Introduction to Lego Kits: Building an Embodied AI Mobile Platform

Xiaofeng Xiong

xizi@mmmi.sdu.dk
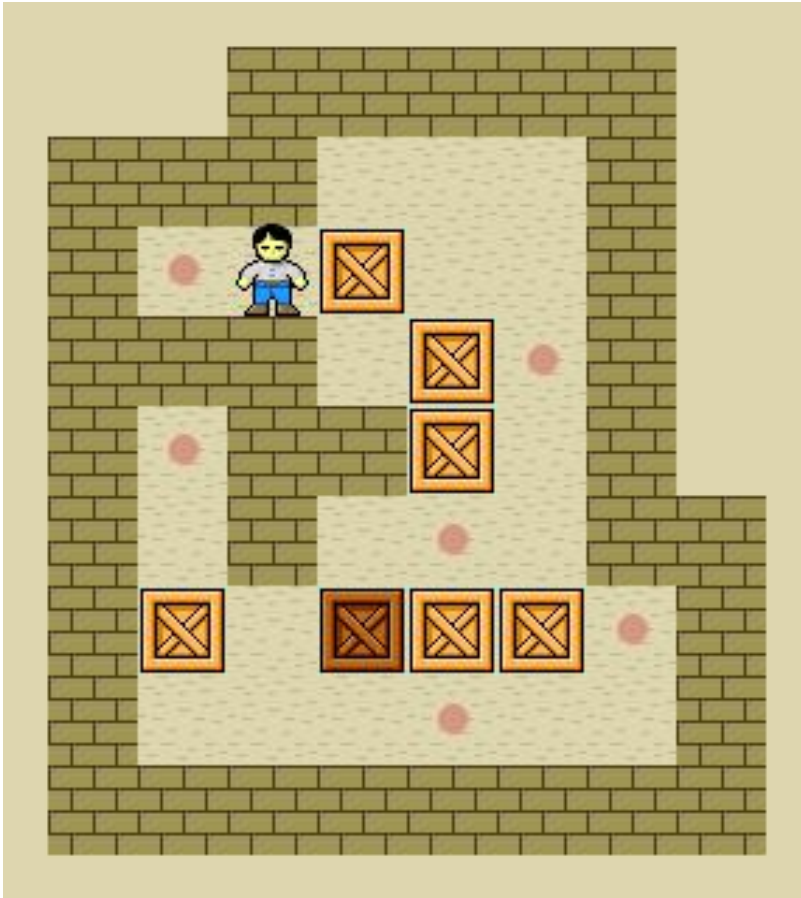
Embodied Systems for Robotics and Learning,

University of Southern Denmark (SDU), Denmark

09/09/2019

Lecture outline

❑ Sokoban Game

❑ Sokoban Competition

❑ Lego Mindstorm EV3 Kits

❑ Motor and Sensor Experiments

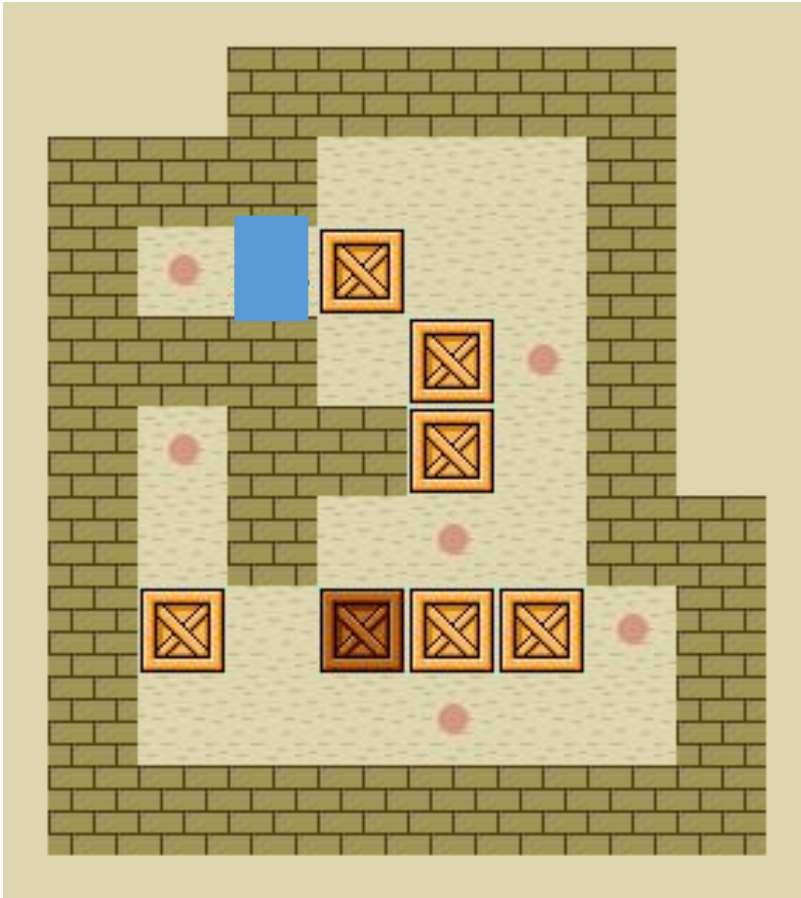❑ Advices and Supports

Sokoban Game

Wall Target Box

- Task: Push boxes to the target positions in a warehouse

- Rules: NOT be pushed into other boxes or walls, and they cannot be pulled.

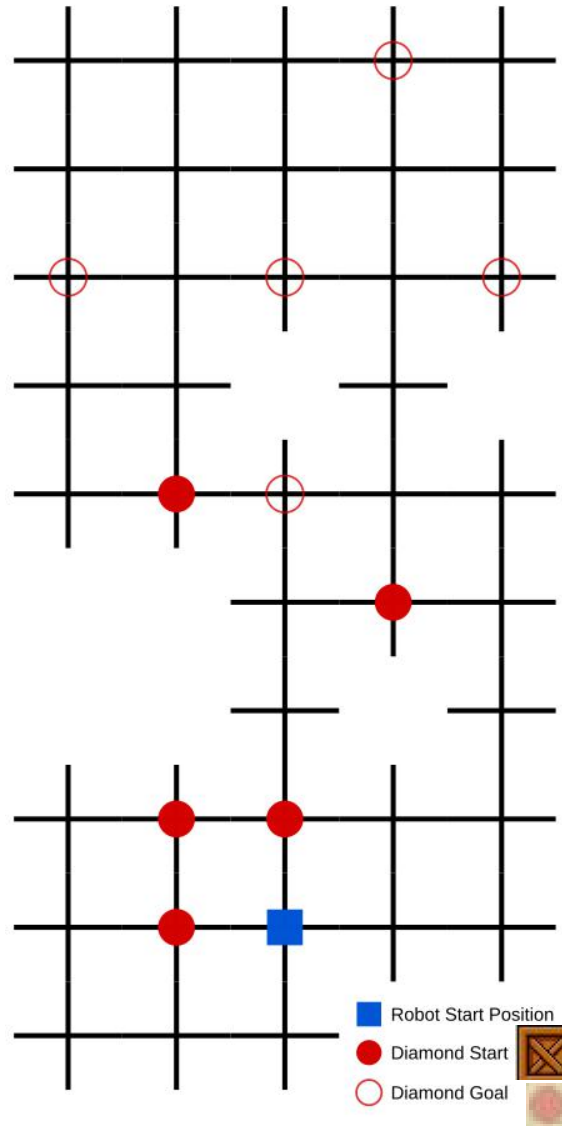- Solved: all boxes are at the target positions

