UNIVERSITY OF TWENTE.

APPLICATION DEVELOPMENT

LECTURE 4: INTRODUCTION TO ARDUINO PROGRAMMING IN C++

class AppDev {

Java

}

Part of SmartProducts

INTRODUCTION

APPLICATION DEVELOPMENT

Fjodor van Slooten
W241 *(Horst-wing West)*
f.vanslooten@utwente.nl

- Introduction to Arduino programming in C++

  (comparison with Java)
- Wheeled vehicles
- Assignment

slides @ vanslooten.com/appdev

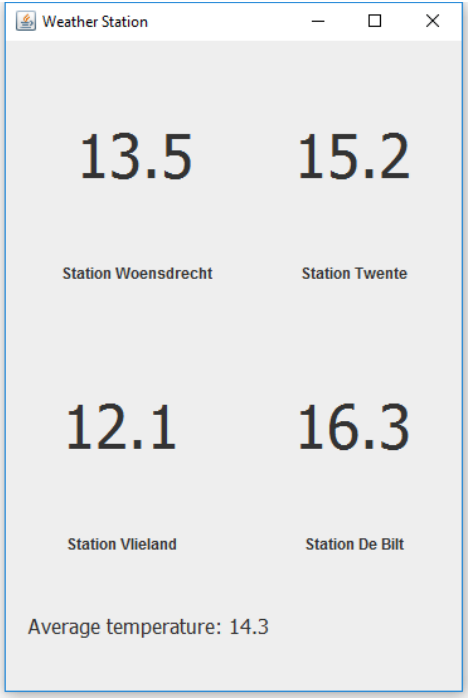
ASSIGNMENT 3

LAST SESSION

- 🙂 Java assignment most of you did very well

Afternoon:

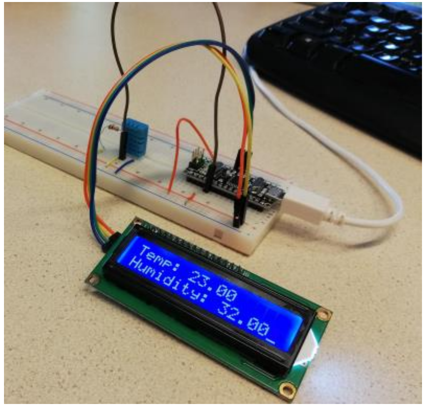
- 🙂 Build Arduino circuit which displays temperature & humidity
- 😥 Connect it (to Wifi and/or Java App)



Weather Station

13.5	15.2
Station Woensdrecht	Station Twente
12.1	16.3
Station Vlieland	Station De Bilt

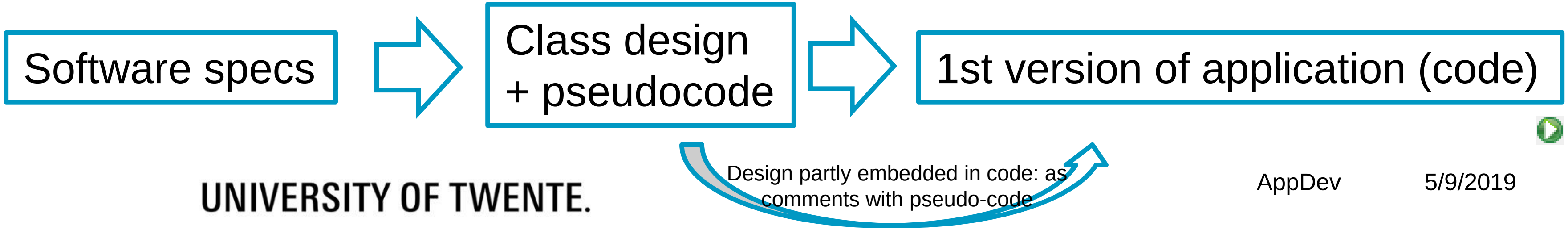
Average temperature: 14.3



FROM SPECS, TO DESIGN, TO CODE

Head First: 1-4 ***Aan de slag met***: 4.8-4.11, 6.1-6.4

- How do I design an application? Check out previous presentations & assignments, read book
- Results of design-phase:
  - Pseudo code
  - Sketches
  - Diagrams (class-diagram, flowchart, …)



Software specs

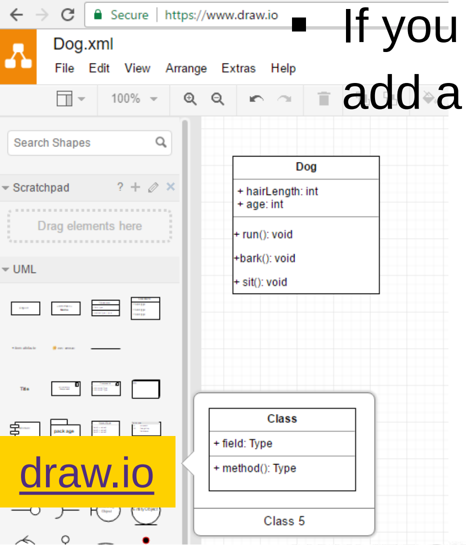
Class design + pseudocode

1st version of application (code)

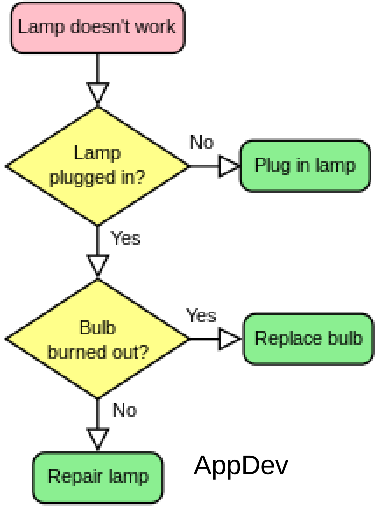
Design partly embedded in code: as comments with pseudo-code

DIAGRAMS

- Can enhance the quality of your design
- If you have multiple classes, create a **class-diagram**
- If you have a complicated flow (of conditions, loops, etc.) add a **Flowchart**



draw.io



FINAL PRODUCT? OR PROTOTYPE?

- Project is focused on prototype (although final product should be considered in design phase)
- @Application Development, design and code is restricted to prototype only
- This means, for the prototype you can:
  - Eliminate/simplify things you can not build
  - Simulate/demonstrate if necessary
- You may show design sketches of final product, as clarification

convince & attract

A prototype demonstrates design to client/ potential customers. And/or you use it for usability tests (goal: prove it can work).

ARDUINO: BASED ON C, C++

DIFFERENCES WITH JAVA

- You can use functions without a class:

You may consider a function is ‘a method without a class’

- Group code, use (call) from multiple locations

Standard functions setup() and loop()

Added function readSensor()

```
void setup(){
  Serial.begin(9600);
}

void loop() {
  int sensorValue = readSensor();
  delay(500);
}

int readSensor(){
  int result = analogRead(A0);
  return result;
}
```

arduino.cc/en/Reference/FunctionDeclaration

A CLASS IN C++

DIFFERENCES WITH JAVA

C++ is used in Arduino IDE

Java:

Dog.java

```
class Dog {
   // properties:
   int hairLength;
   int age;

   // methods:
   public void run() {
   }
   public void bark() {
   }
   public void sit() {
   }
}
```

Arduino/C++:

Dog.h

```
class Dog {
   // properties:
   int hairLength;
   int age;

   // methods declaration:
public:
   void run();
   void bark();
   void sit();
};
```

Declaration

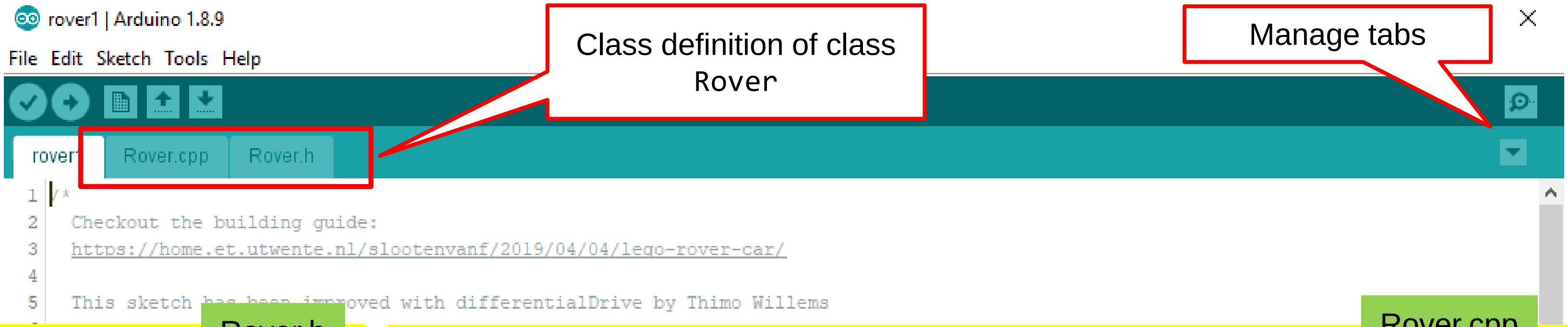
Dog.cpp

```
// methods definition:
void Dog::run() {
}

void Dog::bark() {
}

void Dog::sit() {
}
```

