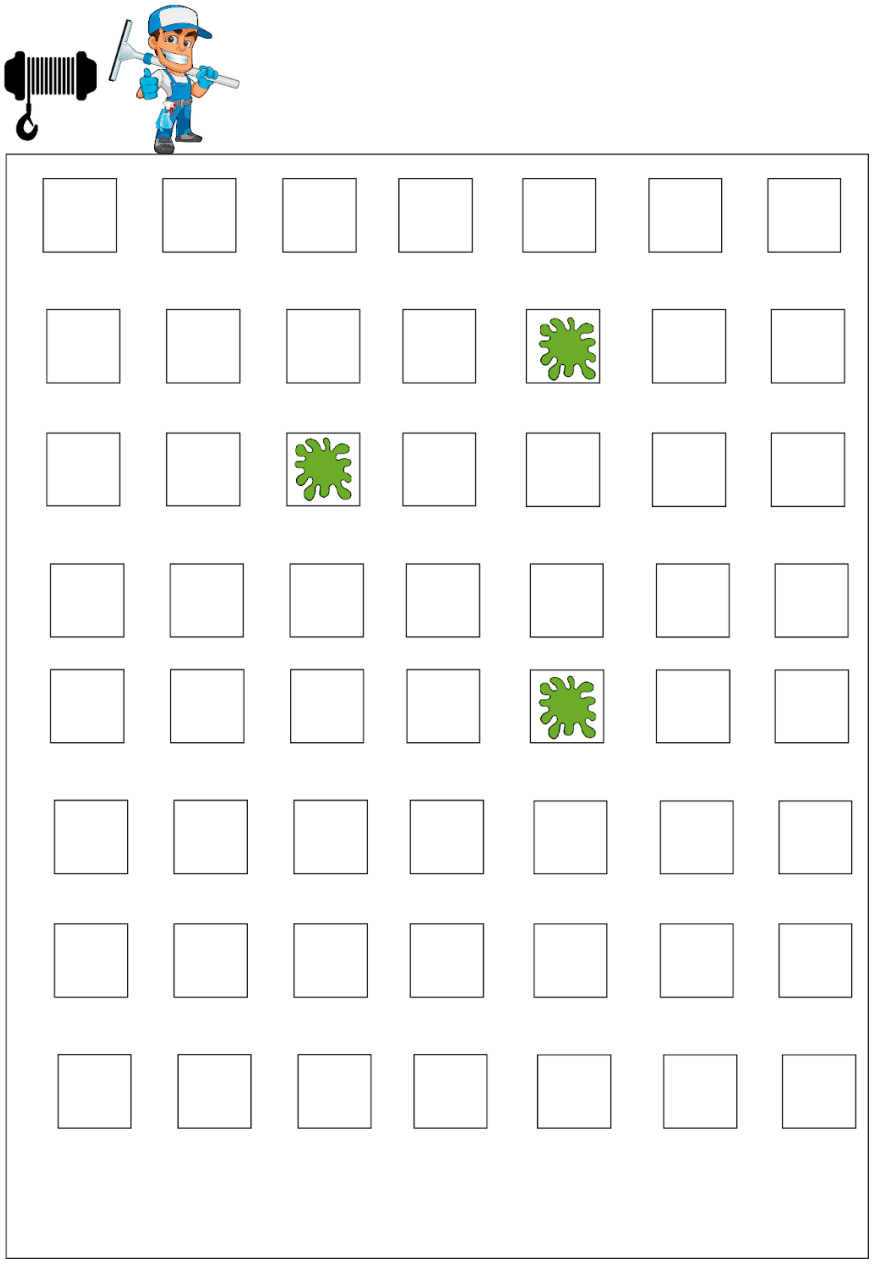
1. Bob is a window cleaner. He has a device installed on the top left corner of the building that he works at that can winch him up and down a building. Using the image below, and assuming that one square is one meter, how many meters does he need to move the winch to the right, and how much cordage does he need to lower to lower his platform so he is about where the dirty windows are?



