Getting Started

LightBot Game

Micro-world - simulated environment using grid using cartesian coordinates (4th quadrant)

program - set of instructions for computer

source code - written in some language, needs to be translated into machine language

compiler - translates a whole program, not necessarily into machine language

interpreter - translates a program line by line into machine language

algorithm - plan for solving specific problem

object - an item / concept relevant to the problem being solved

behavior - action that can take or a task that it can perform in response to a request

method - collection of statements to describe a specific behavior

precondition - something assumed to be true before a method is executed

postcondition - something assumed to be true after a method is executed

message (invoke method) - request for object to perform a task

class - family of objects that all understand the same methods

instantiation - creating a new object

declaration - introducing a new name

constructor - special method used to create new objects

receiver - object method is being called on

```java
Lightbot name = new Lightbot();
//Lightbot name is the declaration
//new Lightbot(); is the instantiation
// Lightbot() is the constructor
```

LightBot Method Chart

Method	What Happens
`move()`	The robot moves forward one square (if it can)
`turnRight()`	The robot turns 90 degrees to its right (clockwise)
`turnLeft()`	The robot turns 90 degrees to its left (counterclockwise)
`jump()`	The robot moves forward by jumping up one block higher, or by jumping down one or more blocks lower (if it can)
`turnLightOn()`	The robot lights up the blue tile, if it is standing on one
`f1()`	The robot carries out whatever sequence of actions you have defined for the method `f1`

`f2()`	The robot carries out whatever sequence of actions you have defined for the method `f1`

Hierarchy

1. Class
2. Objects
3. Methods
4. Statements (Actions)