Take a step]
7.  Repeat 2 times [Turn]
8.  Sit down
9.  Stop
```



# Exercise 4

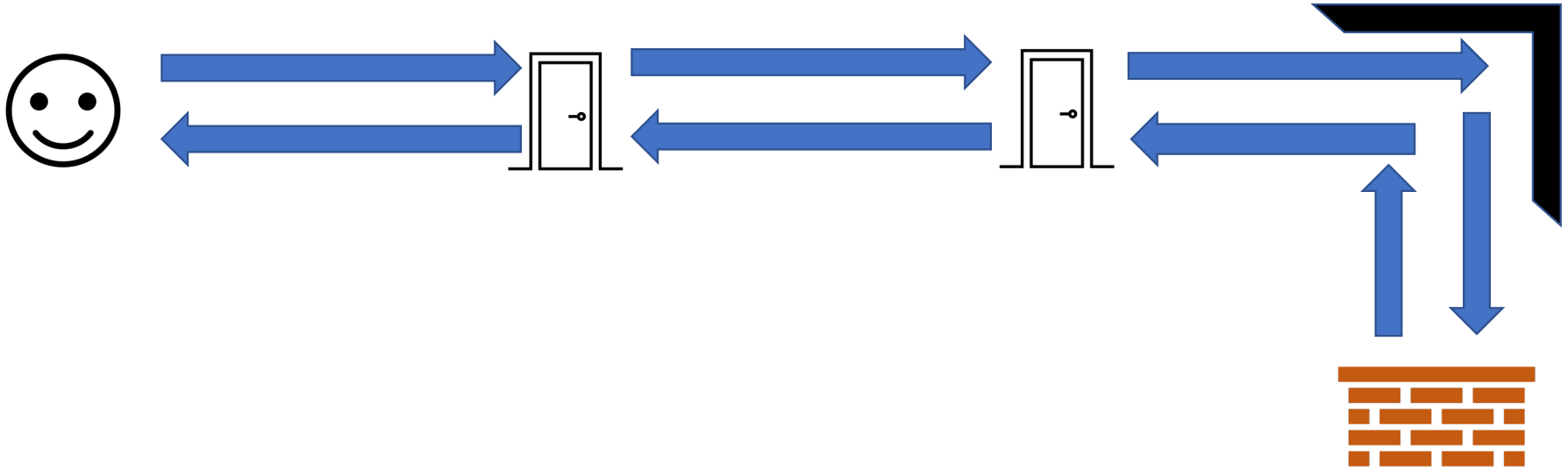
Write necessary instructions for Otto the Robot to count the number of steps to the wall. The robot should stand up, go and find the wall, and come back. Assume that there is at least 1 step between the wall and the Otto.

1) Stand up
2) Raise arms
3) Test: Are your hands touching anything?
   Yes: Repeat 2 times [turn]
   No: Take a step
      Add one
      Goto 3

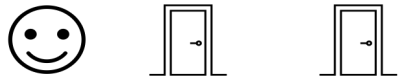
4) Test: Are your hands touching anything?
   Yes: Repeat 2 times[turn]
   No: Take a step
      Goto 4
5) Lower arms
6) Sit down
7) Stop

# Exercise 5

Write the necessary instructions for Otto the Robot to reach the wall and turn back. The robot should stand up, go and pass through 2 doors, turn the corner, find the wall, come back and sit down. Assume that there is at least 1 step between the first door and the Otto.