The question states “about”. This means it does not even have to be exactly where the windows are, and arguably the starting location is about where the stains are. We can

also assume that about means within 3 tiles of all the stains. In this case, he would need to move 2 meters to the right and have 2 meters of cordage

2. Overloading is a very useful tool that exists in Java and in many other programming languages. Essentially, you can give different methods the same name. You can do this with constructors, or mostly anything that can receive input!
    a. What are your options if you want to find the value of a number squared? Which would you choose if you wanted the method to say something to the user?
        i. The options to find the value of a number squared would be using "power(int x)" or "power(int x, int y)" when "y = 2".
        The options if you wanted the method to say something to the user would be "greet(String name)".
    b. What 'Student' would you call if you want to create a student when you don't have any input?
        i. "Student()".
    c. What 'Student' would you call if you had a whole array of data to input into the student object?
        i. "Student(String name, Array[])".
    d. Why do you think this is a capability that Java has? What uses can you imagine for it?
        i. This is a capability that Java has because most languages have arrays to store multiple amounts of data. A use that I can imagine for arrays would be storing multiple amounts of data, organizing that data (ex. Least to greatest character count, least to greatest number, randomize).

Lab

1. Driver: Alan Xiao
   Scribe: Froilan Zarate
    a. Scribe: what image did you use?
        i. roomba.png.
    b. Scribe: write down your expectations before you run the code?
        i. Our expectations before we run the code would display the image of "roomba.png" inside the rectangle we drew.
    c. Scribe: Note any differences between expectations and reality?
        i. The roomba was not in the rectangle.
    d. Driver's Code:

```
public class BasicGraphicViewer
{
    public static void main (String[] args)
    {

        Rectangle r1 = new Rectangle(100,10,20,30);
        r1.draw();

        Picture p1 = new Picture();
```

```
        p1.load("roomba.png"); // Download and save the image in the
cs46a/labs/lab02/lab2a (lab2a folder)
        p1.draw();
        Picture p2 = new Picture();
        p2.load("robot.png");
        p2.translate(0,30);
        p2.draw();

    }
}
```

2. Driver: Froilan Zarate
   Scribe: Alan Xiao
   a. **Scribe: Do you think the robot will appear without additional method calls? Write the answer in your report.**
      i. No, because we did not draw or load the robot yet.

   b. **Scribe: What happens? Describe what appears.**
      i. A list of commands/calls appear as a popup
   c. **Scribe: What is the method called?**
      i. robbie.moveFoward();
   d. What happened to the menu? Do you think this technique could save you time?
      i. The menu popped up without manually looking for it. This technique can definitely be used to save time
   e. Of the following, what methods DON'T exist within the Robot class? Record the letters in your lab document.

      `move()`
      `moveForward()`
      `moveLeft()`
      `turnAround()`
      `turnRight()`
      `turnLeft()`
      `rightHasWall()`
      `leftHasWall()`

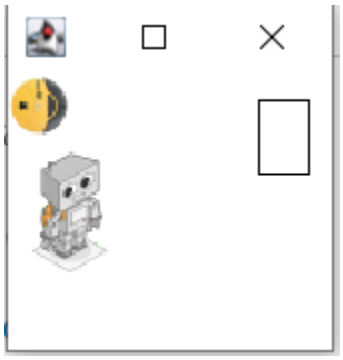
      i. A, c, d, h

   Driver Code:

```
/**
 * Write a description of class RobotDemo here.
 *
 * @author (your name) Froilan Zarate & Alan Xiao
 * @version (a version number or a date) 09-02-2022
 */
public class RobotDemo
```

```
{
    public static void main (String[] args)
    {
        Robot roomba = new Robot(0, 0, "roomba.png");
        roomba.moveForward();
        String name = "Roomba Robot";
        roomba.moveForward();
        roomba.turnRight();
        roomba.moveForward();
        roomba.moveForward();
        roomba.turnRight();
        roomba.moveForward();
        roomba.turnRight();
        roomba.turnRight();
        roomba.moveForward();
        roomba.turnRight();
        roomba.moveForward();
        roomba.moveForward();
        roomba.turnRight();
        roomba.moveForward();
        roomba.moveForward();
    }
}
```

Part1:



Part 2: