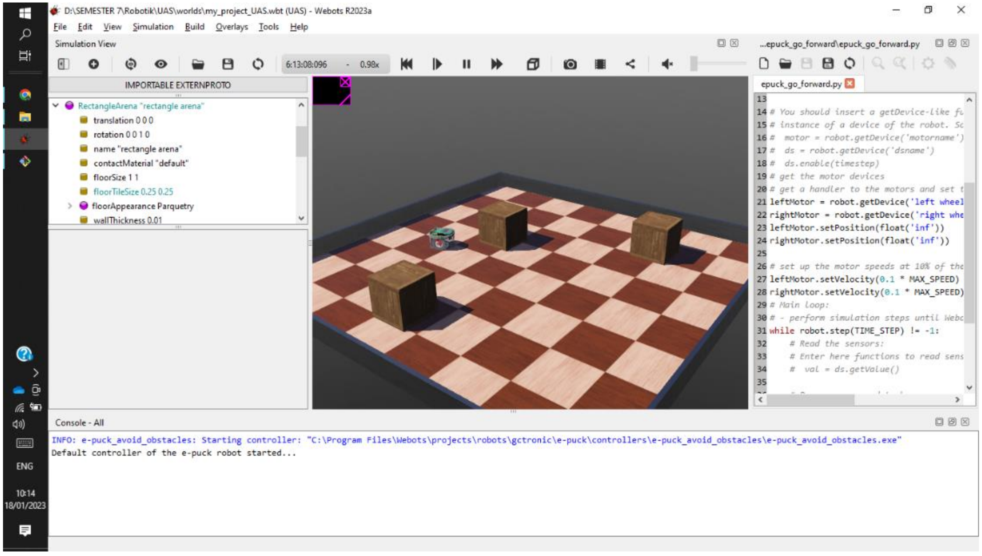
DOCUMENTATION WEBOT TUTORIALS & MICROMOUSE

NAMA:AGUSTIO

NIM:1103194102

1. Tutorial 1: Your first Simulation in Webots

- create robot objects and boxes and can move them
- create first simulation containing a simple environment: an arena with floor and walls, a few boxes, an e-puck robot and a controller program that will make the robot move.


D:\SEMESTER 7\Robotik\UAS\worlds\my_project_UAS.wbt (UAS) - Webots R2023a
File Edit View Simulation Build Overlays Tools Help
Simulation View
6:13:08:096 - 0.98x
IMPORTABLE EXTERNPROTO
...epuck_go_forward\epuck_go_forward.py
epuck_go_forward.py
Console - All
ENG
10:14
18/01/2023

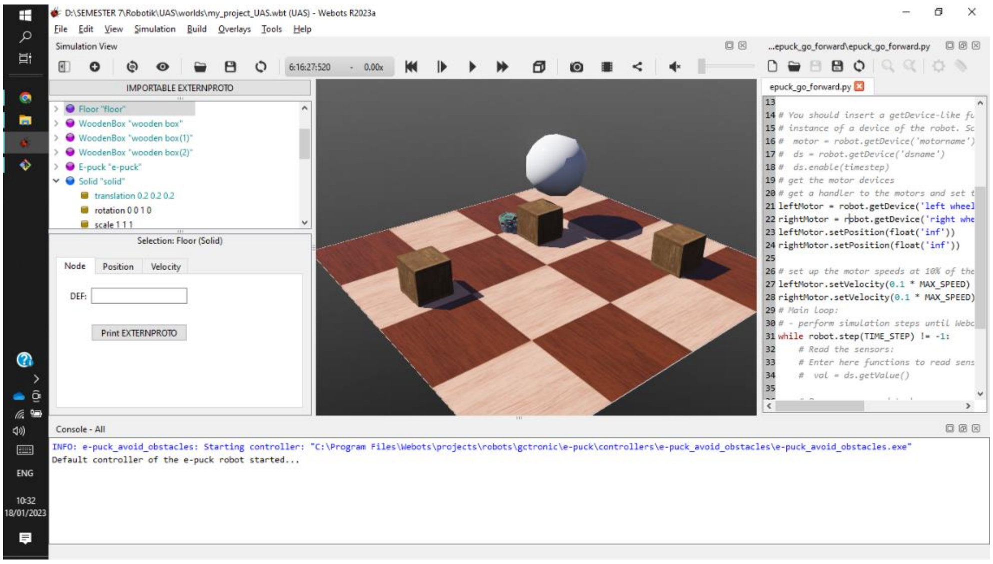
RectangleArena "rectangle arena"
translation 0 0 0
rotation 0 0 1 0
name "rectangle arena"
contactMaterial "default"
floorSize 1 1
floorTileSize 0.25 0.25
floorAppearance Parquetry
wallThickness 0.01

```
# You should insert a getDevice-like fu
# instance of a device of the robot. Sc
#  motor = robot.getDevice('motorname')
#  ds = robot.getDevice('dsname')
#  ds.enable(timestep)
# get the motor devices
# get a handler to the motors and set t
leftMotor = robot.getDevice('left wheel
rightMotor = robot.getDevice('right whe
leftMotor.setPosition(float('inf'))
rightMotor.setPosition(float('inf'))

# set up the motor speeds at 10% of the
leftMotor.setVelocity(0.1 * MAX_SPEED)
rightMotor.setVelocity(0.1 * MAX_SPEED)
# Main loop:
# - perform simulation steps until Webc
while robot.step(TIME_STEP) != -1:
    # Read the sensors:
    # Enter here functions to read sens
    #  val = ds.getValue()

```

```
INFO: e-puck_avoid_obstacles: Starting controller: "C:\Program Files\Webots\projects\robots\gctronic\e-puck\controllers\e-puck_avoid_obstacles\e-puck_avoid_obstacles.exe"
Default controller of the e-puck robot started...
```

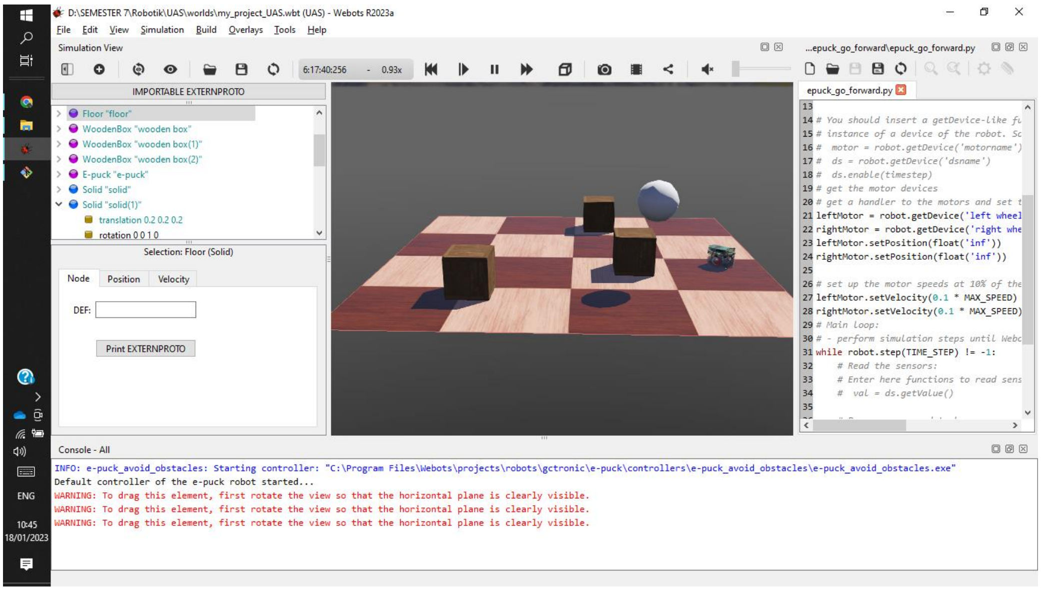
2. Tutorial 2: Modification of the Environment

- in this step learn how to make a ball, by combining 2 shapes
- at this stage learn the concept of nodes, how to create nodes and node inheritance



```
# You should insert a getDevice-like fu
# instance of a device of the robot. Sc
#  motor = robot.getDevice('motorname')
#  ds = robot.getDevice('dsname')
#  ds.enable(timestep)
# get the motor devices
# get a handler to the motors and set t
leftMotor = robot.getDevice('left wheel
rightMotor = robot.getDevice('right whe
leftMotor.setPosition(float('inf'))
rightMotor.setPosition(float('inf'))

# set up the motor speeds at 10% of the
leftMotor.setVelocity(0.1 * MAX_SPEED)
rightMotor.setVelocity(0.1 * MAX_SPEED)
# Main loop:
# - perform simulation steps until Webc
while robot.step(TIME_STEP) != -1:
    # Read the sensors:
    # Enter here functions to read sens
    #  val = ds.getValue()
```

```
INFO: e-puck_avoid_obstacles: Starting controller: "C:\Program Files\Webots\projects\robots\gctronic\e-puck\controllers\e-puck_avoid_obstacles\e-puck_avoid_obstacles.exe"
Default controller of the e-puck robot started...
```

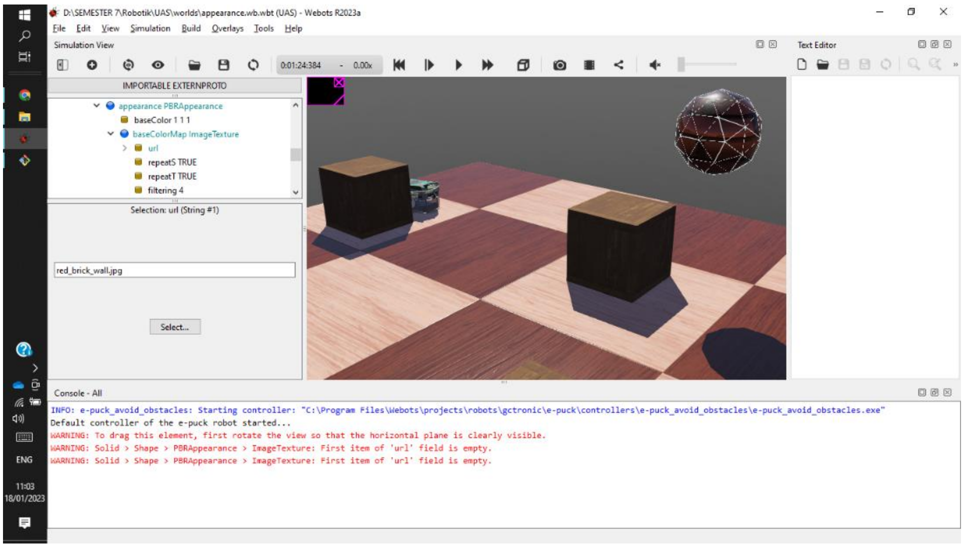


```
# You should insert a getDevice-like fu
# instance of a device of the robot. Sc
#  motor = robot.getDevice('motorname')
#  ds = robot.getDevice('dsname')
#  ds.enable(timestep)
# get the motor devices
# get a handler to the motors and set t
leftMotor = robot.getDevice('left wheel
rightMotor = robot.getDevice('right whe
leftMotor.setPosition(float('inf'))
rightMotor.setPosition(float('inf'))

# set up the motor speeds at 10% of the
leftMotor.setVelocity(0.1 * MAX_SPEED)
rightMotor.setVelocity(0.1 * MAX_SPEED)
# Main loop:
# - perform simulation steps until Webc
while robot.step(TIME_STEP) != -1:
    # Read the sensors:
    # Enter here functions to read sens
    #  val = ds.getValue()
```

```
INFO: e-puck_avoid_obstacles: Starting controller: "C:\Program Files\Webots\projects\robots\gctronic\e-puck\controllers\e-puck_avoid_obstacles\e-puck_avoid_obstacles.exe"
Default controller of the e-puck robot started...
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
```

3. Tutorial 3: Appearance

- at this stage learn the graphic display well. starting from coloring the objects and providing textures that can be imported from local images

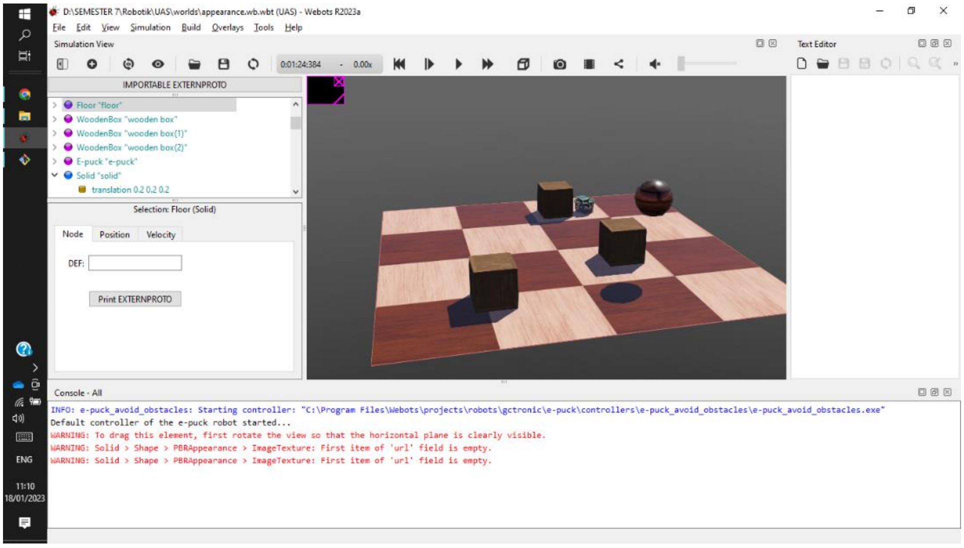

D:\SEMESTER 7\Robotik\UAS\worlds\appearance.wb.wbt (UAS) - Webots R2023a
File Edit View Simulation Build Overlays Tools Help
Simulation View
0:01:24:384
0.00x
Text Editor
IMPORTABLE EXTERNPROTO
Select...
Console - All
ENG
11:03
18/01/2023

appearance PBRAppearance
baseColor 1 1 1
baseColorMap ImageTexture
url
repeatS TRUE
repeatT TRUE
filtering 4

Selection: url (String #1)

red_brick_wall.jpg

```
INFO: e-puck_avoid_obstacles: Starting controller: "C:\Program Files\Webots\projects\robots\gctronic\e-puck\controllers\e-puck_avoid_obstacles\e-puck_avoid_obstacles.exe"
Default controller of the e-puck robot started...
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: Solid > Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
WARNING: Solid > Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
```



Floor "floor"
WoodenBox "wooden box"
WoodenBox "wooden box(1)"
WoodenBox "wooden box(2)"
E-puck "e-puck"
Solid "solid"
translation 0.2 0.2 0.2

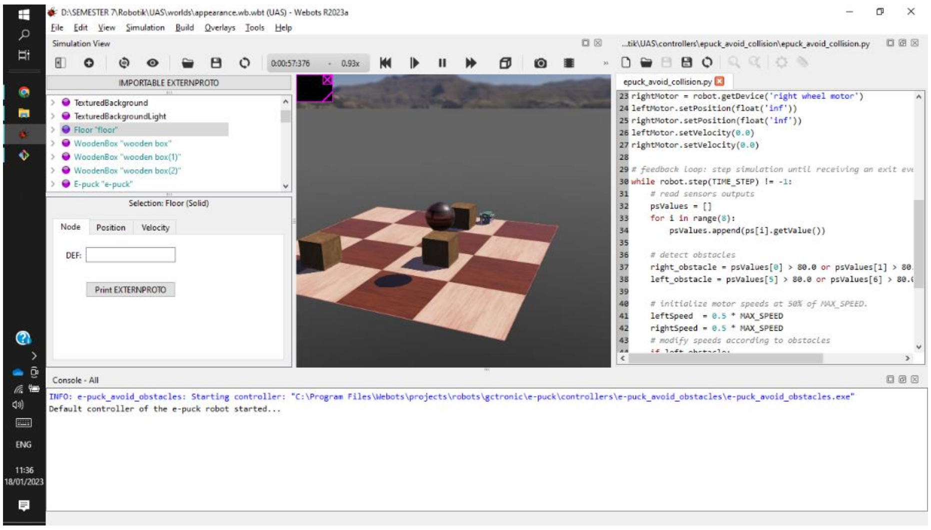
Selection: Floor (Solid)

DEF:

```
INFO: e-puck_avoid_obstacles: Starting controller: "C:\Program Files\Webots\projects\robots\gctronic\e-puck\controllers\e-puck_avoid_obstacles\e-puck_avoid_obstacles.exe"
Default controller of the e-puck robot started...
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: Solid > Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
WARNING: Solid > Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
```

4. Tutorial 4: More about Controllers

- This tutorial will introduce to the basics of robot programming in Webots. At the end of this chapter, you should understand what is the link between the scene tree nodes and the controller API, how the robot controller has to be initialized and cleaned up, how to initialize the robot devices, how to get the sensor values, how to command the actuators, and how to program a simple feedback loop.