Hiroyuki Imabayashi, 1981

Credits to Wiki images

Sokoban Competition
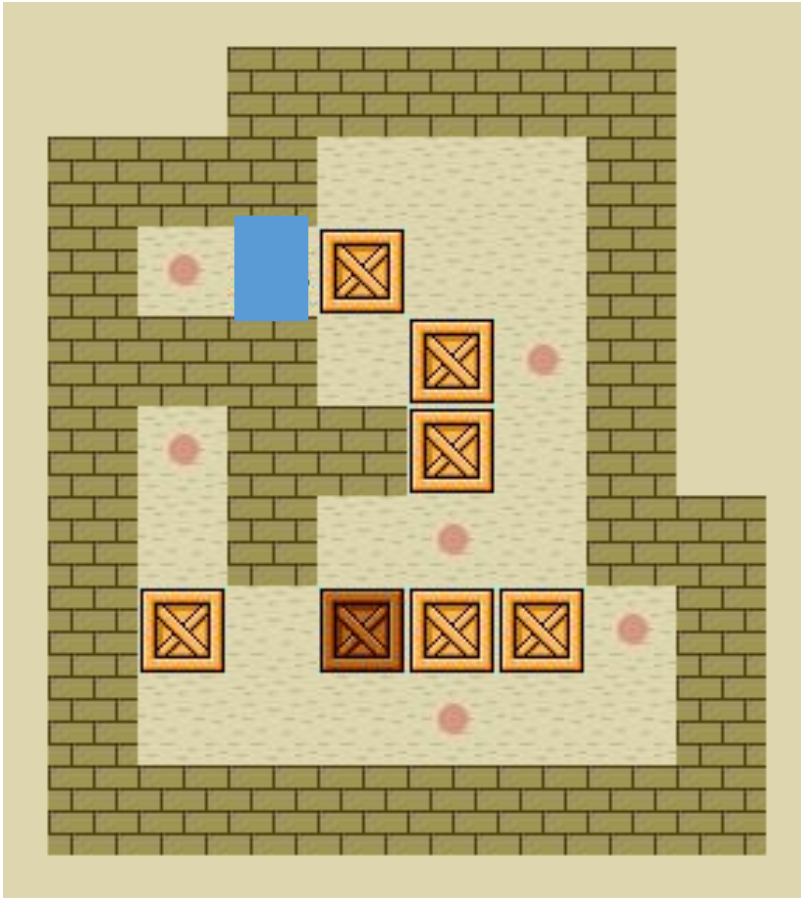
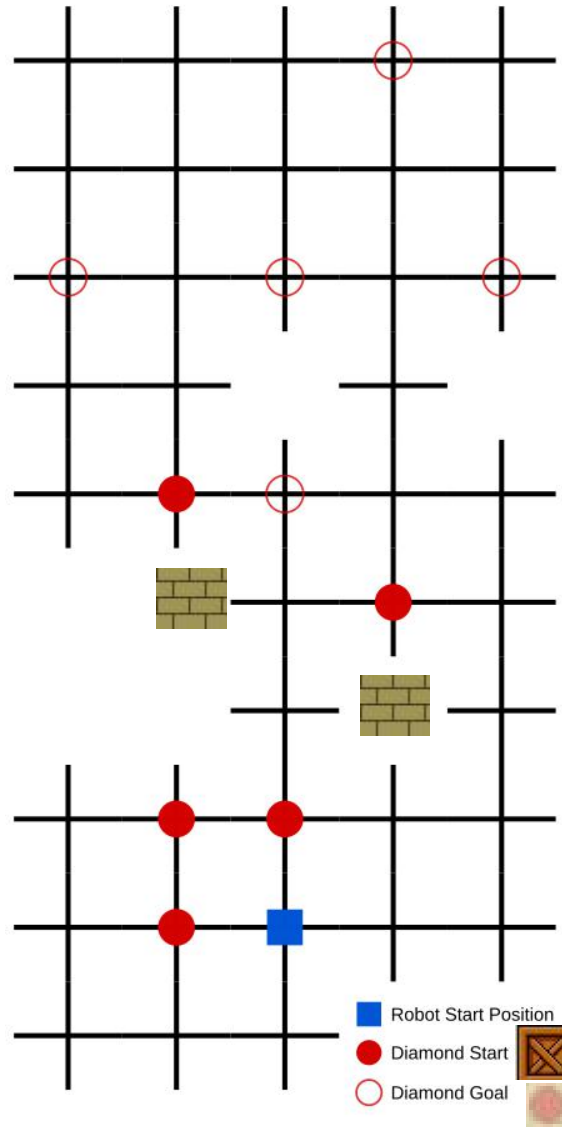**Wall**   **Target**   **Box**

Hiroyuki Imabayashi, 1981

- The surface is white paper with black lines
- placed in the horizontal plane on the floor

Robot Start Position
Diamond Start
Diamond Goal

Sokoban competition map

Sokoban Competition

Wall **Target** **Box**

Hiroyuki Imabayashi, 1981

- The lines' color will be black
- The width of the lines is 14mm±2mm
- The walls are represented by ending the black lines

■ Robot Start Position
● Diamond Start
○ Diamond Goal

Sokoban competition map

Sokoban Competition

Wall
Target
Box

D:54 mm; H: 72 mm

Hiroyuki Imabayashi, 1981

■ Robot Start Position
● Diamond Start
○ Diamond Goal
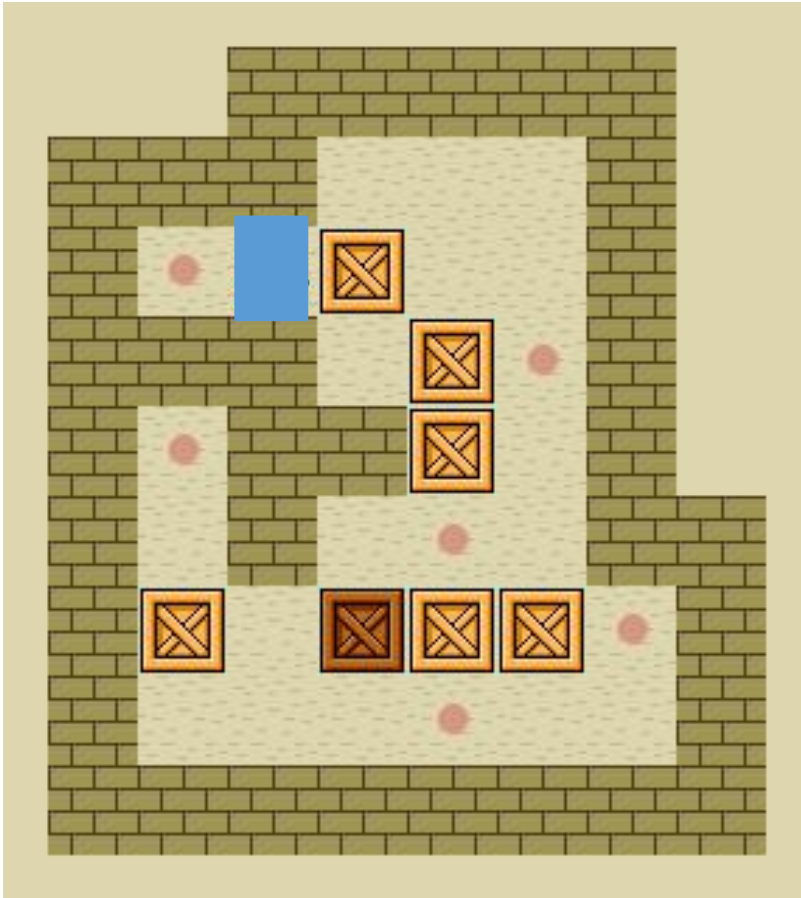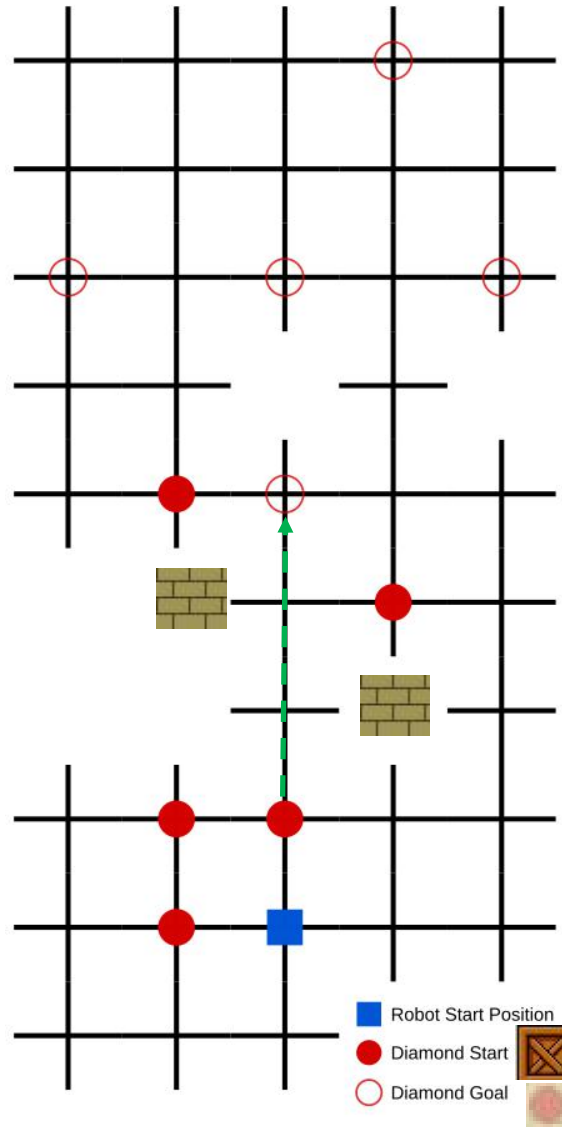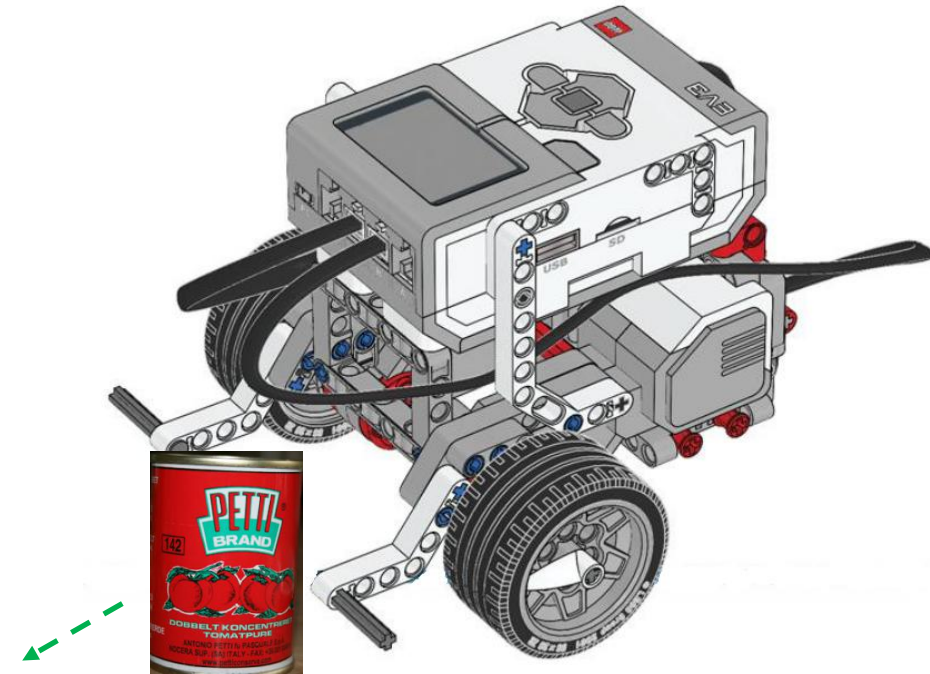
Sokoban competition map

Sokoban Competition

Assemble kits to form a mobile robot

Hand-code programme to read sensors and control motor wheel(s)

Decision making by the programme in terms of the map

Sokoban competition map

Lego Mindstorm EV3 Kits

Large Motor
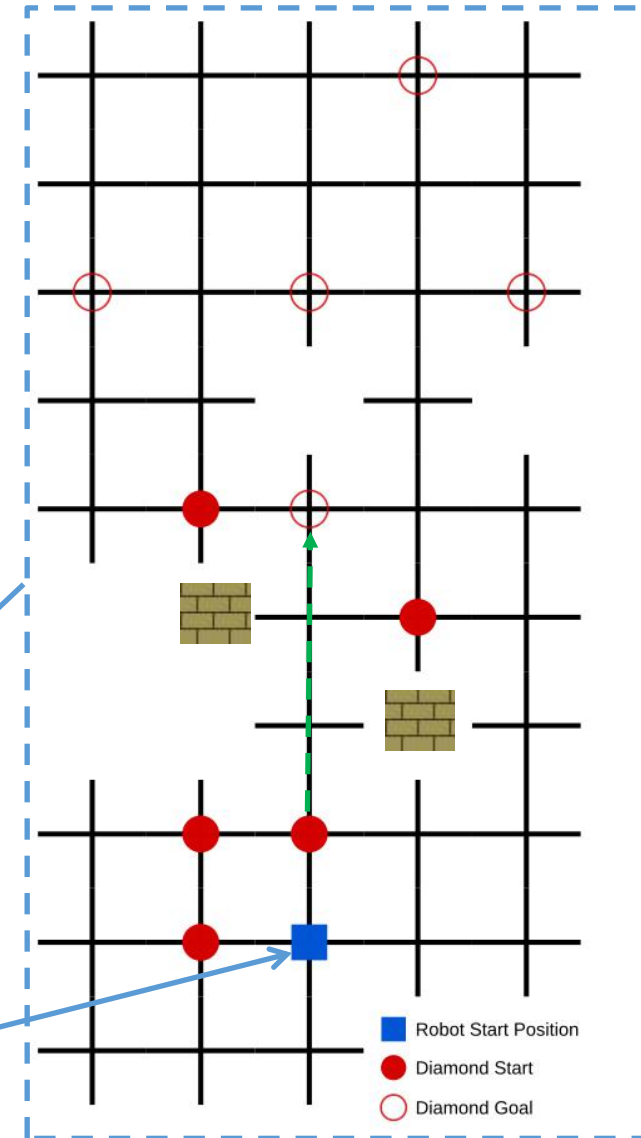+ Lets you program precise and powerful robotic action.

