



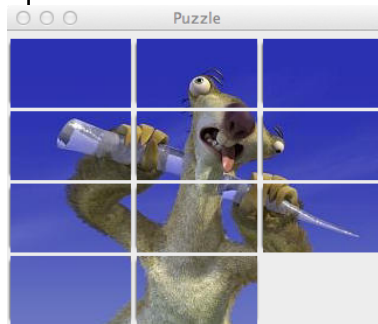
Maze Program

TOPICS

- Graphical Programming
- Using Classes (Objects)
- Multiple Files (Eclipse)
- Maze Logistics



Graphical Programming (1)



CS 160, Spring Semester 2014

2



Graphical Programming (2)

- No, we're not going to show the code for the `Puzzle` program! (yet)
- 150 lines of Java Swing code
- What kinds of things does it do?
 - Set window size, title, and location
 - Setup a frame and panel, add buttons
 - Read a photo and extract parts of it
 - Listen for mouse and keyboard events

CS 160, Spring Semester 2014

3



Graphical Programming (3)

- But, why not have you write code that controls a graphical program?
- Maze program:
 - Move student around a maze
 - ~280 lines of graphical programming
 - You write the main method
 - You instantiate the Maze
 - You control the movement

CS 160, Spring Semester 2014

4



Maze Program: Objectives

- Use objects (classes) developed by someone else
 - Create an instance of a class
 - Call methods on the object
 - For example: graphical programming!
 - Multiple source files
 - Resource and data files



Maze Program: Initial Code

```
// Create maze
String fileName = args[0];
Maze maze = new Maze(fileName);
System.out.println("Maze name: " + fileName);

// Get dimensions
int mazeWidth = maze.getWidth();
int mazeHeight = maze.getHeight();
System.out.println("Maze width: " + mazeWidth);
System.out.println("Maze height: " + mazeHeight);
```



Maze Program: Moving

```
// Move commands
boolean success;
success = maze.moveRight(); // move right
success = maze.moveLeft(); // move left
success = maze.moveUp(); // move up
success = maze.moveDown(); // move down
boolean done;
done = maze.isDone(); // true if done, else false
```

HINT: Must call maze.isDone after every move!

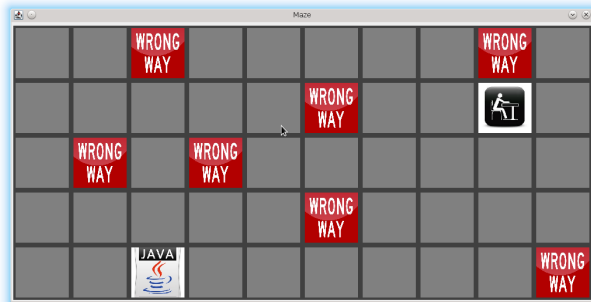


Maze Program: Output

```
Maze name: Maze5.txt
Maze width: 10
Maze height: 5
Moved to row 0, column 1
Moved to row 1, column 1
Moved to row 1, column 2
```



Maze Program: User Interface



Maze Program: Algorithm

- Rules for moving student around maze:
 - Student always starts top left corner
 - Traverse all rows top to bottom in order
 - Left to right on even rows
 - Right to left on odd rows
 - If wrong way, go down, around, up
 - To cover a row or column, how many moves?
 - Must follow algorithm exactly!
 - Try to make your code simple and efficient



Maze Program: Setup

- ~/workspace/P4
 - DoNotPass.jpg
 - Java.jpg
 - Success.jpg
 - Student.jpg (replace!)
 - Maze*.txt
- ~/workspace/P4/src
 - Maze.java
 - P4.java



Associated Exercises

- Recitation R5:
 - Setup files, create maze, move student
 - Put your own photo in Student.jpg
- Assignment P4:
 - Implement algorithm as described
 - Automated testing for several mazes
 - You can make your own maze
 - Not all mazes can be solved by algorithm