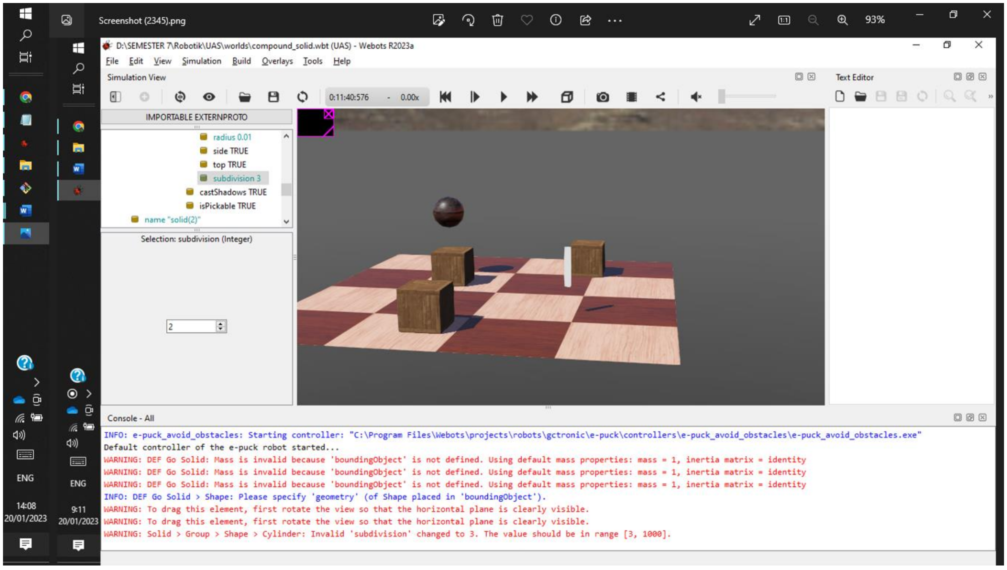

D:\SEMESTER 7\Robotik\UAS\worlds\appearance.wb.wbt (UAS) - Webots R2023a
File Edit View Simulation Build Overlays Tools Help
Simulation View
IMPORTABLE EXTERNPROTO
TexturedBackground
TexturedBackgroundLight
Floor "floor"
WoodenBox "wooden box"
WoodenBox "wooden box(1)"
WoodenBox "wooden box(2)"
E-puck "e-puck"
Selection: Floor (Solid)
Node
Position
Velocity
DEF:
Print EXTERNPROTO
...tik\UAS\controllers\epuck_avoid_collision\epuck_avoid_collision.py
epuck_avoid_collision.py
Console - All

```
rightMotor = robot.getDevice('right wheel motor')
leftMotor.setPosition(float('inf'))
rightMotor.setPosition(float('inf'))
leftMotor.setVelocity(0.0)
rightMotor.setVelocity(0.0)

# feedback loop: step simulation until receiving an exit eve
while robot.step(TIME_STEP) != -1:
    # read sensors outputs
    psValues = []
    for i in range(8):
        psValues.append(ps[i].getValue())

    # detect obstacles
    right_obstacle = psValues[0] > 80.0 or psValues[1] > 80.
    left_obstacle = psValues[5] > 80.0 or psValues[6] > 80.

    # initialize motor speeds at 50% of MAX_SPEED.
    leftSpeed  = 0.5 * MAX_SPEED
    rightSpeed = 0.5 * MAX_SPEED
    # modify speeds according to obstacles
```

```
INFO: e-puck_avoid_obstacles: Starting controller: "C:\Program Files\Webots\projects\robots\gctronic\e-puck\controllers\e-puck_avoid_obstacles\e-puck_avoid_obstacles.exe"
Default controller of the e-puck robot started...
```

5. Tutorial 5: Compound Solid and Physics Attributes

- In the following simulation, learn more about physics, by combining several objects so that they become dumbbells
- there are three children who have their own inheritance, there are two balls and a tube which has a ball at each end
- step 1 create node cylinder, setup size, radius, and position


Screenshot (2345).png
93%
D:\SEMESTER 7\Robotik\UAS\worlds\compound_solid.wbt (UAS) - Webots R2023a
File Edit View Simulation Build Overlays Tools Help
Simulation View
Text Editor
IMPORTABLE EXTERNPROTO
Console - All

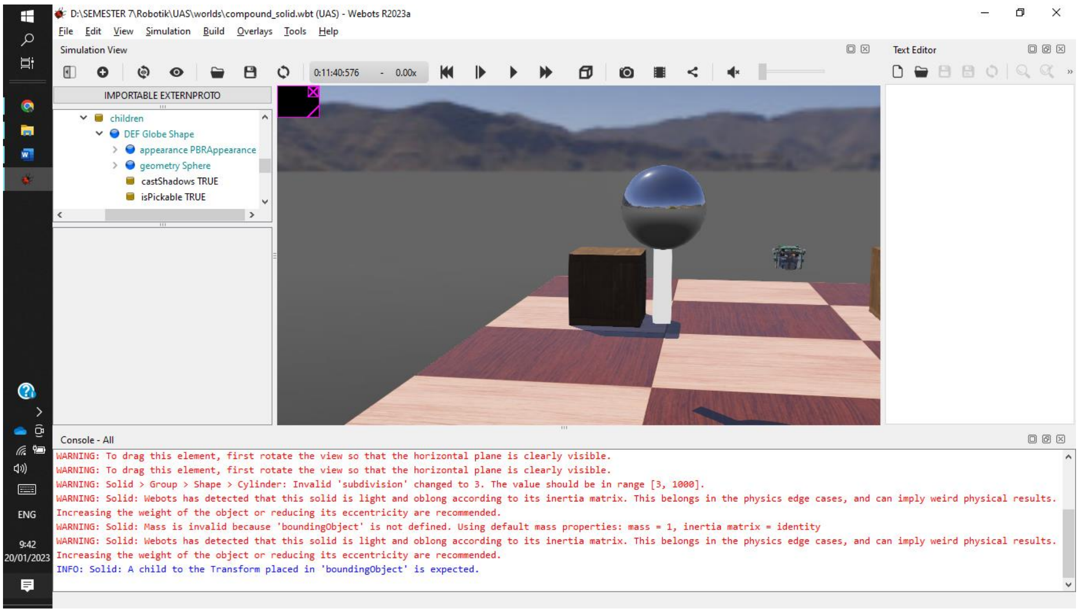
radius 0.01
side TRUE
top TRUE
subdivision 3
castShadows TRUE
isPickable TRUE
name "solid(2)"

Selection: subdivision (Integer)

2

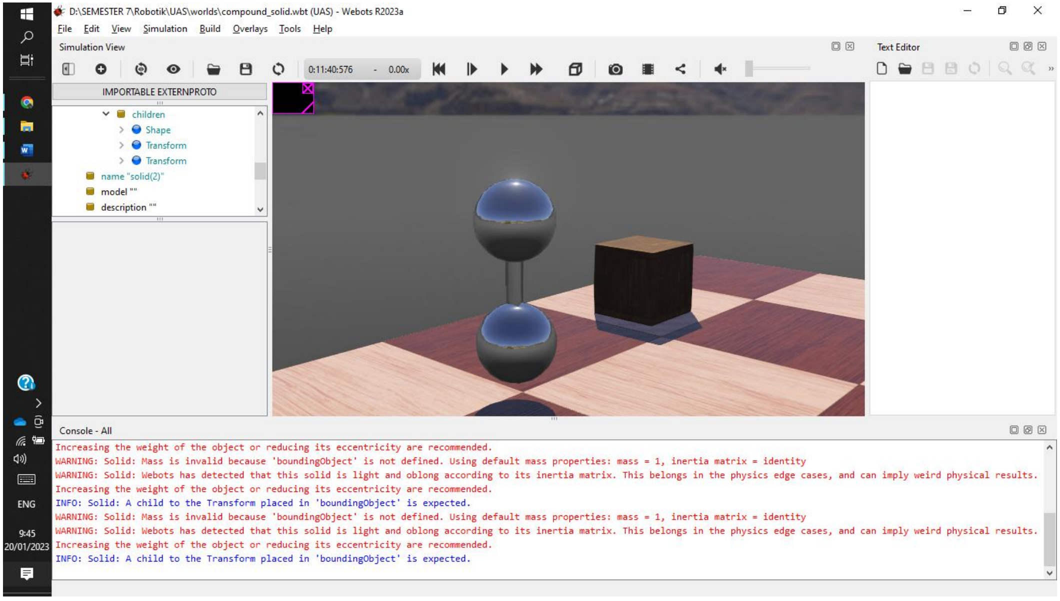
```
INFO: e-puck_avoid_obstacles: Starting controller: "C:\Program Files\Webots\projects\robots\gctronic\e-puck\controllers\e-puck_avoid_obstacles\e-puck_avoid_obstacles.exe"
Default controller of the e-puck robot started...
WARNING: DEF Go Solid: Mass is invalid because 'boundingObject' is not defined. Using default mass properties: mass = 1, inertia matrix = identity
WARNING: DEF Go Solid: Mass is invalid because 'boundingObject' is not defined. Using default mass properties: mass = 1, inertia matrix = identity
WARNING: DEF Go Solid: Mass is invalid because 'boundingObject' is not defined. Using default mass properties: mass = 1, inertia matrix = identity
INFO: DEF Go Solid > Shape: Please specify 'geometry' (of Shape placed in 'boundingObject').
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: Solid > Group > Shape > Cylinder: Invalid 'subdivision' changed to 3. The value should be in range [3, 1000].
```

- the next step, transform the object by adding a transform node, and create a new sphere object node by inheriting the shape. Set up transformation x, y, z-axis until position at the end of the cylinder.