Medium Motor
+ Maintains precision, while trading some power for compact size and faster response.

EV3 Brick
+ Serves as the control center and power station for your robot.

Ultrasonic Sensor
+ Uses reflected sound waves to measure distance between the sensor and any objects in its path.

Color Sensor
+ Recognizes seven different colors and measures light intensity.

Gyro Sensor
+ Measures how fast and how far your robot is turning.

Rechargeable Battery
+ Economical, environmentally friendly, and convenient power source for your robot.

Touch Sensor
+ Recognizes three conditions—touched, bumped, and released.

Lego Mindstorm EV3 Kits

Large Motor
+ Lets you program precise and powerful robotic action.

Medium Motor
+ Maintains precision, while trading some power for compact size and faster response.

EV3 Brick
+ Serves as the control center and power station for your robot.

Ultrasonic Sensor
+ Uses reflected sound waves to measure distance between

Input Ports
Input Ports 1, 2, 3, and 4 are used to connect sensors to the EV3 Brick.

Rechargeable Battery
+ Economical, environmentally friendly, and convenient power source for your robot.

Touch Sensor
+ Recognizes three conditions—touched, bumped, and released.

Gyro Sensor
+ Measures how fast and how far your robot is turning.

Lego Mindstorm EV3 Kits

Large Motor
+ Lets you program precise and powerful robotic action.

Medium Motor
+ Maintains precision, while trading some power for compact size and faster response.

EV3 Brick
+ Serves as the control center and power station for your robot.

Ultrasonic Sensor
+ Uses reflected sound waves to measure distance between the sensor and any objects in its path.

PC Port
The Mini-USB PC Port, located next to the D port, is used to connect the EV3 Brick to a computer.

Output Ports
Output Ports A, B, C, and D are used to connect motors to the EV3 Brick.

ensor
nizes seven different and measures light ty.

ow far

Rechargeable Battery
+ Economical, environmentally friendly, and convenient power source for your robot.

Lego Mindstorm EV3 Kits

Connect EV3 (e.g., ssh)

Our modified image can be found at https://esrl.dk/ai1_ev3_sdcard.img

```
xiong@xiong-HP-EliteBook-840-G4:~$ ssh ai1@192.168.0.1
ai1@192.168.0.1's password:

        ___     ____     _____     ____   _____  _    _
       / _ \   |  _ \   |___ /    |  _ \ |  ___|| |  | |
      |  __/   | |_) |   |_ \     | | | || |__  | |  | |
      |____|   |____/   |___/    |____/ |____| |_|  |_|

Debian jessie on LEGO MINDSTORMS EV3!

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 25 22:37:40 2018 from 192.168.0.2
ai1@ev3dev:~$
```
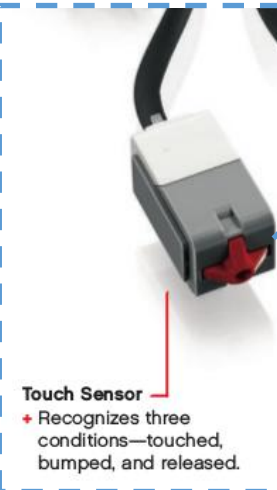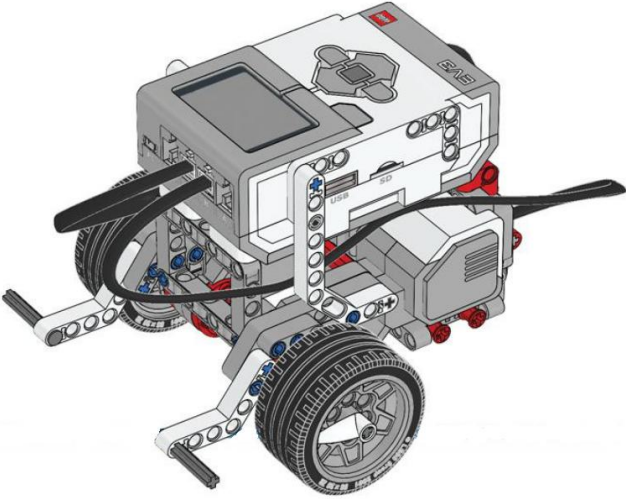
Password: ai1rockz

Copy the programme to EV3 (e.g., scp)

```
Connection to 192.168.0.1 closed.
xiong@xiong-HP-EliteBook-840-G4:~$ scp ~/courses/lego_ev3/tutorials/codes/exampl
e.py ai1@192.168.0.1:
ai1@192.168.0.1's password:
example.py                              100% 1137     1.1KB/s   00:00
xiong@xiong-HP-EliteBook-840-G4:~$
```

Password: ai1rockz

USB cable
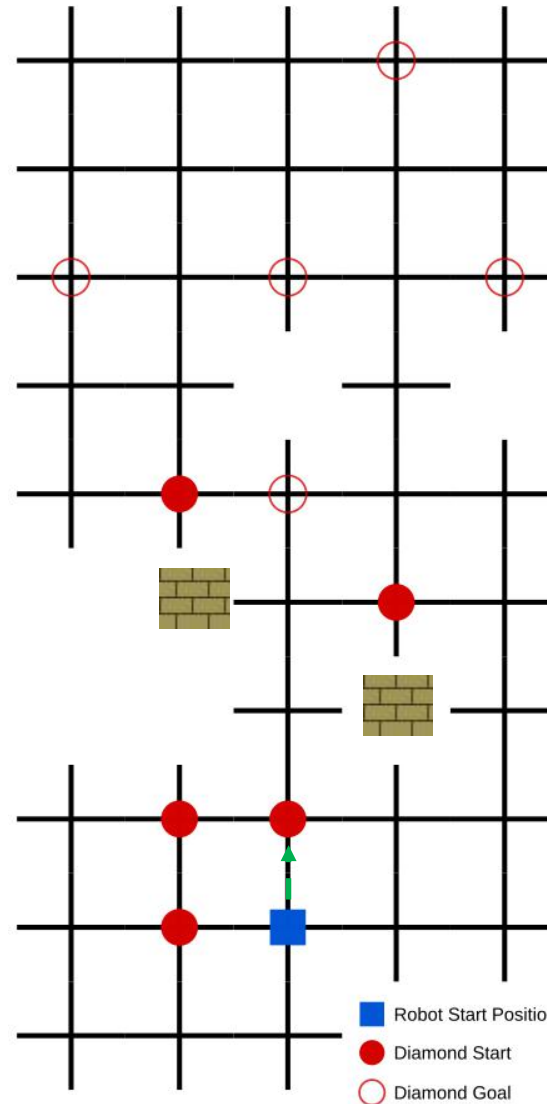
11

Motor and Sensor Experiments

Obtain a can?

