Last Time

- Variables
  - Different types (`int, float, double, boolean, String`)
  - Can assign values to them, e.g. `int angle = 270;`
  - Can perform calculations (`+, -, *. /, %`) with them
  - Can be constants (`static final …`)
- Functions (or methods)
  - Refer to a well-defined subtask of the overall task
  - Function header (return type, name, input list)
  - Function body (code that specifies how function works)
  - Use comments to describe the function and its input/output

Tasks

1. Create a robot that can drive forward
2. Create a robot that can rotate in place
3. Create a robot that can drive forward *by exactly x cm*
4. Create a robot that can rotate in place *by exactly y degree*

Step 1: Robot Design

- Need a robot capable of performing our tasks; as simple as possible yet as capable as necessary
- Need a machine with wheels
  - needs to be able to drive forwards & backwards
  - needs to be able to turn “in place”

Suggestion: 4-wheeled car-like robot

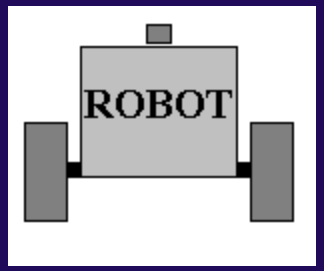
  - cannot turn “in place”

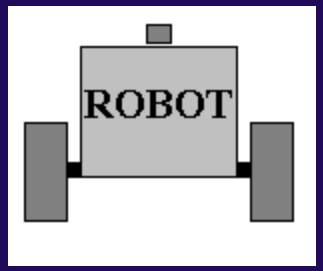
Solution: *Differential Drive Robot*

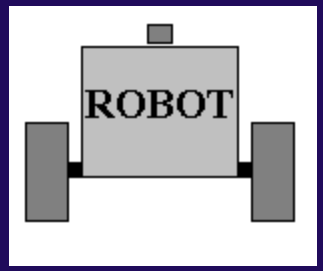
Differential Drive Robot

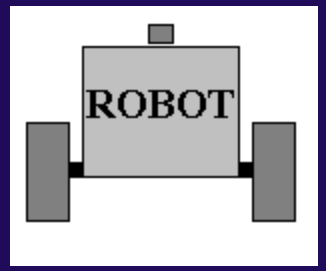
- Robot with 2 wheels on a common axis
- Each wheel can be independently powered either forward or backward

... might need a 3rd unpowered wheel for stability









Step 2: The Model

real model

Differential Drive Robot and Model

Task 1

Make our robot drive forward

```
public static void main(String args[])
{
        // variable angle tells how much to rotate each motor in degrees
        int angle = 720;
        // now engaging both motors
        Motor.A.rotate(angle, true);
        Motor.C.rotate(angle);
}
```

State Variables

State variables are defined *before* the main method

- They describe the *state* of the robot, i.e. everything that the robot *has*
- They remain valid throughout the program (regular variables are only valid inside the function in which they are defined)
- Usage:

```
static int axlelength = 11.3;
```

Task 1: Drive forward

```
public class Driver
{
    // state variables (are frequently constant)
    static NXTRegulatedMotor leftMotor = Motor.A;
    static NXTRegulatedMotor rightMotor = Motor.C;

    static double wheelRadius = 2.8
    static double axleWidth = 14.0;

    public static void main(String args[])
    {
        // variable angle tells how much to rotate each motor in degrees
        int angle = 720;
        // now engaging both motors
        leftMotor.rotate(angle, true);
        rightMotor.rotate(angle);
    }
}
```

Task 2: Rotate in Place

```
public class Rotator
{
    static Motor leftMotor = Motor.A;
    static Motor rightMotor = Motor.C;
    static double wheelRadius = 2.5;
    static double axleLength = 13.0;

    public static void main(String args[])
    {
        // variable angle tells how much to rotate each motor in degrees
        int angle = 720;
        // now engaging both motors in opposite directions
        leftMotor.rotate(angle, true);
        rightMotor.rotate(-angle);
    }
}
```

Task 3 (*the math*)

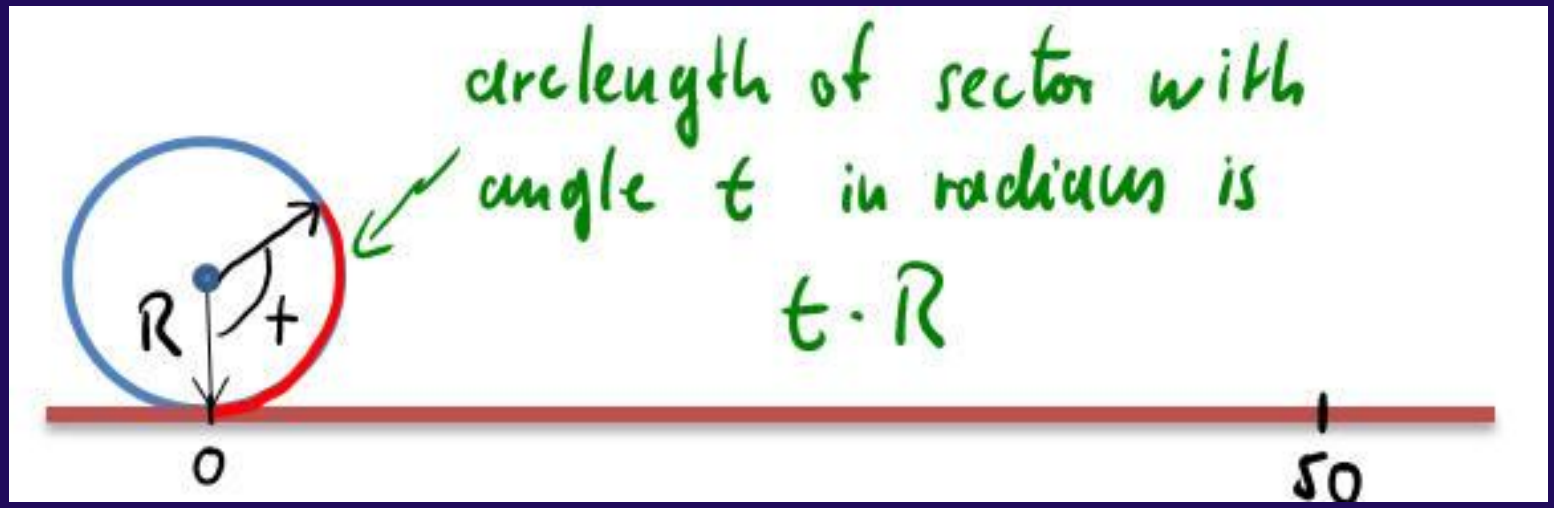
Create a robot that can drive forward *x* cm

Model

Task 3 (*the math*)

Create a robot that can drive forward *x* cm

Model



Task 3 (*the math*)

- If *angleR* is an angle in radians and *radius* is the radius of a circle, then:

```
distance = angleR * radius
```

- If *angleD* is an angle in degrees:

```
angleD/360 = angleR/2 Pi
```

- Combining those equations:

```
distance = angleD / 180 * Pi * radius
```

- Or equivalently:

```
distance / radius * 180 / Pi = angleD
```