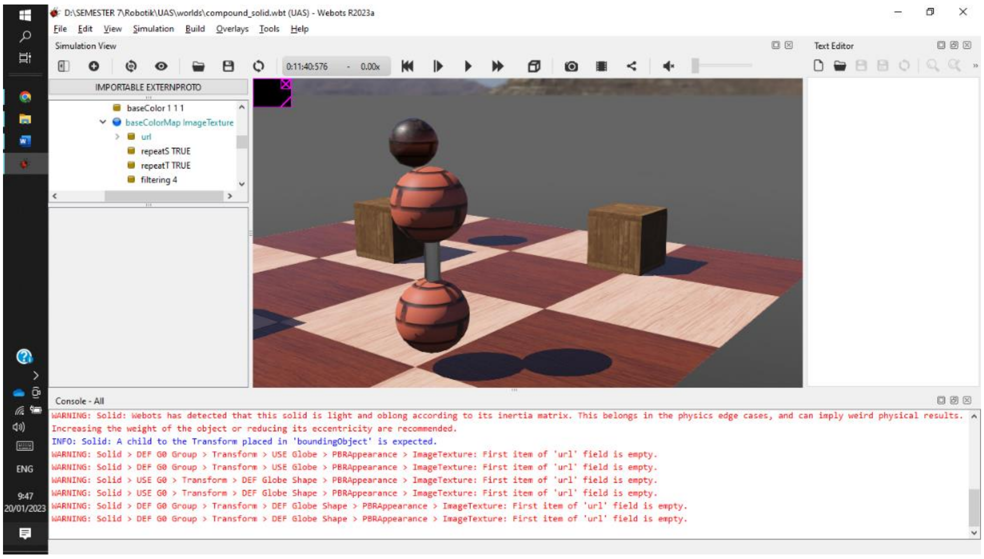
```
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: To drag this element, first rotate the view so that the horizontal plane is clearly visible.
WARNING: Solid > Group > Shape > Cylinder: Invalid 'subdivision' changed to 3. The value should be in range [3, 1000].
WARNING: Solid: Webots has detected that this solid is light and oblong according to its inertia matrix. This belongs in the physics edge cases, and can imply weird physical results.
Increasing the weight of the object or reducing its eccentricity are recommended.
WARNING: Solid: Mass is invalid because 'boundingObject' is not defined. Using default mass properties: mass = 1, inertia matrix = identity
WARNING: Solid: Webots has detected that this solid is light and oblong according to its inertia matrix. This belongs in the physics edge cases, and can imply weird physical results.
Increasing the weight of the object or reducing its eccentricity are recommended.
INFO: Solid: A child to the Transform placed in 'boundingObject' is expected.
```

- copy the previous node and set the transform to be at the position of the other end of the tube



```
Increasing the weight of the object or reducing its eccentricity are recommended.
WARNING: Solid: Mass is invalid because 'boundingObject' is not defined. Using default mass properties: mass = 1, inertia matrix = identity
WARNING: Solid: Webots has detected that this solid is light and oblong according to its inertia matrix. This belongs in the physics edge cases, and can imply weird physical results.
Increasing the weight of the object or reducing its eccentricity are recommended.
INFO: Solid: A child to the Transform placed in 'boundingObject' is expected.
WARNING: Solid: Mass is invalid because 'boundingObject' is not defined. Using default mass properties: mass = 1, inertia matrix = identity
WARNING: Solid: Webots has detected that this solid is light and oblong according to its inertia matrix. This belongs in the physics edge cases, and can imply weird physical results.
Increasing the weight of the object or reducing its eccentricity are recommended.
INFO: Solid: A child to the Transform placed in 'boundingObject' is expected.
```

- finally , give a texture at the ball and cylinder so that beautifully object



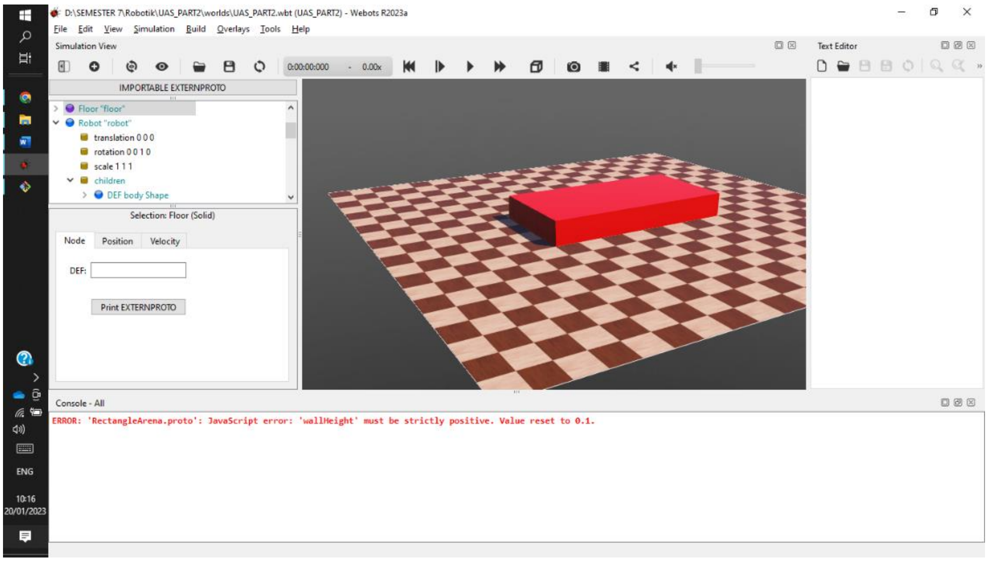
D:\SEMESTER 7\Robotik\UAS\worlds\compound_solid.wbt (UAS) - Webots R2023a
File Edit View Simulation Build Overlays Tools Help
Simulation View
Text Editor
0:11:40:576
0.00x
IMPORTABLE EXTERNPROTO
Console - All
ENG
9:47
20/01/2023

baseColor 1 1 1
baseColorMap ImageTexture
url
repeatS TRUE
repeatT TRUE
filtering 4

```
WARNING: Solid: Webots has detected that this solid is light and oblong according to its inertia matrix. This belongs in the physics edge cases, and can imply weird physical results.
Increasing the weight of the object or reducing its eccentricity are recommended.
INFO: Solid: A child to the Transform placed in 'boundingObject' is expected.
WARNING: Solid > DEF G0 Group > Transform > USE Globe > PBRAppearance > ImageTexture: First item of 'url' field is empty.
WARNING: Solid > DEF G0 Group > Transform > USE Globe > PBRAppearance > ImageTexture: First item of 'url' field is empty.
WARNING: Solid > USE G0 > Transform > DEF Globe Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
WARNING: Solid > USE G0 > Transform > DEF Globe Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
WARNING: Solid > DEF G0 Group > Transform > DEF Globe Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
WARNING: Solid > DEF G0 Group > Transform > DEF Globe Shape > PBRAppearance > ImageTexture: First item of 'url' field is empty.
```

6. Tutorial 6: 4-Wheels Robot

- at this time will make a robot with a sensor
- first create node solid nodes in the form of a rectangular block