Sokoban competition map

■ Robot Start Position
● Diamond Start
○ Diamond Goal

Motor and Sensor Experiments

Two motor wheels and one touch sensor

Large Motor
+ Lets you program precise and powerful robotic action.

Touch Sensor
+ Recognizes three conditions—touched, bumped, and released.

```python
#!/usr/bin/python3.4

import ev3dev.ev3 as ev3
from time import sleep

import signal


btn = ev3.Button()

mA = ev3.LargeMotor('outA')
mB = ev3.LargeMotor('outB')

THRESHOLD_LEFT = 30
THRESHOLD_RIGHT = 350

BASE_SPEED = 30
TURN_SPEED = 80

#lightSensorLeft = ev3.ColorSensor('in1')
#lightSensorRight = ev3.LightSensor('in2')

TouchSensor = ev3.TouchSensor('in3')

#assert lightSensorLeft.connected, "LightSensorLeft(ColorSensor) is not connected"
#assert lightSensorRight.connected, "LightSensorRight(LightSensor) is not conected"

assert TouchSensor.connected, "Touch sensor is not connected"

mB.run_direct()
mA.run_direct()
```
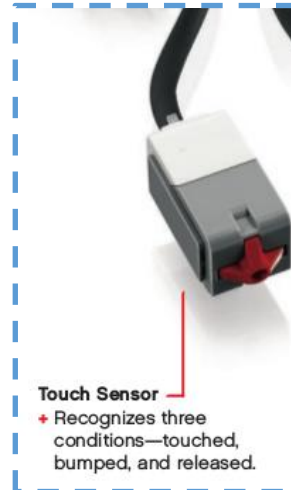
example_touch.py

Motor and Sensor Experiments

Two motor wheels and one touch sensor

Large Motor
+ Lets you program precise and powerful robotic action.

Touch Sensor
+ Recognizes three conditions—touched, bumped, and released.
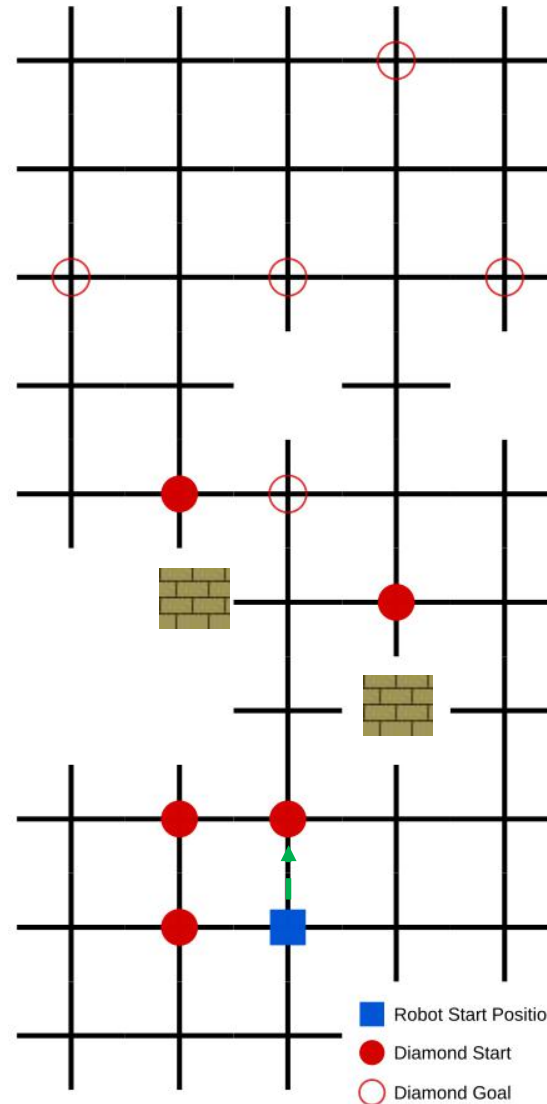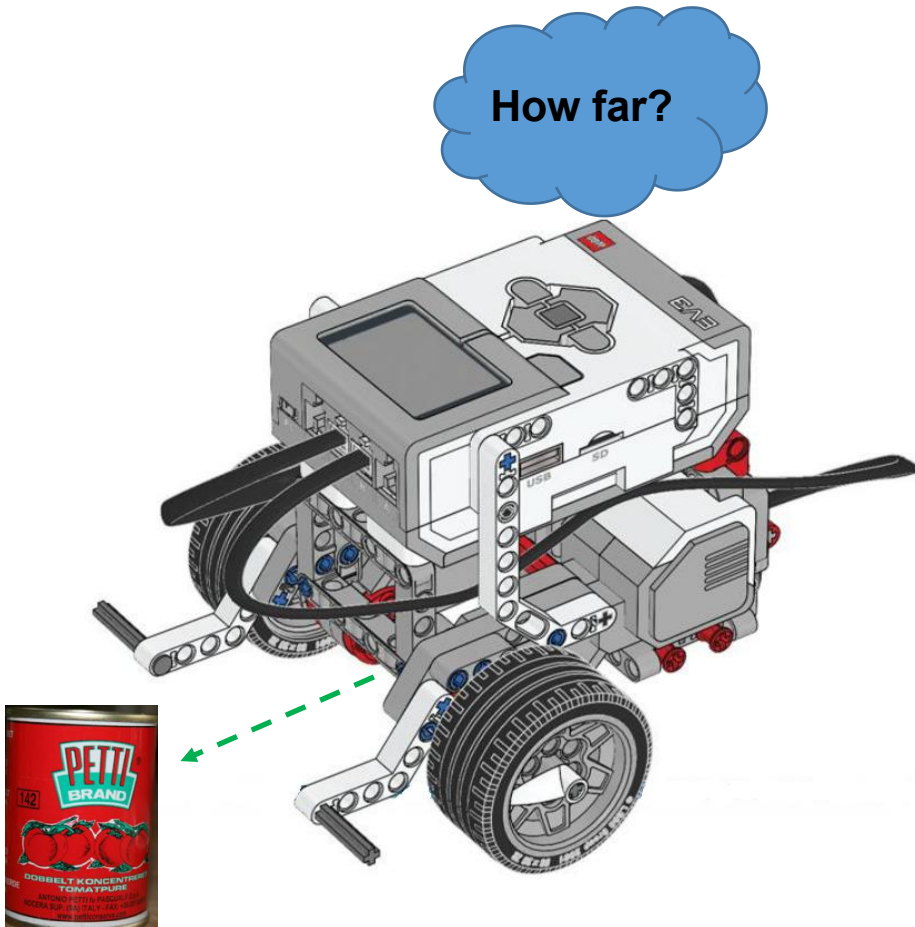
```python
while True:
    mA.duty_cycle_sp = BASE_SPEED
    mB.duty_cycle_sp = BASE_SPEED
    tou_val = TouchSensor.value()

    if btn.any():
        ev3.Sound.beep().wait()
        mA.duty_cycle_sp = 0
        mB.duty_cycle_sp = 0
        exit()
    else:
        print("Touch sensor value: ", tou_val)
```
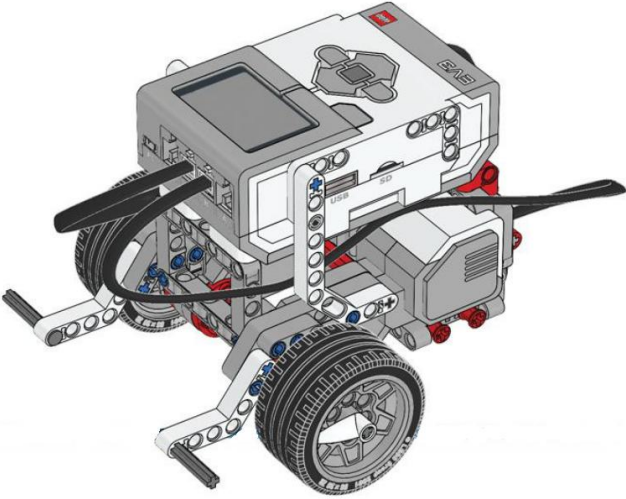
example_touch.py

