What you need to know to program Kara for the exercises on this Sheet:

Actions

You program Kara, you write a list of actions that will be processed in order. Available actions:

move();	move one step in the current direction
turnLeft();	change direction 90° to the left
turnRight();	change direction 90° to the right
putLeaf();	put a leaf on the current position
removeLeaf();	remove leaf from the current position

To use them, you just put them one after another into your program.

Sensors and the If-Statement

Kara can react to its environment using sensors.
Available sensors:

onLeaf()	true if the current tile contains a leaf
treeFront()	true if the next tile in the current direction contains a tree
treeLeft()	true if tile to the left – again relative to current direction contains a tree.
treeRight()	Same for treeRight, only to the right.
mushroomFront()	true if the next tile in the current direction contains a mushroom.

To use the sensors, you need a conditional statement, the Java if-statement (the else part can be omitted)

```
if (onLeaf()) {
	// put the commands that should be executed if onLeaf() is true here
} else {
	// put the commands that should be executed otherwise here
}
```

Example:

```
if (onLeaf()){
	removeLeaf();
}
```

Defining own actions for Kara (Methods)

If you need to do the same set of actions repeatedly, you can define an own action for Kara by defining a Java method:
This is especially useful in Exercise 6, where you need to walk around a tree several times. So you would define a method named "walkAroundTree" in Java like so:

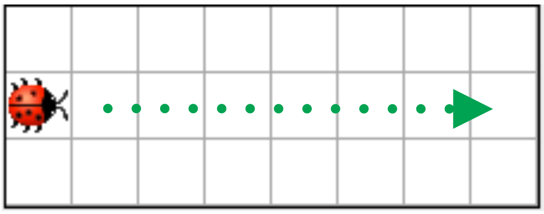
```
public void walkAroundTree(){
	//put  instructions to walk around tree here
}
```

You can then use the method as any other action listed above:

```
walkAroundTree();
move();
walkAroundTree();
move();
move();
….
```

Exercise 1: Walk across the lawn

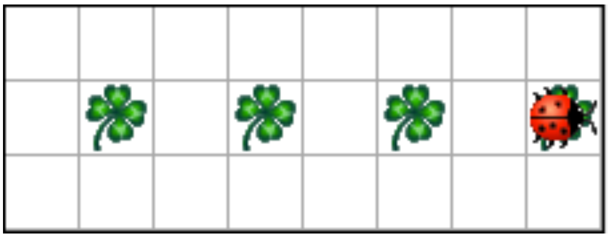
Make Kara walk across the lawn.



Test Area

Program (to test, but the red meeple in front of the instruction that will be executed next.)

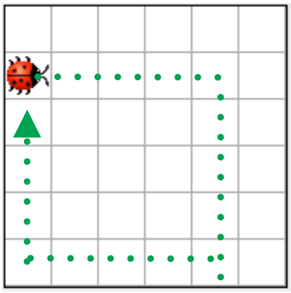
Exercise 2: Walk across the lawn and drop a leaf on every second tile.



Test Area

Program (again, use the red meeple to mark your position in the program)

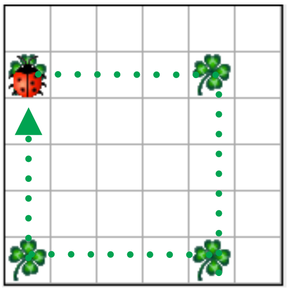
Exercise 3: Walk in a square.



Test Area

Program (again, use the red meeple to mark your position in the program)

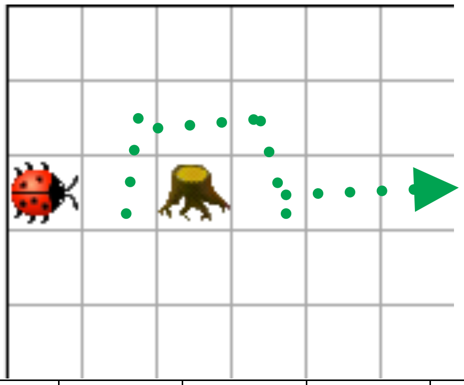
Exercise 4: Walk in a Square, and drop a leaf in every corner.



Test Area

Program (again, use the red meeple to mark your position in the program)

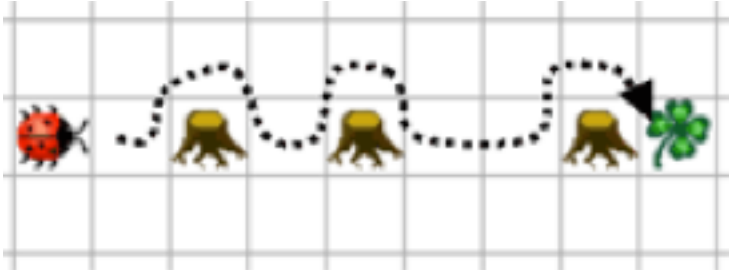
Exercise 5: Walk around a Tree.



Test Area

Program (again, use the red meeple to mark your position in the program)

Exercise 6: Walk through the forest and pick up a leaf.



Test Area – you can use green or brown meeples for the trees.

		🌳		🌳			🌳		

Program (again, use the red meeple to mark your position in the program)

Exercise 7: Same as Exercise 6, but define a method to walk around the tree.

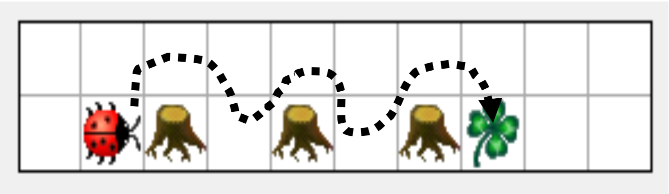
Test Area – you can use green or brown meeples for the trees.

		🌳		🌳			🌳		

Program (again, use the red meeple to mark your position in the program)

	Program		Method to walk around a tree

Exercise 8: What happens if the forrest changes?



Test Area – you can use green or brown meeples for the trees.

Program (again, use the red meeple to mark your position in the program. You can just reuse your method from Excercise 7.)