Method definition

C++

COMPARED TO JAVA



```
/*
Checkout the building guide:
https://home.et.utwente.nl/slootenvanf/2019/04/04/lego-rover-car/

This sketch has been improved with differentialDrive by Thimo Willems
```

Class definition of class Rover

Manage tabs

Rover.h

```
class Rover {
// class variables:
private:
// pointers to objects:
  EVShield * evshield;
  NewPing * sonar;
  EVs_NXTTouch * touch;
...
void drive(int distance = 0);
void stop();
...
```

Rover.cpp

```
void Rover::stop() {
  Serial.println("stop()");
  dr_forward=false; dr_backward=false;
  speed=start_speed;
  evshield->bank_a.motorSetSpeed(SH_Motor_Both, start_speed);
  evshield->bank_a.motorStop(SH_Motor_Both, SH_Next_Action_Float);
  evshield->bank_b.ledSetRGB(165, 255, 0);
                        // led orange (indicates ready for driving)
}
```

-> is pointer reference: to call method **ledSetRGB()** (you might be used to use a . here)

CLASS A USES CLASS B

This does not work: “class name does not name a type”
(chicken-and-egg situation)

```
#include <B.h>

class A {
 public:
    A(int id);
 private:
    int id;
    B * objectB;
};
```

```
#include <A.h>

class B {
 public:
    B(int id);
 private:
    int id;
    A * objectA;
};
```

Solution: use *forward declaration:*

```
class B;

class A {
 public:
    A(int id);
 private:
    int id;
    B * objectB;
};
```

```
class A;

class B {
 public:
    B(int id);
 private:
    int id;
    A * objectA;
};
```

'someFunction' was not declared in this scope
+solution

Google the error! > stackoverflow.com/questions/3608305/class-name-does-not-name-a-type-in-c

IF … DRIVING? SEE SOMETHING?

CONDITIONS

These boolean variables are set by driving methods

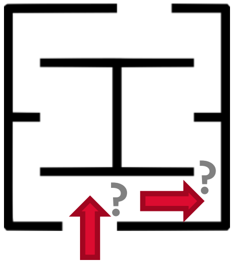
```
class Rover {
...
boolean dr_forward = false, dr_backward = false;
            // moving in forward or backward direction
```

… sensors (bumper, ultrasonic) only need to be checked <u>while driving</u> (forward).
How to check if we are driving?

```
if (dr_forward) {
    // is touchsensor in bumper pressed?
    // does ultrasonic sensor ‘see’ something?
}
```

```
unsigned int distance = sonar->ping_cm();
if (distance > 0 && distance < 30) {
    // find a way out
}
```

How to check if ultrasonic sensor sees something?



MATH

- PI
- abs()

Java:

```
double circumference = car_wheel_diam * Math.PI;
unsigned int degrees = (Math.abs(distance)/circumference) * 360;
```

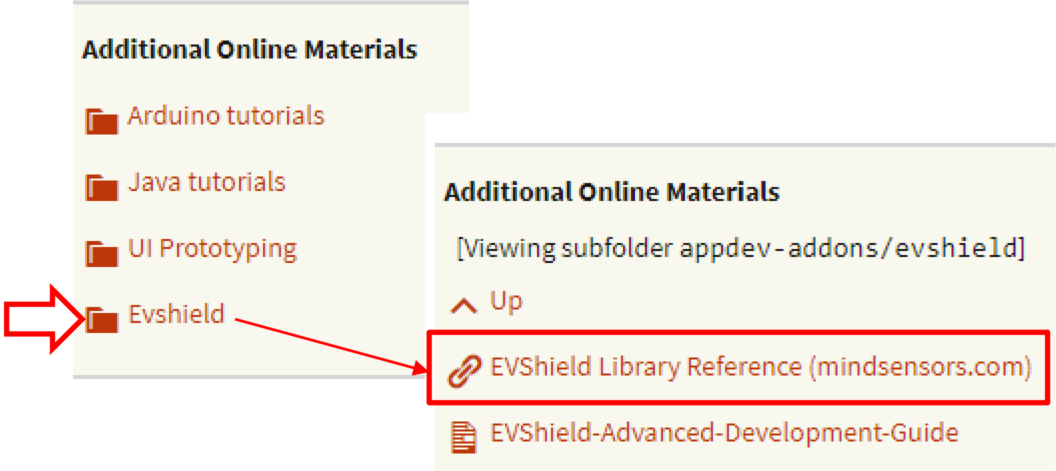
C++:

```
double circumference = car_wheel_diam * PI;
unsigned int degrees = (abs(distance)/circumference) * 360;
```

arduino.cc/en/Math/H

ACCESS EVSHIELD LIBRARY REFERENCE

Motor… commands?



Additional Online Materials

- Arduino tutorials
- Java tutorials
- UI Prototyping
- Evshield

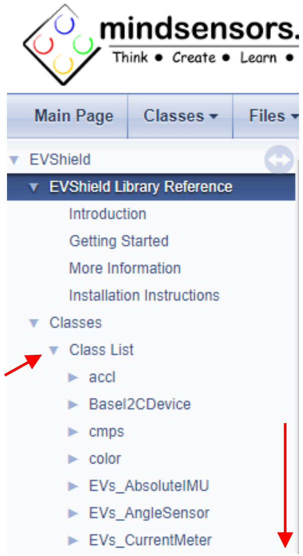
Additional Online Materials

[Viewing subfolder appdev-addons/evshield]

^ Up

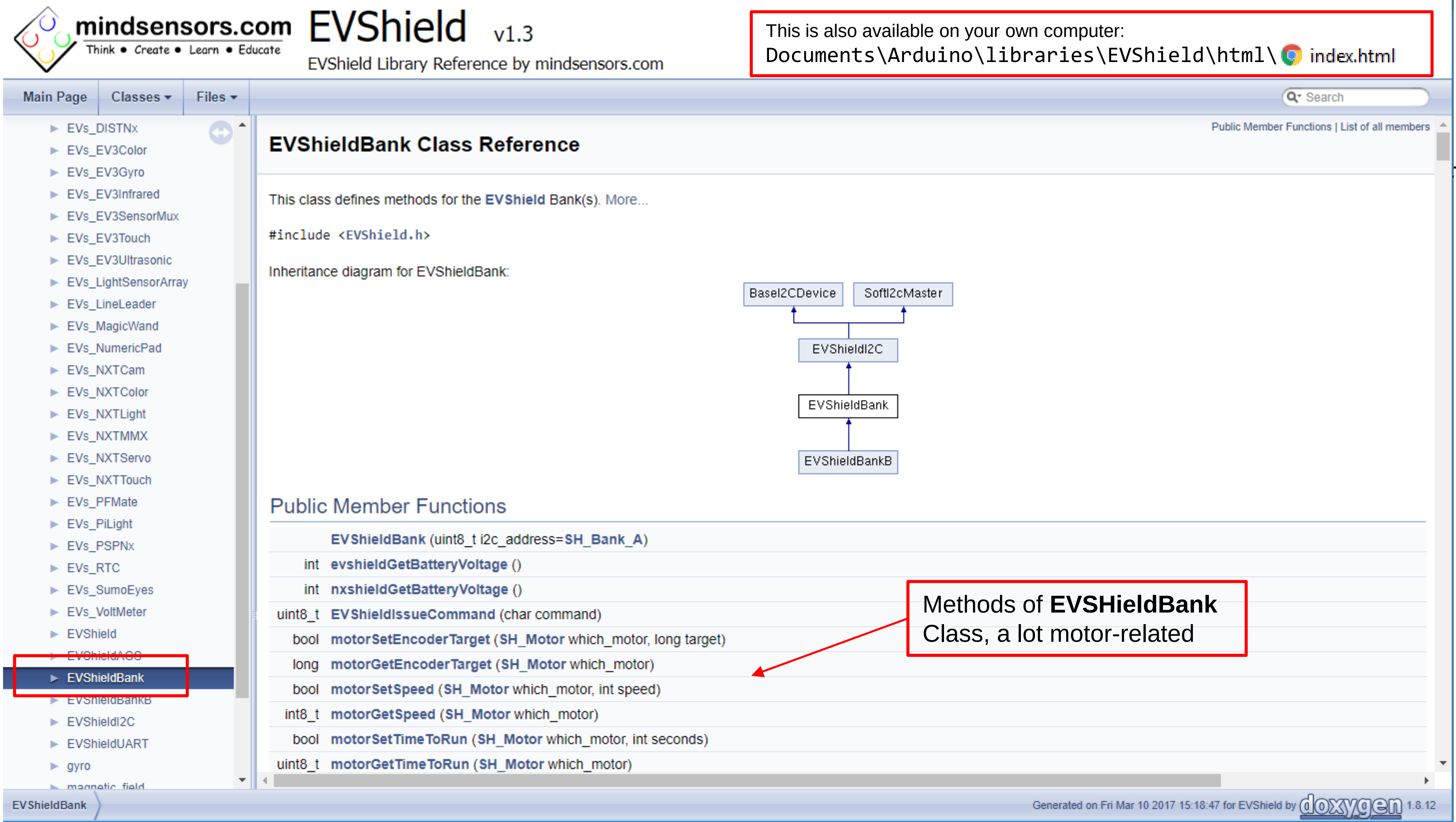
EVShield Library Reference (mindsensors.com)

EVShield-Advanced-Development-Guide



- EVShield
  - EVShield Library Reference
    - Introduction
    - Getting Started
    - More Information
    - Installation Instructions
  - Classes
    - Class List
      - accl
      - BaseI2CDevice
      - cmps
      - color
      - EVs_AbsoluteIMU
      - EVs_AngleSensor
      - EVs_CurrentMeter

scroll down until “**EVShieldBank**”, click that



This is also available on your own computer:
Documents\Arduino\libraries\EVShield\html\ index.html

mindsensors.com
Think • Create • Learn • Educate

EVShield v1.3

EVShield Library Reference by mindsensors.com

Main Page | Classes | Files | Search

- EVs_DISTNx
- EVs_EV3Color
- EVs_EV3Gyro
- EVs_EV3Infrared
- EVs_EV3SensorMux
- EVs_EV3Touch
- EVs_EV3Ultrasonic
- EVs_LightSensorArray
- EVs_LineLeader
- EVs_MagicWand
- EVs_NumericPad
- EVs_NXTCam
- EVs_NXTColor
- EVs_NXTLight
- EVs_NXTMMX
- EVs_NXTServo
- EVs_NXTTouch
- EVs_PFMate
- EVs_PiLight
- EVs_PSPNx
- EVs_RTC
- EVs_SumoEyes
- EVs_VoltMeter
- EVShield
- EVShieldAGS
- EVShieldBank
- EVShieldBankB
- EVShieldI2C
- EVShieldUART
- gyro
- magnetic_field

Public Member Functions | List of all members

EVShieldBank Class Reference

This class defines methods for the **EVShield** Bank(s). More...

```
#include <EVShield.h>
```

Inheritance diagram for EVShieldBank:

Public Member Functions

	EVShieldBank (uint8_t i2c_address=**SH_Bank_A**)
int	**evshieldGetBatteryVoltage** ()
int	**nxshieldGetBatteryVoltage** ()
uint8_t	**EVShieldIssueCommand** (char command)
bool	**motorSetEncoderTarget** (**SH_Motor** which_motor, long target)
long	**motorGetEncoderTarget** (**SH_Motor** which_motor)
bool	**motorSetSpeed** (**SH_Motor** which_motor, int speed)
int8_t	**motorGetSpeed** (**SH_Motor** which_motor)
bool	**motorSetTimeToRun** (**SH_Motor** which_motor, int seconds)
uint8_t	**motorGetTimeToRun** (**SH_Motor** which_motor)

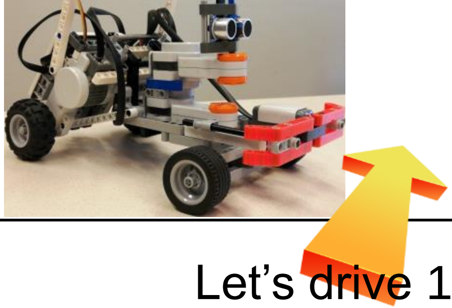
Methods of **EVSHieldBank** Class, a lot motor-related

EVShieldBank

Generated on Fri Mar 10 2017 15:18:47 for EVShield by doxygen 1.8.12

CALCULATE MOTOR DEGREES

TO DRIVE A GIVEN DISTANCE



Let’s drive 1m

$$C_w = D_w \times \pi$$

$$degrees = \frac{distance}{C_w} \times 360$$

Drive a given distance (variable `distance` is parameter)

```
double circumference = WHEEL_DIAM * PI; // PI is-build in definition
unsigned long degrees = (distance / circumference) * 360;

evshield.bank_a.motorRunDegrees(SH_Motor_Both, direction, speed, degrees, SH_Completion_Wait_For, SH_Next_Action_BrakeHold );
```

Run the motor for the calculated amount of degrees

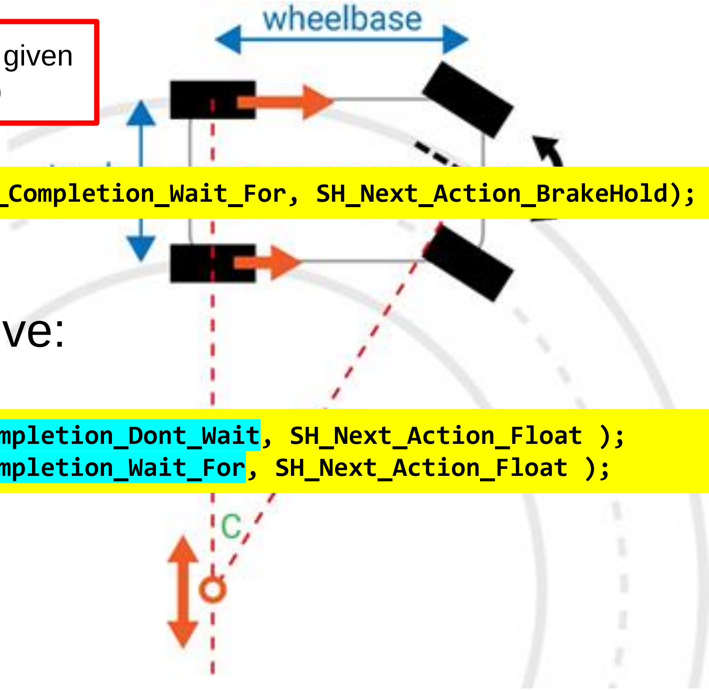
Download example **evshield_drive_1m.ino** or see method **Rover::drive(int distance = 0)** in code of assignment

UNIVERSITY OF TWENTE.

STEERING A CAR

USING DIFFERENTIAL POWER ON DRIVING WHEELS

More in LegoPracticalSession 2



Steer (front wheels):

Turn front wheels with given degrees (e.g. 15)

```
evshield->bank_b.motorRunDegrees(steerMotor, dir, SH_Speed_Slow, abs(degrees), SH_Completion_Wait_For, SH_Next_Action_BrakeHold);
```

At the same time drive using differential drive:

(reduce power to inner wheel, more power to outer wheel)

```
evshield->bank_a.motorRunDegrees(SH_Motor_1, dir, speed * ratio_L, degrees, SH_Completion_Dont_Wait, SH_Next_Action_Float );
evshield->bank_a.motorRunDegrees(SH_Motor_2, dir, speed * ratio_R, degrees, SH_Completion_Wait_For, SH_Next_Action_Float );
```

In this case, commands should run at same time! How? For first command use **`SH_Completion_Dont_Wait`**

See methods **`Rover::steer()`** and **`Rover::differentialDrive()`** in code of assignment

UNIVERSITY OF TWENTE.

