CS 11114

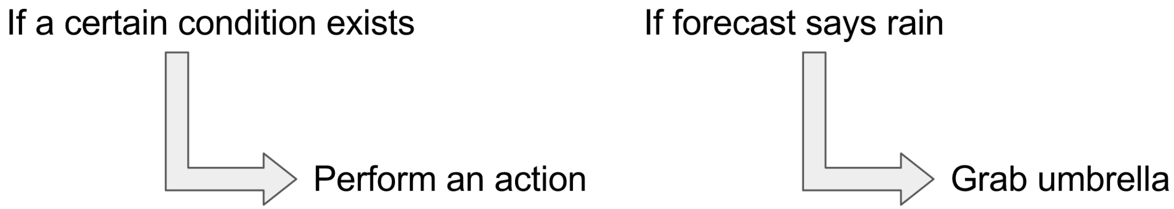
Introduction to Software Design

Spring 2017 - Michael Irwin

Program #1

Making Decisions

- We make decisions everyday based on input
  - If the forecast says it’s going to rain, I will grab my umbrella
  - If my laptop is low on power, I will plug it in
- In simple cases, there's a pattern to the decision...



If a certain condition exists

Perform an action

If forecast says rain

Grab umbrella

Representing Decisions in Code

- In code, we use an **if statement**
- The conditional statements must evaluate something to be **true** or **false**
  - Boolean methods
  - Boolean expressions

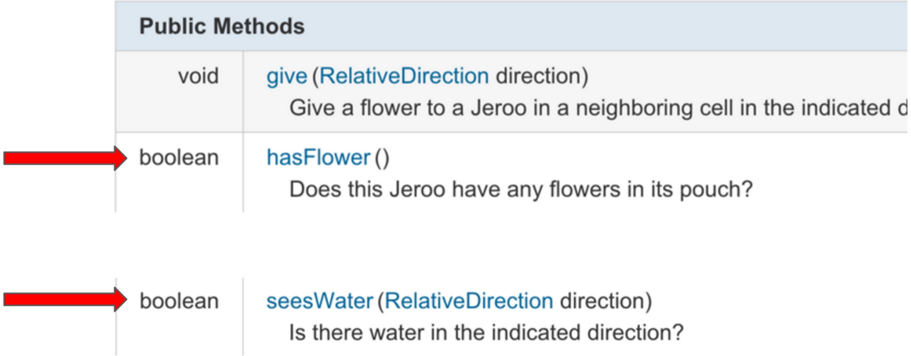
```
if (<conditional statement>) {
  // actions go here
}
```

```
if (willRainToday()) {
  grabUmbrella();
}
```

Boolean Methods

- A **boolean** is a primitive type that has two values - **true** or **false**
- A **boolean method** is a method that returns a boolean

Jeroo class documentation

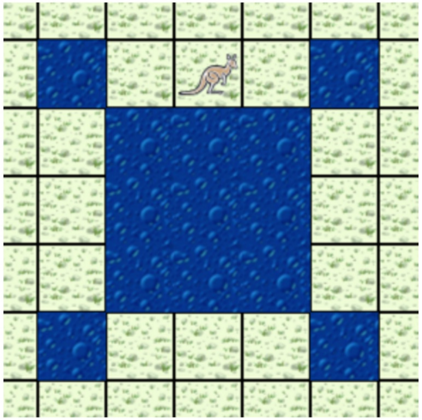


Public Methods	
void	give (RelativeDirection direction) Give a flower to a Jeroo in a neighboring cell in the indicated d
boolean	hasFlower () Does this Jeroo have any flowers in its pouch?
boolean	seesWater (RelativeDirection direction) Is there water in the indicated direction?

Boolean Method Examples

- What's the return value for the method below?

```
jeroo.seesWater(AHEAD);
```

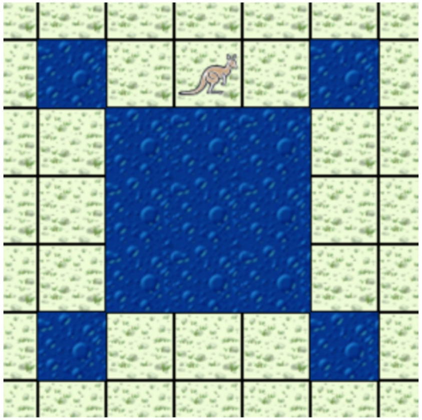


Boolean Method Examples

- What's the return value for the method below?

```
jeroo.seesWater(AHEAD);
```

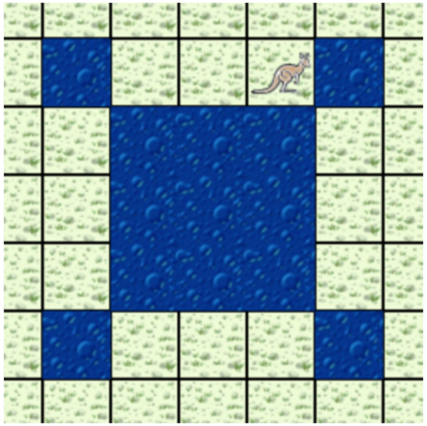
- The return would be `false` because there is no water ahead of the Jeroo



Boolean Method Examples, #2

- How about now?

```
jeroo.seesWater(AHEAD);
```

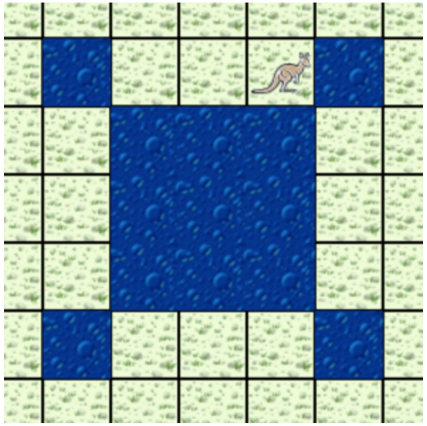


Boolean Method Examples, #2

- How about now?

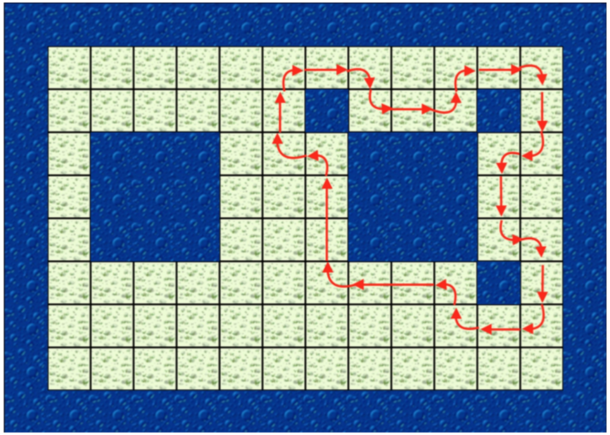
```
jeroo.seesWater(AHEAD);
```

- The return would be `true` because there IS water ahead of the Jeroo



Our Scenario

- The neighboring enemy destroyed one of our turrets!
- Need to make our `TurretJeroo` (or bot) more flexible so it can patrol the most optimized route
- What's our algorithm?



The Algorithm

- If we don’t see water ahead...
  do a normal corner turn
- If we do see water ahead...
  do our custom turret turn

```
public void turnCorner() {
    if (!seesWater(AHEAD)) {
        super.turnCorner();
    }
    if (seesWater(AHEAD)) {
        this.turn(LEFT);

        super.turnCorner();
        super.turnCorner();
        super.turnCorner();

        this.turn(LEFT);
    }
}
```

Refining the Algorithm

- Using an `if-else` statement, we can make the code even cleaner

```
if (<conditional statement>) {
  // actions if condition is true
}
else {
  // actions if condition is false
}
```

```
public void turnCorner() {
    if (seesWater(AHEAD)) {
        this.turn(LEFT);

        super.turnCorner();
        super.turnCorner();
        super.turnCorner();

        this.turn(LEFT);
    }
    else {
        super.turnCorner();
    }
}
```