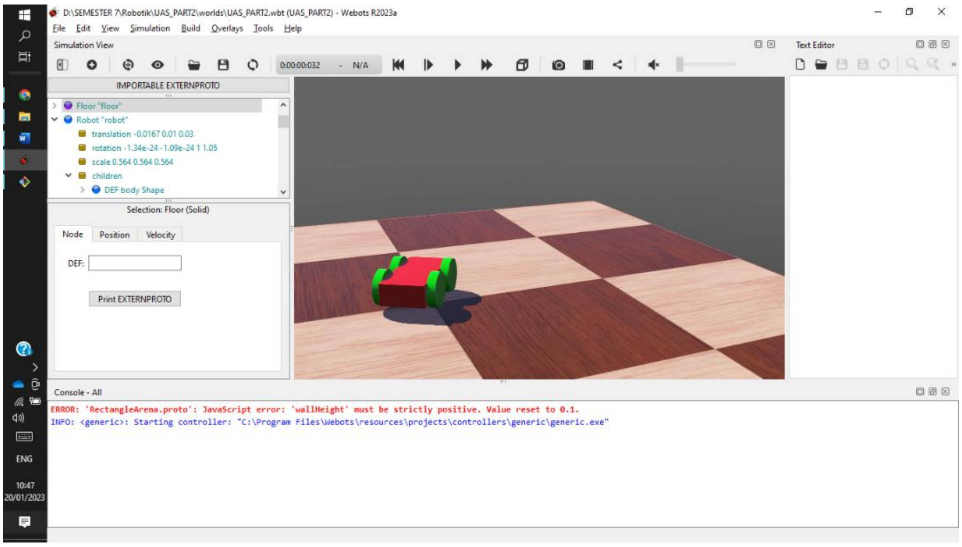
Floor "floor"
Robot "robot"
translation 0 0 0
rotation 0 0 1 0
scale 1 1 1
children
DEF body Shape

Selection: Floor (Solid)

DEF:

```
ERROR: 'RectangleArena.proto': JavaScript error: 'wallHeight' must be strictly positive. Value reset to 0.1.
```

- the next step, make wheels on the robot using the hingejoint node and set the size and rotation, then define it as wheel1. so is done on the next 4 wheels



D:\SEMESTER 7\Robotik\UAS_PART2\worlds\UAS_PART2.wbt (UAS_PART2) - Webots R2023a

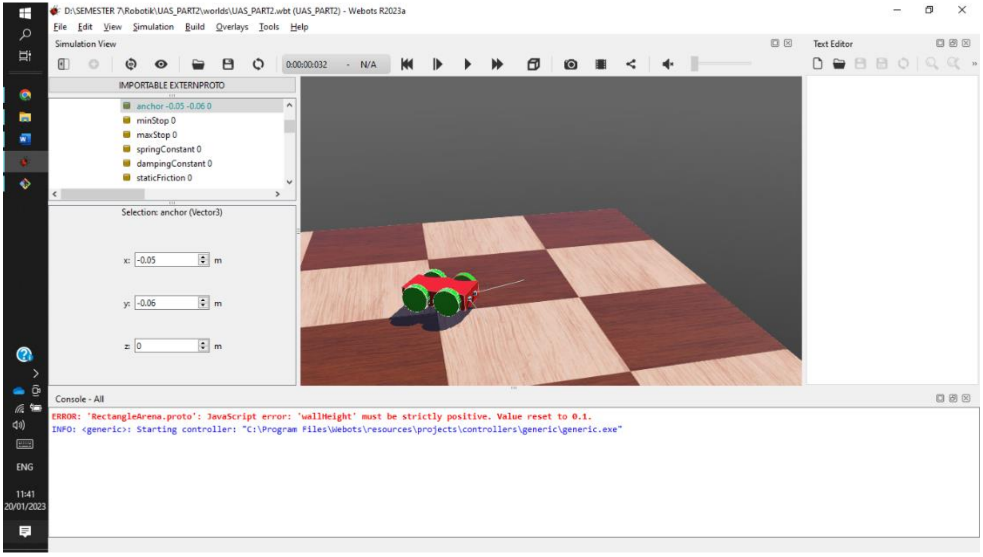
Floor "floor"
Robot "robot"
translation -0.0167 0.01 0.03
rotation -1.34e-24 -1.09e-24 1 1.05
scale 0.564 0.564 0.564
children
DEF body Shape

Selection: Floor (Solid)

DEF:

```
ERROR: 'RectangleArena.proto': JavaScript error: 'wallHeight' must be strictly positive. Value reset to 0.1.
INFO: <generic>: Starting controller: "C:\Program Files\Webots\resources\projects\controllers\generic\generic.exe"
```

- then add a proximity sensor on the front of the robot



D:\SEMESTER 7\Robotik\UAS_PART2\worlds\UAS_PART2.wbt (UAS_PART2) - Webots R2023a

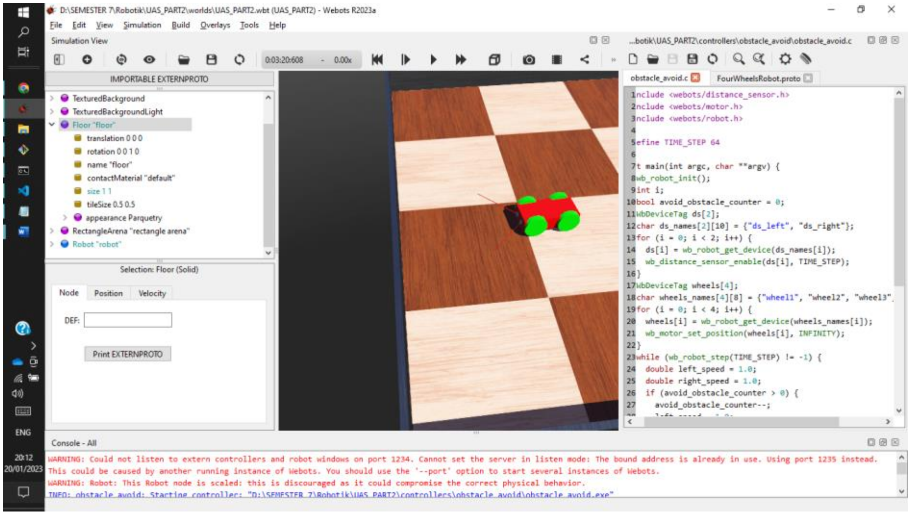
anchor -0.05 -0.06 0
minStop 0
maxStop 0
springConstant 0
dampingConstant 0
staticFriction 0

Selection: anchor (Vector3)

x: -0.05 m
y: -0.06 m
z: 0 m

```
ERROR: 'RectangleArena.proto': JavaScript error: 'wallHeight' must be strictly positive. Value reset to 0.1.
INFO: <generic>: Starting controller: "C:\Program Files\Webots\resources\projects\controllers\generic\generic.exe"
```

- then give the controller to the robot so it can walk



D:\SEMESTER 7\Robotik\UAS_PART2\worlds\UAS_PART2.wbt (UAS_PART2) - Webots R2023a

TexturedBackground
TexturedBackgroundLight
Floor "floor"
translation 0 0 0
rotation 0 0 1 0
name "floor"
contactMaterial "default"
size 1 1
tileSize 0.5 0.5
appearance Parquetry
RectangleArena "rectangle arena"
Robot "robot"

Selection: Floor (Solid)

DEF:

...botik\UAS_PART2\controllers\obstacle_avoid\obstacle_avoid.c

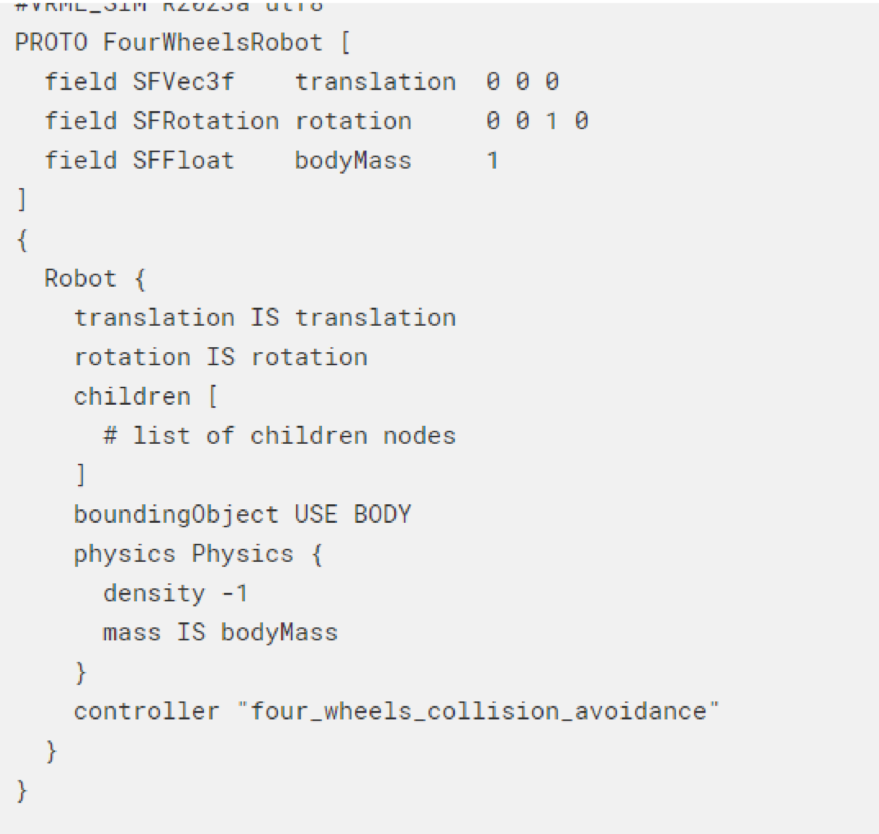
obstacle_avoid.c FourWheelsRobot.proto

```
nclude <webots/distance_sensor.h>
nclude <webots/motor.h>
nclude <webots/robot.h>

efine TIME_STEP 64

t main(int argc, char **argv) {
wb_robot_init();
int i;
bool avoid_obstacle_counter = 0;
WbDeviceTag ds[2];
char ds_names[2][10] = {"ds_left", "ds_right"};
for (i = 0; i < 2; i++) {
  ds[i] = wb_robot_get_device(ds_names[i]);
  wb_distance_sensor_enable(ds[i], TIME_STEP);
}
WbDeviceTag wheels[4];
char wheels_names[4][8] = {"wheel1", "wheel2", "wheel3",
for (i = 0; i < 4; i++) {
  wheels[i] = wb_robot_get_device(wheels_names[i]);
  wb_motor_set_position(wheels[i], INFINITY);
}
while (wb_robot_step(TIME_STEP) != -1) {
  double left_speed = 1.0;
  double right_speed = 1.0;
  if (avoid_obstacle_counter > 0) {
    avoid_obstacle_counter--;
```

```
WARNING: Could not listen to extern controllers and robot windows on port 1234. Cannot set the server in listen mode: The bound address is already in use. Using port 1235 instead.
This could be caused by another running instance of Webots. You should use the '--port' option to start several instances of Webots.
WARNING: Robot: This Robot node is scaled: this is discouraged as it could compromise the correct physical behavior.
INFO: obstacle_avoid: Starting controller: "D:\SEMESTER 7\Robotik\UAS_PART2\controllers\obstacle_avoid\obstacle_avoid.exe"
```

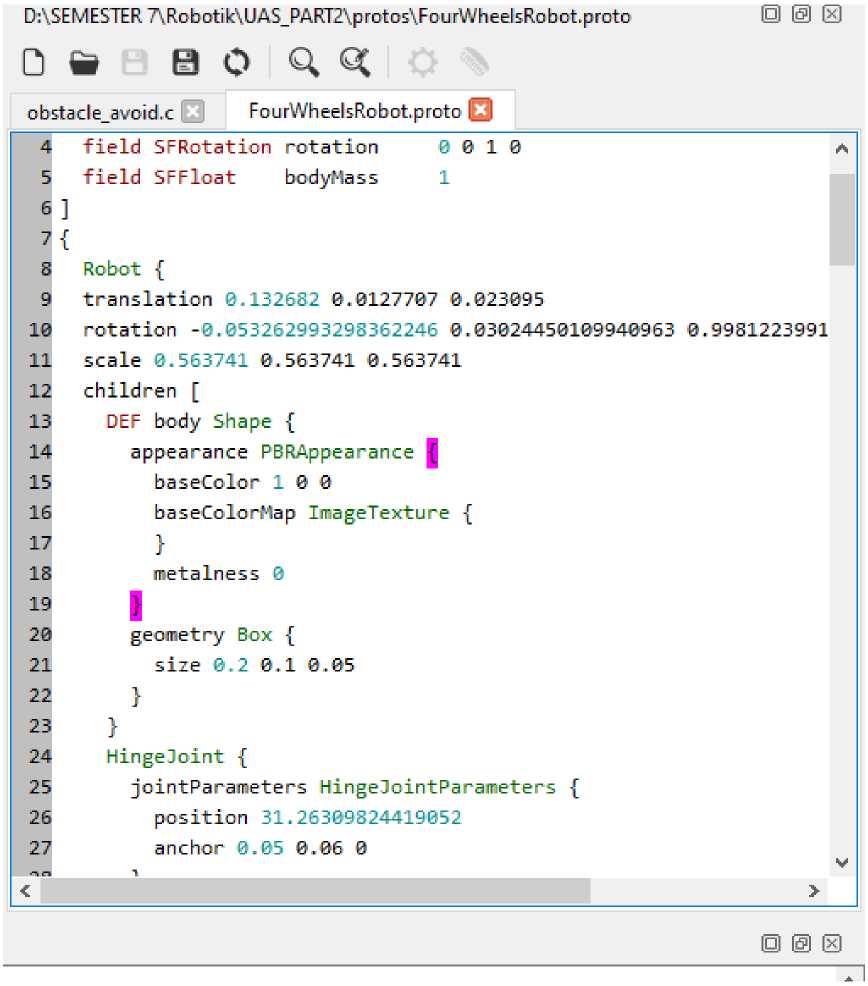
7. Tutorial 7: Your First PROTO

- create a new proto then initialize the following code



```
#VRML_SIM R2023a utf8
PROTO FourWheelsRobot [
  field SFVec3f    translation  0 0 0
  field SFRotation rotation     0 0 1 0
  field SFFloat    bodyMass     1
]
{
  Robot {
    translation IS translation
    rotation IS rotation
    children [
      # list of children nodes
    ]
    boundingObject USE BODY
    physics Physics {
      density -1
      mass IS bodyMass
    }
    controller "four_wheels_collision_avoidance"
  }
}
```

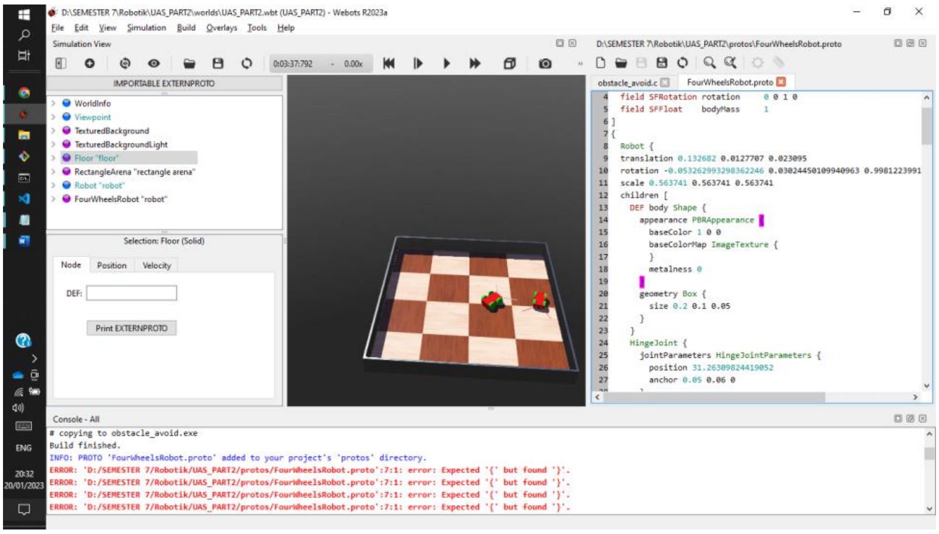
- after that copy the code in the .wbt file about the robot definition



D:\SEMESTER 7\Robotik\UAS_PART2\protos\FourWheelsRobot.proto

```
  field SFRotation rotation      0 0 1 0
  field SFFloat     bodyMass     1
]
{
  Robot {
  translation 0.132682 0.0127707 0.023095
  rotation -0.053262993298362246 0.03024450109940963 0.9981223991
  scale 0.563741 0.563741 0.563741
  children [
    DEF body Shape {
      appearance PBRAppearance {
        baseColor 1 0 0
        baseColorMap ImageTexture {
        }
        metalness 0
      }
      geometry Box {
        size 0.2 0.1 0.05
      }
    }
    HingeJoint {
      jointParameters HingeJointParameters {
        position 31.26309824419052
        anchor 0.05 0.06 0
```

- adding the new node the new proto. Walla , now me have two same robots



D:\SEMESTER 7\Robotik\UAS_PART2\worlds\UAS_PART2.wbt (UAS_PART2) - Webots R2023a

- WorldInfo
- Viewpoint
- TexturedBackground
- TexturedBackgroundLight
- Floor "floor"
- RectangleArena "rectangle arena"
- Robot "robot"
- FourWheelsRobot "robot"