DRIVING & STEERING

WHEELED VEHICLES

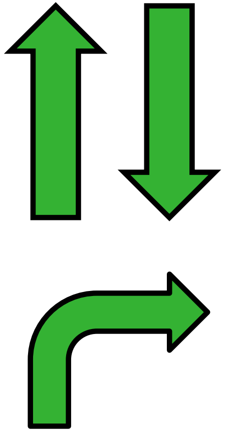
```
class Rover {
```

Declaration in Rover.h, Definition in Rover.cpp

```
// driving related methods:
void reverseDirection();
void drive(int distance = 0);
void stop();
void steer(int angle);
```

Use (method calls)

```
// methods calls:
reverseDirection();
drive(); // drive unlimited (in current direction)
drive(200);
drive(-30);
stop();
turn(90); // turn right
turn(-90); // turn left
```



TRIP DATA

WHEELED VEHICLES

`class Rover {`

Calculate travelled distance:

- Get number of degrees motor has turned (since its reset):

```
evshield->bank_a.motorGetEncoderPosition(SH_Motor_1)
```

- Calculate:

```
double circumference = car_wheel_diam * PI;
int d = evshield->bank_a.motorGetEncoderPosition(SH_Motor_1);
Serial.print(F("Travelled distance: "));
Serial.println(abs(d)/360*circumference/100);
```

Blue part: number of rotations

Divide by 100 to get meters

what will go wrong here?
if we drive 1m forward, then 1m backward, travelled distance will be 0…

…what would be a solution?

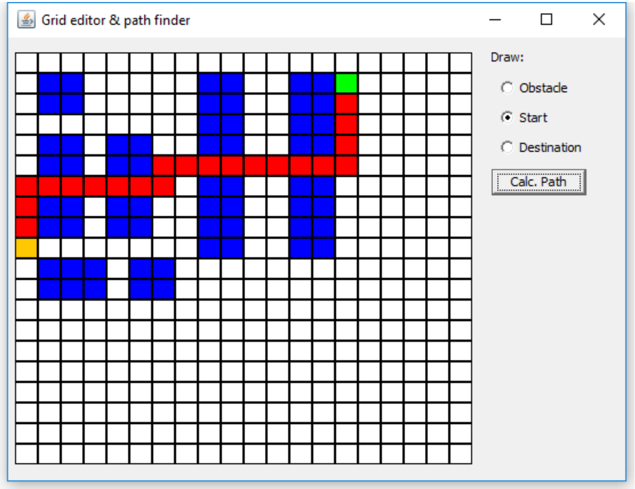
SMART VEHICLES… PATH-FINDING?

More on this topic in session #6. Assignment 6 will be 'hands-on' path-finding.

Reading:

- Wikipedia: Pathfinding
- Wikipedia: A* search algorithm
- Introduction to A* pathfinding
- Path finding using A* Algorithm: Java Example
- Draw a grid-based representation of a room
(solution at bottom using PixelPainter class)

Assignment 6 will be a combination of these 2



Grid editor & path finder

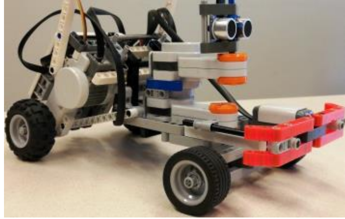
Draw:

○ Obstacle

● Start

○ Destination

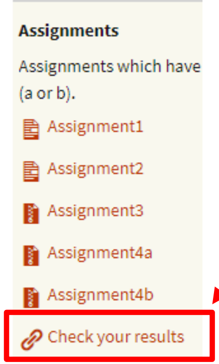
ASSIGNMENT #4



This afternoon: teacher available for help with project & catch-up session if you missed a practical session or what to finish an assignment

- “Program the Rover car” (or any other model)
- Read intro carefully, review all points in the checklist
- Sit together with project-group, do assignment in couples of 2 students!

Checkpoint



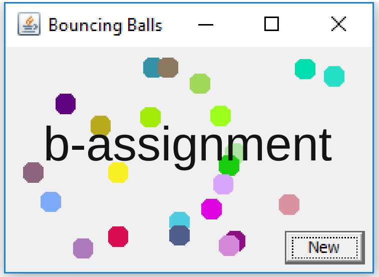
Assignments

Assignments which have (a or b).

- Assignment1
- Assignment2
- Assignment3
- Assignment4a
- Assignment4b
- Check your results

- Check assignment results

a or b?
Assignments which have 'a’ / 'b' variant:
a-version = Arduino programming
b-version = Java programming (no Lego/Arduino required).
You have to do only one (a or b).



Slides, assignments etc @ vanslooten.com/appdev