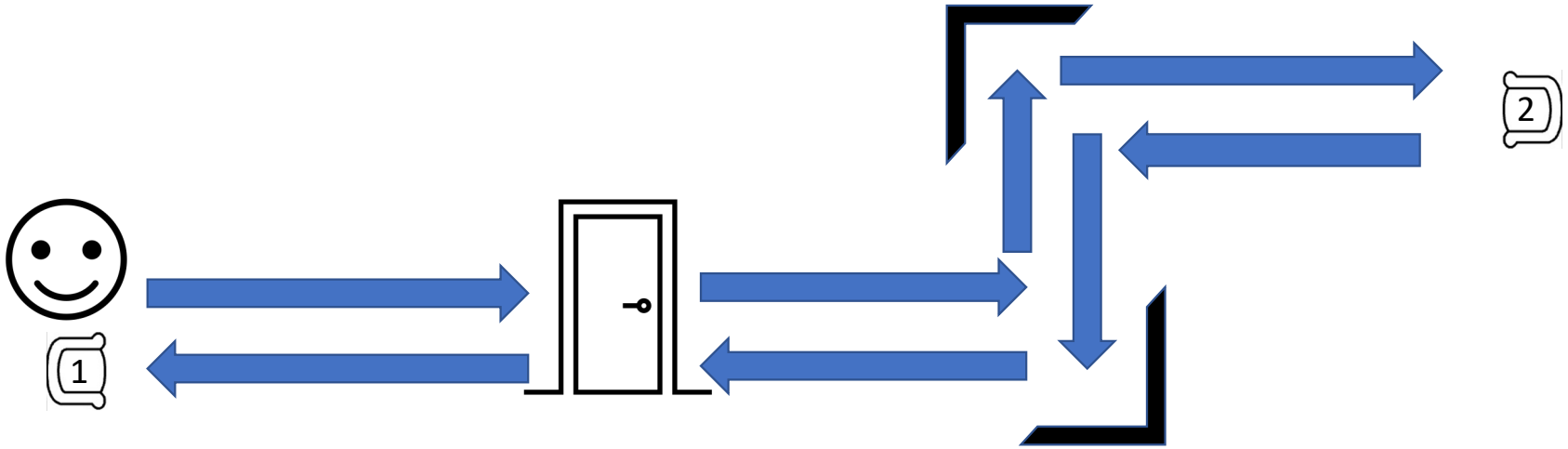
# Exercise 5 (continued)



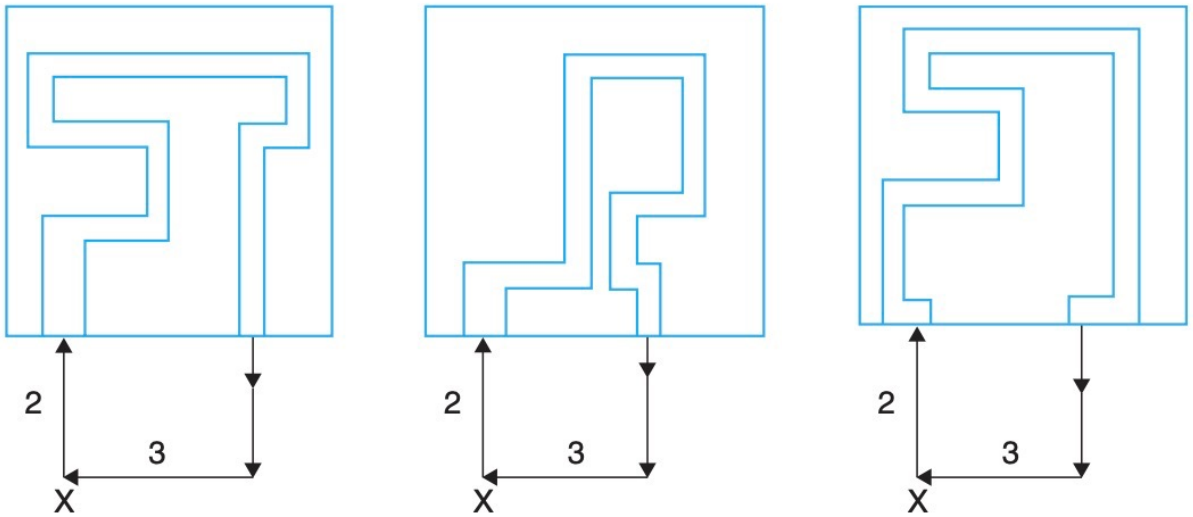
1) Stand up
2) Raise arms
3) Test: Are your hands touching a door?
   Yes: Open door
   No: Take a step
   Goto 3
4) Test: Are your hands touching a door?
   Yes: Open door
   No: Take a step
   Goto 4
5) Test: Are your hands touching anything?
   Yes: Turn
   No: Take a step
   Goto 5
6) Test: Are your hands touching anything?
   Yes: Repeat 2 times [turn]
   No: Take a step
   Goto 6
7) Test: Are your hands touching anything?
   Yes: Repeat 3 times [turn]
   No: Take a step
   Goto 7
8) Test: Are your hands touching a door?
   Yes: Open door
   No: Take a step
   Goto 8
9) Test: Are your hands touching a door?
   Yes: Open door
   No: Take a step
   Goto 9
10) Test: Are your hands touching anything?
   Yes: Repeat 2 times [turn]
   No: Take a step
   Goto 10
11) Lower arms
12) Sit down
13) Stop

# Problem Set 01A

P1. Write instructions for Otto The Robot to find chair 2 and sit down, get up and return to chair 1 using the following map.



P2. Write one set of instructions that will direct Otto through any maze with its entrance and exit on the same side. The maze has no dead ends and no intersections. From Otto's starting position, the entrance is two steps forward. When Otto is facing the entrance, the exit is three steps to its right. The figures below show three example mazes for you to test your solution. Do not make separate solutions for each maze. Your solution should allow Otto to walk through any maze.



# References

Sprankle, M., & Hubbard, J. (2008). *Problem solving and programming concepts*. Prentice Hall Press.