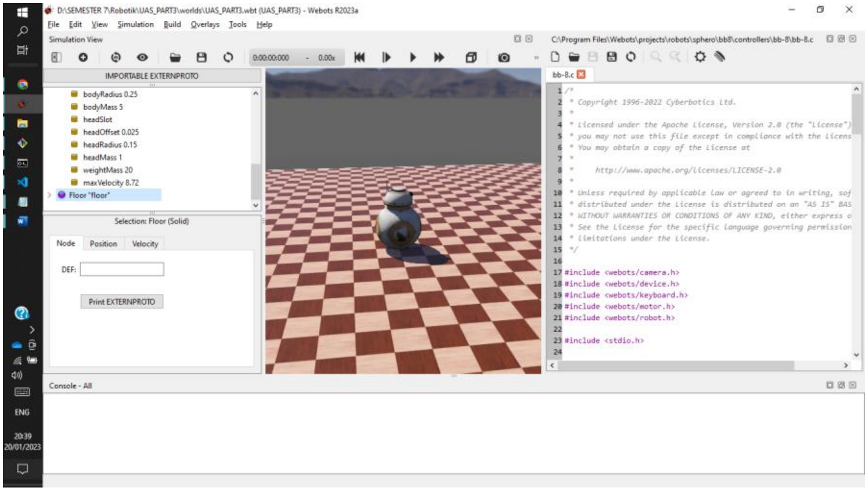
D:\SEMESTER 7\Robotik\UAS_PART2\protos\FourWheelsRobot.proto

```
  field SFRotation rotation      0 0 1 0
  field SFFloat     bodyMass     1
]
{
  Robot {
  translation 0.132682 0.0127707 0.023095
  rotation -0.053262993298362246 0.03024450109940963 0.9981223991
  scale 0.563741 0.563741 0.563741
  children [
    DEF body Shape {
      appearance PBRAppearance {
        baseColor 1 0 0
        baseColorMap ImageTexture {
        }
        metalness 0
      }
      geometry Box {
        size 0.2 0.1 0.05
      }
    }
    HingeJoint {
      jointParameters HingeJointParameters {
        position 31.26309824419052
        anchor 0.05 0.06 0
```

```
# copying to obstacle_avoid.exe
Build finished.
INFO: PROTO 'FourWheelsRobot.proto' added to your project's 'protos' directory.
ERROR: 'D:/SEMESTER 7/Robotik/UAS_PART2/protos/FourWheelsRobot.proto':7:1: error: Expected '{' but found '}'.
ERROR: 'D:/SEMESTER 7/Robotik/UAS_PART2/protos/FourWheelsRobot.proto':7:1: error: Expected '{' but found '}'.
ERROR: 'D:/SEMESTER 7/Robotik/UAS_PART2/protos/FourWheelsRobot.proto':7:1: error: Expected '{' but found '}'.
ERROR: 'D:/SEMESTER 7/Robotik/UAS_PART2/protos/FourWheelsRobot.proto':7:1: error: Expected '{' but found '}'.
```

8. Tutorial 8: The Supervisor

- Create the new directory and add node robot bb-8



D:\SEMESTER 7\Robotik\UAS_PART3\worlds\UAS_PART3.wbt (UAS_PART3) - Webots R2023a

bodyRadius 0.25
bodyMass 5
headSlot
headOffset 0.025
headRadius 0.15
headMass 1
weightMass 20
maxVelocity 8.72
Floor "floor"

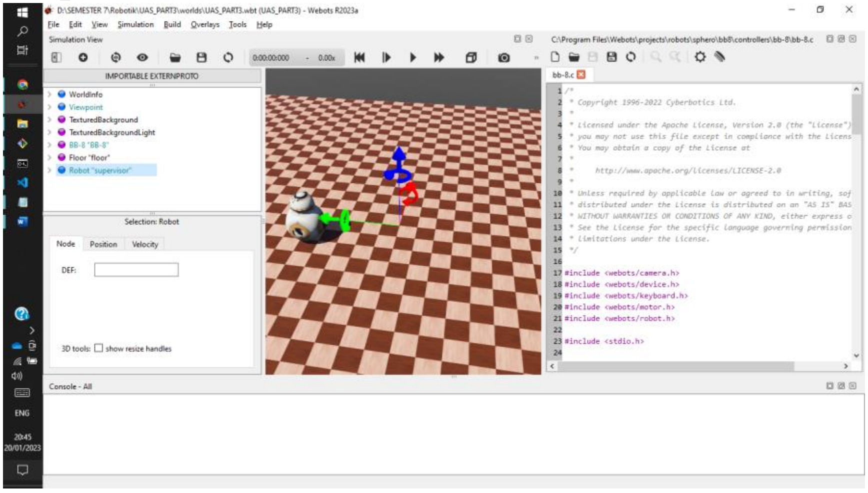
Selection: Floor (Solid)

DEF:

C:\Program Files\Webots\projects\robots\sphero\bb8\controllers\bb-8\bb-8.c

```
/*
 * Copyright 1996-2022 Cyberbotics Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License")
 * you may not use this file except in compliance with the Licens
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, sof
 * distributed under the License is distributed on an "AS IS" BAS
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express o
 * See the License for the specific language governing permission
 * limitations under the License.
 */

#include <webots/camera.h>
#include <webots/device.h>
#include <webots/keyboard.h>
#include <webots/motor.h>
#include <webots/robot.h>

#include <stdio.h>

```

- Create new robot at base node and rename become “supervisor”



D:\SEMESTER 7\Robotik\UAS_PART3\worlds\UAS_PART3.wbt (UAS_PART3) - Webots R2023a

WorldInfo
Viewpoint
TexturedBackground
TexturedBackgroundLight
BB-8 "BB-8"
Floor "floor"
Robot "supervisor"

Selection: Robot

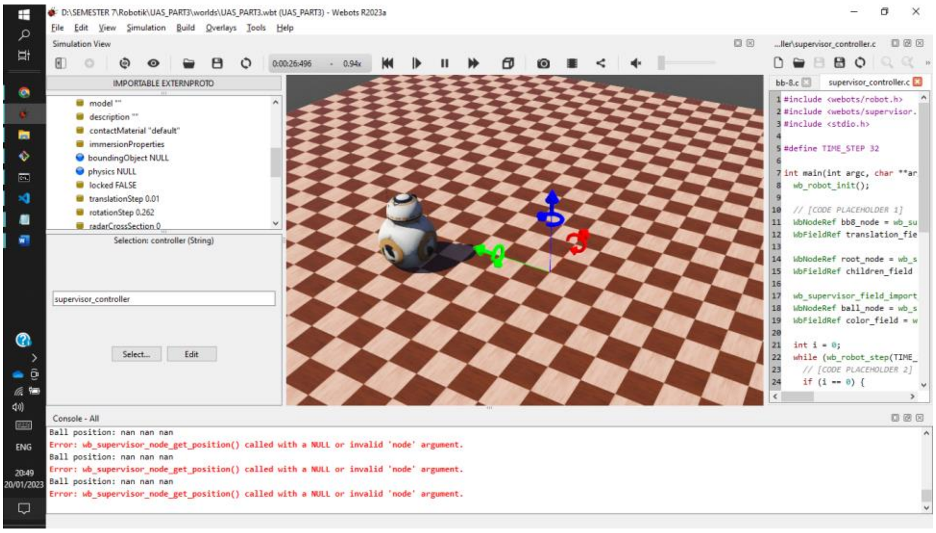
DEF:

3D tools: □ show resize handles

C:\Program Files\Webots\projects\robots\sphero\bb8\controllers\bb-8\bb-8.c

```
/*
 * Copyright 1996-2022 Cyberbotics Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License")
 * you may not use this file except in compliance with the Licens
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, sof
 * distributed under the License is distributed on an "AS IS" BAS
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express o
 * See the License for the specific language governing permission
 * limitations under the License.
 */

#include <webots/camera.h>
#include <webots/device.h>
#include <webots/keyboard.h>
#include <webots/motor.h>
#include <webots/robot.h>

#include <stdio.h>

```

- Finally create new controller rename as “supervisor_controller” and copy code web that. The result!