Motor and Sensor Experiments

How far?

Sokoban competition map

■ Robot Start Position
● Diamond Start
○ Diamond Goal

Motor and Sensor Experiments

Two motor wheels and one ultrasonic sensor

Large Motor
+ Lets you program precise and powerful robotic action.

Ultrasonic Sensor
+ Uses reflected sound waves to measure distance between the sensor and any objects in its path.

```python
#!/usr/bin/python3.4

import ev3dev.ev3 as ev3
from time import sleep

import signal


btn = ev3.Button()

mA = ev3.LargeMotor('outA')
mB = ev3.LargeMotor('outB')

THRESHOLD_LEFT = 30
THRESHOLD_RIGHT = 350

BASE_SPEED = 30
TURN_SPEED = 80

us = ev3.UltrasonicSensor('in4')
#lightSensorRight = ev3.LightSensor('in2')

TouchSensor = ev3.TouchSensor('in3')

us.mode = 'US-DIST-CM'

assert us.connected, "Ultrasonic sensor is not connected"
#assert lightSensorRight.connected, "LightSensorRight(LightSensor) is not conected"

assert TouchSensor.connected, "Touch sensor is not connected"
```
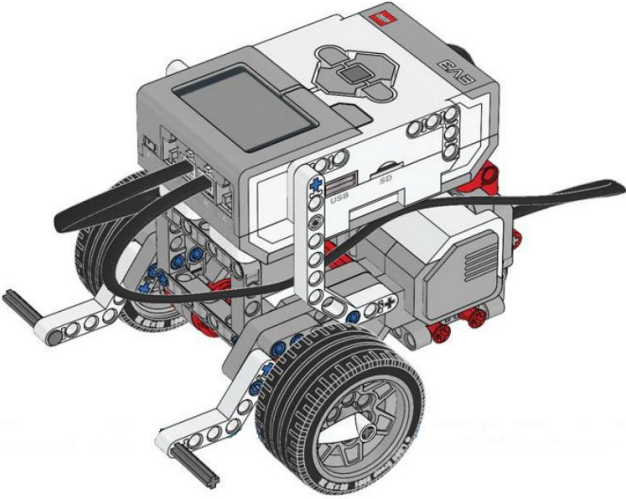
example_us.py

Motor and Sensor Experiments

Large Motor
+ Lets you program precise and powerful robotic action.

Two motor wheels and one ultrasonic sensor

Ultrasonic Sensor
+ Uses reflected sound waves to measure distance between the sensor and any objects in its path.

```python
while True:
    mA.duty_cycle_sp = BASE_SPEED
    mB.duty_cycle_sp = BASE_SPEED
    tou_val = TouchSensor.value()

    if tou_val == 1:
        ev3.Sound.beep().wait()
        mA.duty_cycle_sp = 0
        mB.duty_cycle_sp = 0
        exit()
    else:
        dis = us.value()/10
        print(str(dis) + " " + unit)
```
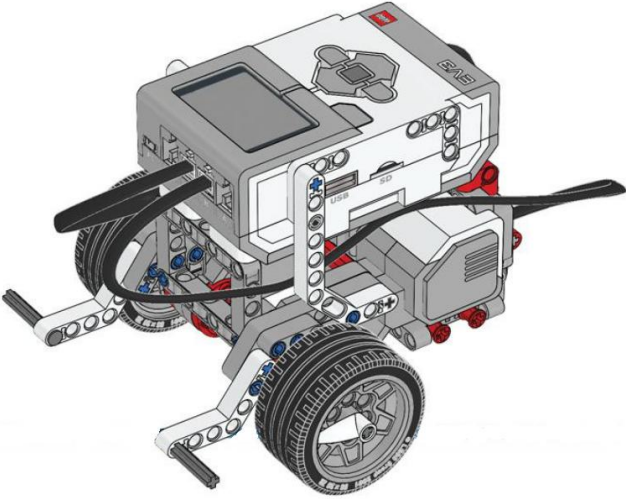
example_us.py

Motor and Sensor Experiments

Arrive at the target?
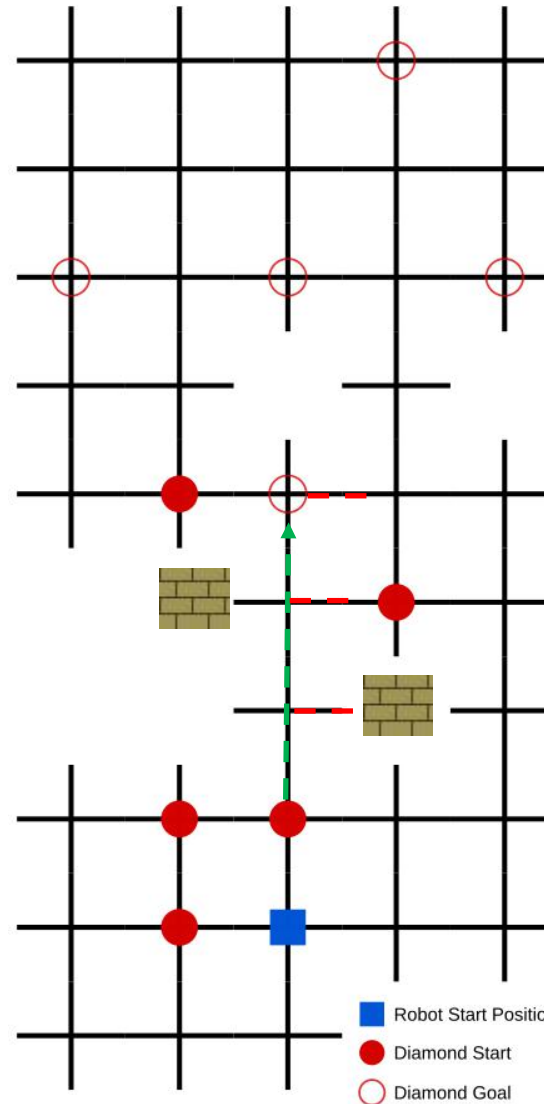
Detect and count the horizontal lines