D:\SEMESTER 7\Robotik\UAS_PART3\worlds\UAS_PART3.wbt (UAS_PART3) - Webots R2023a
File Edit View Simulation Build Overlays Tools Help
Simulation View
0:00:26:496 - 0.94x
IMPORTABLE EXTERNPROTO
Select... Edit
...ller\supervisor_controller.c
bb-8.c
supervisor_controller.c
Console - All
ENG
20:49
20/01/2023

model ""
description ""
contactMaterial "default"
immersionProperties
boundingObject NULL
physics NULL
locked FALSE
translationStep 0.01
rotationStep 0.262
radarCrossSection 0

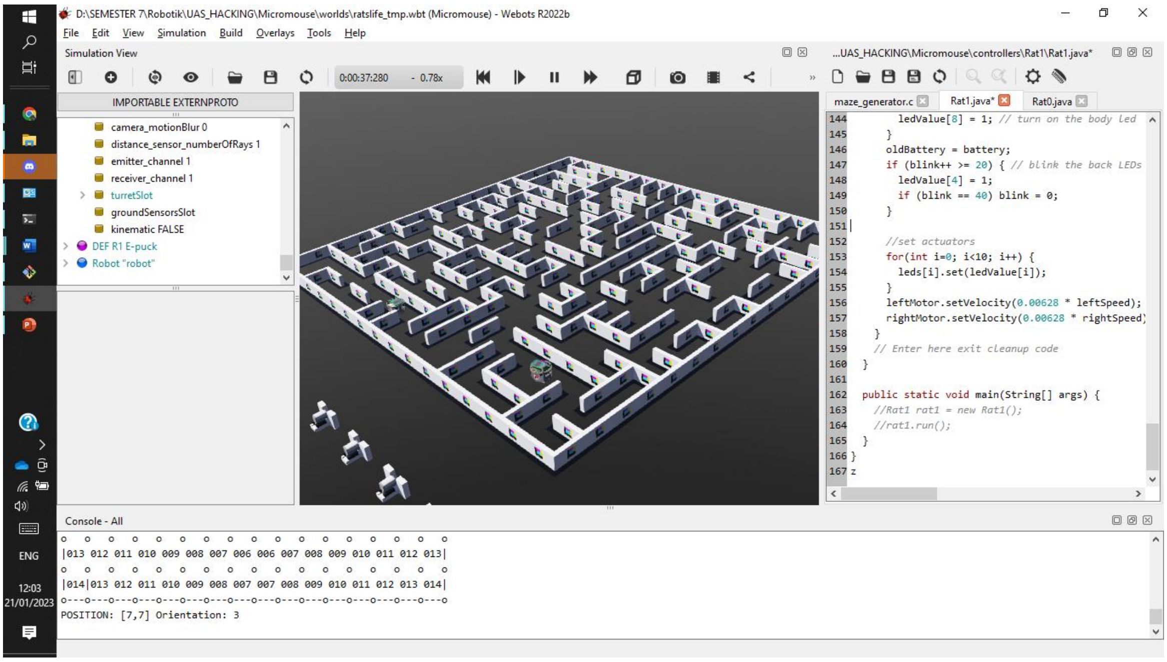
Selection: controller (String)

supervisor_controller

```
#include <webots/robot.h>
#include <webots/supervisor.
#include <stdio.h>

#define TIME_STEP 32

int main(int argc, char **ar
  wb_robot_init();

  // [CODE PLACEHOLDER 1]
  WbNodeRef bb8_node = wb_su
  WbFieldRef translation_fie

  WbNodeRef root_node = wb_s
  WbFieldRef children_field

  wb_supervisor_field_import
  WbNodeRef ball_node = wb_s
  WbFieldRef color_field = w

  int i = 0;
  while (wb_robot_step(TIME_
    // [CODE PLACEHOLDER 2]
    if (i == 0) {
```

```
Ball position: nan nan nan
Error: wb_supervisor_node_get_position() called with a NULL or invalid 'node' argument.
Ball position: nan nan nan
Error: wb_supervisor_node_get_position() called with a NULL or invalid 'node' argument.
Ball position: nan nan nan
Error: wb_supervisor_node_get_position() called with a NULL or invalid 'node' argument.
```

HACKING MICROMOUSE

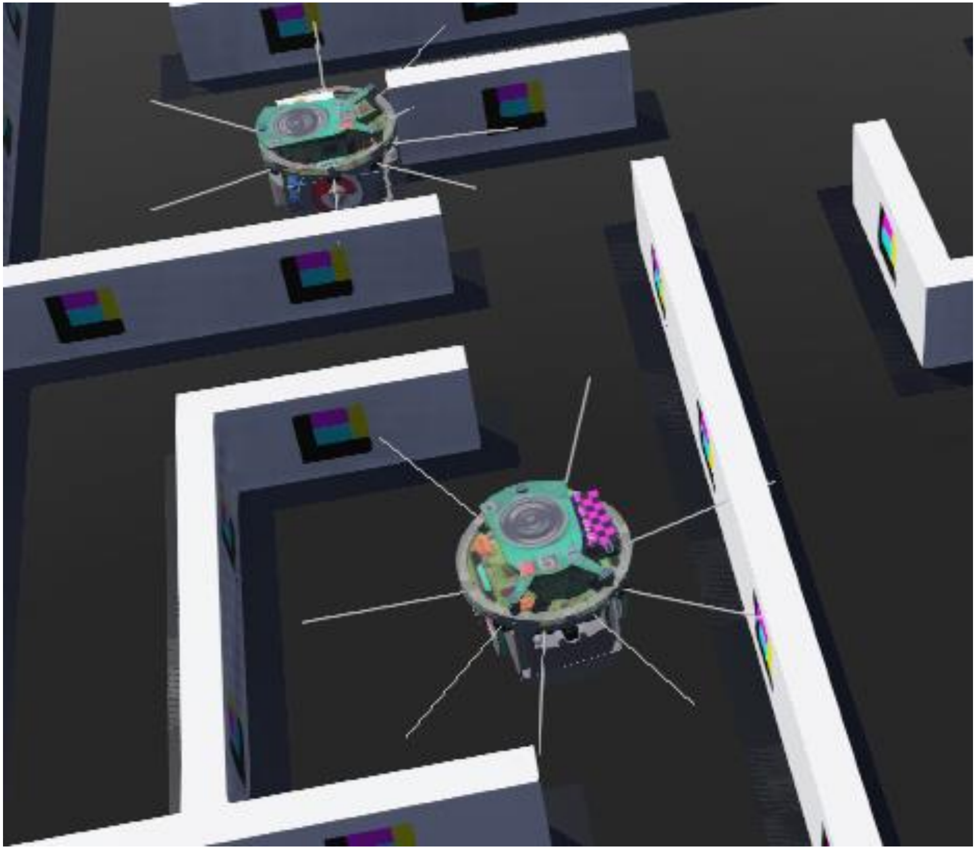

D:\SEMESTER 7\Robotik\UAS_HACKING\Micromouse\worlds\ratslife_tmp.wbt (Micromouse) - Webots R2022b
File Edit View Simulation Build Overlays Tools Help
Simulation View
0:00:37:280 - 0.78x
IMPORTABLE EXTERNPROTO
...UAS_HACKING\Micromouse\controllers\Rat1\Rat1.java*
maze_generator.c
Rat1.java*
Rat0.java
Console - All
ENG
12:03
21/01/2023

camera_motionBlur 0
distance_sensor_numberOfRays 1
emitter_channel 1
receiver_channel 1
turretSlot
groundSensorsSlot
kinematic FALSE
DEF R1 E-puck
Robot "robot"

```
        ledValue[8] = 1; // turn on the body led
      }
      oldBattery = battery;
      if (blink++ >= 20) { // blink the back LEDs
        ledValue[4] = 1;
        if (blink == 40) blink = 0;
      }

      //set actuators
      for(int i=0; i<10; i++) {
        leds[i].set(ledValue[i]);
      }
      leftMotor.setVelocity(0.00628 * leftSpeed);
      rightMotor.setVelocity(0.00628 * rightSpeed)
    }
    // Enter here exit cleanup code
  }

  public static void main(String[] args) {
    //Rat1 rat1 = new Rat1();
    //rat1.run();
  }
}
z
```

```
o   o   o   o   o   o   o   o   o   o   o   o   o   o   o   o   o
|013 012 011 010 009 008 007 006 006 007 008 009 010 011 012 013|
o   o   o   o   o   o   o   o   o   o   o   o   o   o   o   o   o
|014|013 012 011 010 009 008 007 007 008 009 010 011 012 013 014|
o---o---o---o---o---o---o---o---o---o---o---o---o---o---o---o---o
POSITION: [7,7] Orientation: 3
```

This is a micromouse simulation, where there is a labyrinth and in it there is a robot that is programmed to go through the maze to its destination. the e-puck robot is programmed in 2 controllers, the first is rat0 and rat1. of the two controllers, only controller rat0 can work past the labyrinth path obstacles.



For the robot used is the e-punk robot, there are 8 sensors located on each side of the robot, these sensors function to detect maze obstacles so that the robot can not hit obstacles