Robot Start Position
Diamond Start
Diamond Goal

Sokoban competition map

Motor and Sensor Experiments

Two motor wheels and one color sensor

```python
import ev3dev.ev3 as ev3
from time import sleep

import signal


btn = ev3.Button()

mA = ev3.LargeMotor('outA')
mB = ev3.LargeMotor('outB')

THRESHOLD_LEFT = 30
THRESHOLD_RIGHT = 350

BASE_SPEED = 30
TURN_SPEED = 80

cl = ev3.ColorSensor('in1')
#lightSensorRight = ev3.LightSensor('in2')

TouchSensor = ev3.TouchSensor('in3')

cl.mode = 'COL-COLOR'

assert cl.connected, "LightSensorLeft(ColorSensor) is not connected"
#assert lightSensorRight.connected, "LightSensorRight(LightSensor) is not conected"

assert TouchSensor.connected, "Touch sensor is not connected"

colors = ('unknown','black','blue','green', 'yellow', 'red', 'white', 'brown')
```
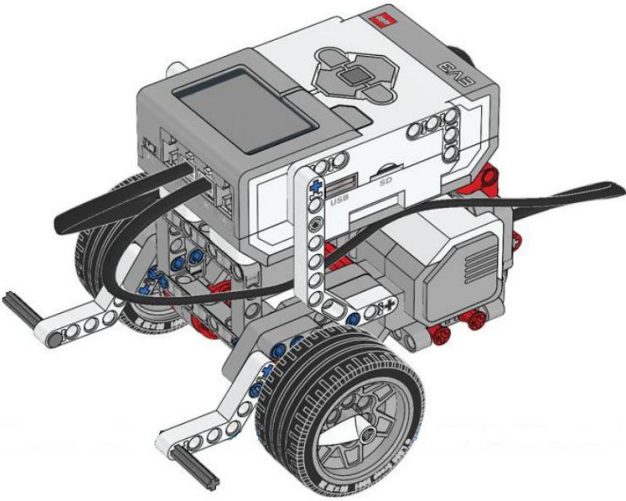
example_color.py

Motor and Sensor Experiments

Two motor wheels and one color sensor

Large Motor
+ Lets you program precise and powerful robotic action.

Color Sensor
+ Recognizes seven different colors and measures light intensity.
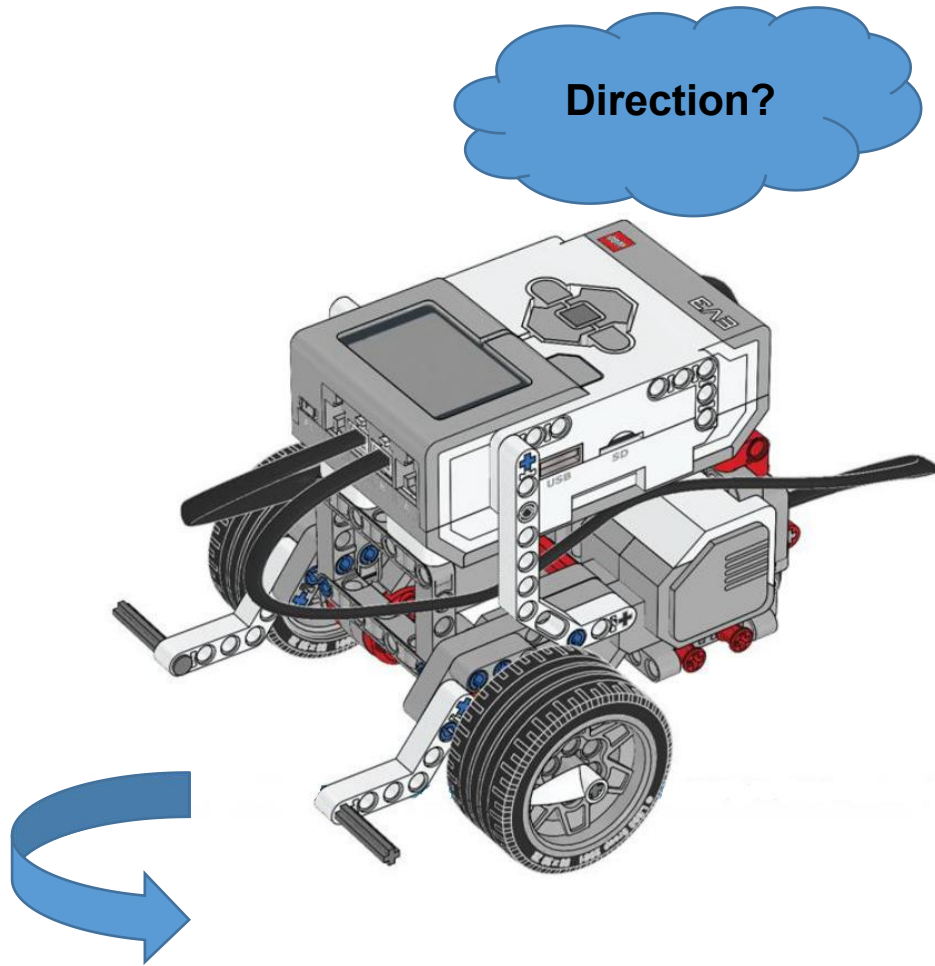
```python
while True:
    mA.duty_cycle_sp = BASE_SPEED
    mB.duty_cycle_sp = BASE_SPEED
    tou_val = TouchSensor.value()

    if tou_val == 1:
        ev3.Sound.beep().wait()
        mA.duty_cycle_sp = 0
        mB.duty_cycle_sp = 0
        exit()
    else:

        print(colors[cl.value()])
```
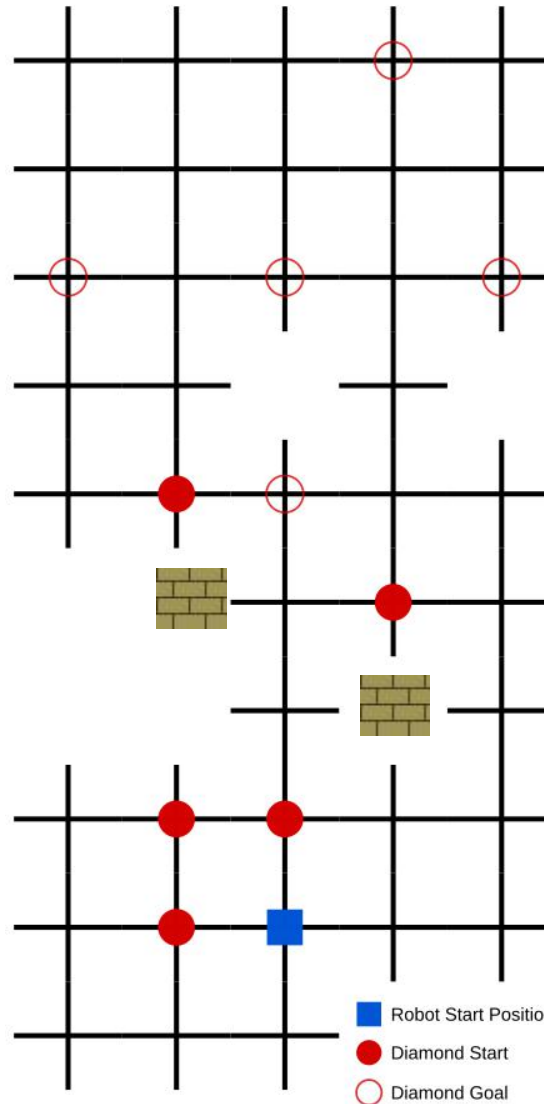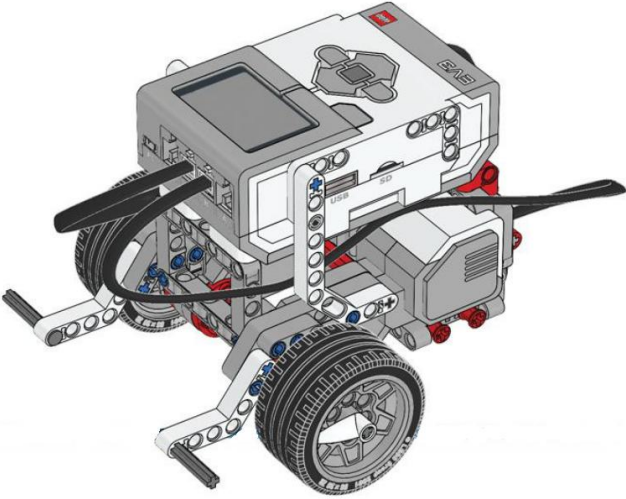
example_color.py
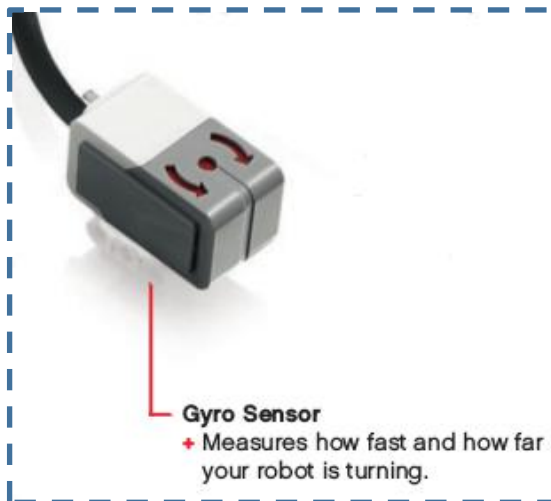
Motor and Sensor Experiments

Direction?

Turning

Sokoban competition map

■ Robot Start Position
● Diamond Start
○ Diamond Goal

Motor and Sensor Experiments

Two motor wheels and one gyro sensor

Large Motor
+ Lets you program precise and powerful robotic action.

Gyro Sensor
+ Measures how fast and how far your robot is turning.

```python
#!/usr/bin/python3.4

import ev3dev.ev3 as ev3
from time import sleep

import signal


btn = ev3.Button()

mA = ev3.LargeMotor('outA')
mB = ev3.LargeMotor('outB')

THRESHOLD_LEFT = 30
THRESHOLD_RIGHT = 350

BASE_SPEED = 30
TURN_SPEED = 80

gy = ev3.GyroSensor('in2')
#lightSensorRight = ev3.LightSensor('in2')

TouchSensor = ev3.TouchSensor('in3')

gy.mode = 'GYRO-ANG'

assert gy.connected, "Gyro sensor is not connected"
#assert lightSensorRight.connected, "LightSensorRight(LightSensor) is not conected"

assert TouchSensor.connected, "Touch sensor is not connected"
```
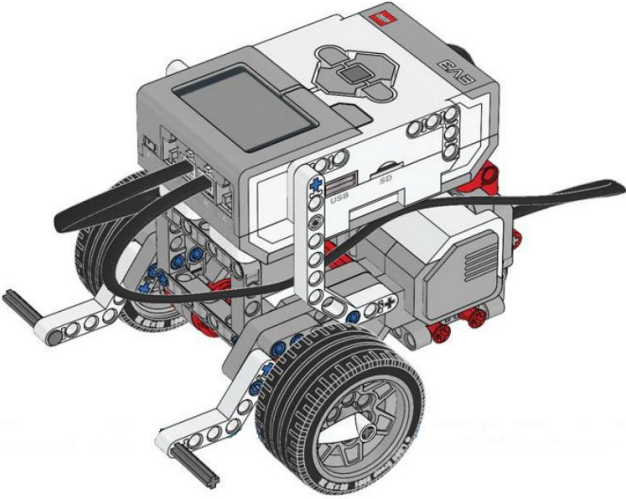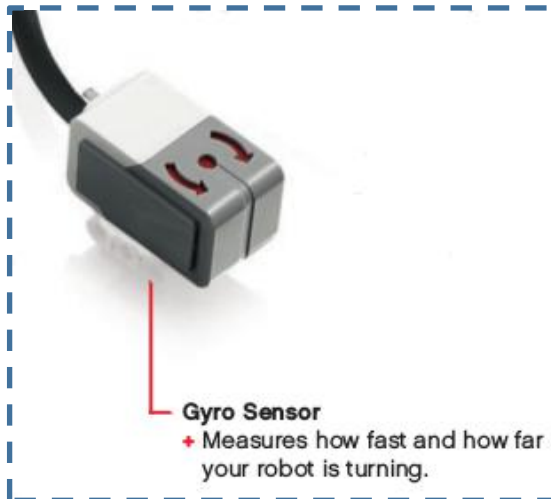
example_gy.py

Motor and Sensor Experiments

Two motor wheels and one gyro sensor

Large Motor
+ Lets you program precise and powerful robotic action.

Gyro Sensor
+ Measures how fast and how far your robot is turning.

```python
while True:
    mA.duty_cycle_sp = BASE_SPEED
    mB.duty_cycle_sp = BASE_SPEED
    tou_val = TouchSensor.value()

    if tou_val == 1:
        ev3.Sound.beep().wait()
        mA.duty_cycle_sp = 0
        mB.duty_cycle_sp = 0
        exit()
    else:
        ang = gy.value()
        print(str(ang) + " " + unit)
```

example_gy.py

Advices and Supports

- Use minimalistic kits (e.g., sensors) to win the competition
- Install an OS, e.g., Ubuntu
- Install a programming language, e.g., Python3. (see options)
- Ev3python
- Python for beginners
- Work at home or in the Mærsk building's lobby
- An earlier competition map will be available in the Mærsk building's lobby for testing purposes
- Utilize Google and YouTube
- **9th of Oct. 12:00** Preliminary report due, as PDF, by emails to john@mmmi.sdu.dk and xizi@mmmi.sdu.dk
- **22nd of Nov.** releasing the Sokoban Competition map
- **2nd of Dec.** Sokoban Competition, up to three attempts per group but no more than 15 min
- **18th of Dec. 12:00** Final report due, as PDF, by email to john@mmmi.sdu.dk

Contacts

- Mathias Neerup, mane@mmmi.sdu.dk
- Christian Quist Nielsen, chni215@student.sdu.dk
- Oliver Klinggaard, olkli15@student.sdu.dk

Questions