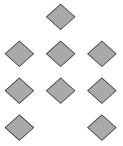
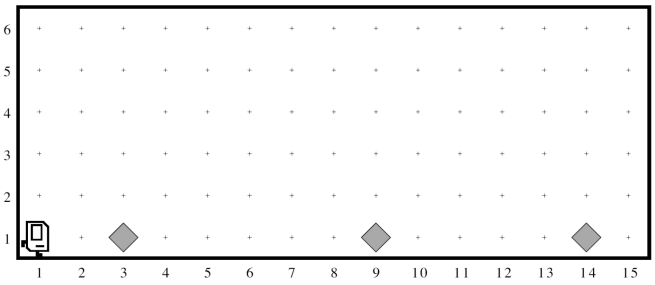
CS 106A Section 1 Handout (Week 2)

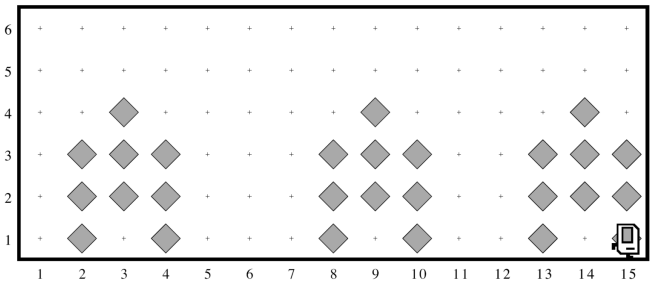
Karel

1. **United Nations Karel**. The United Nations is using Karel to help repair a flood-damaged area. Each robot begins at the west end of a street, and along the street are beepers. Each beeper represents a pile of debris. You should pick up the debris and build a house, centered where the debris was. The houses should look like the figure to the right.



Here is an example of how your program should run. The diagram on the left is the world Karel starts in; by the end of your program, Karel should create the beeper arrangement on the right. Karel must end up facing east at the southeast corner of the world. Moreover, Karel should not run into a wall if it builds a house that extends into that final corner. Your solution should work for a world of any size.

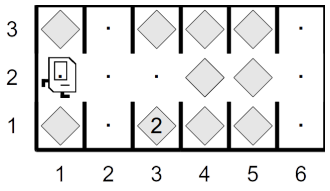


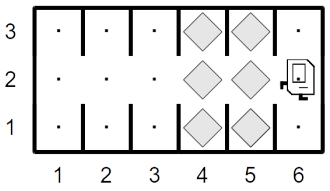


Karel may assume the following facts about the world:

- Karel starts off facing east at (1, 1) with an infinite number beepers in its beeper bag.
- The initial beepers are spaced so there is room to build houses without overlapping or hitting walls.

2. **Karel Defends Democracy**. The 2000 Presidential Elections were plagued by the hanging-chad problem. To vote, voters punched columns out of a paper ballot, but if they only punched partially, the column was left hanging. Luckily, Karel is here to save us!





A ballot consists of a series of columns. Each column corresponds to a column a voter could punch out. Karel starts on the left of a ballot and should progress through each column. If a column contains a beeper in the center row (row 2), the voter did not intend to vote on that column, and Karel should move to the next column. However, if a column contains no beeper in the center row, Karel must make sure that there is no hanging-chad. In other words, Karel should check the corners above and below to make sure all beepers in that column were removed. A corner may contain any number of beepers. Karel must finish execution facing east at the rightmost edge of the ballot. For example, if we start out with the ballot on the top, your program should create the ballot on the bottom.

Karel may assume the following facts about the world:

- Every column is exactly three spaces tall.
- The world can consist of any number of columns.
- The ballot is completely made from punchable columns, with no buffer to the left or right.
- Karel starts facing east on the leftmost column in row 2.
- Karel can travel along row 2 without hitting a wall.

Your program should work for a ballot with any number of columns.

Java Fundamentals

3. **Evaluating Expressions.** Evaluate the following; use the proper type (i.e. 7.0 for a double instead of 7).

```
a) 3 * 4 + 2 * 3
b) 2 + 19 % 5 - (11 * (5 / 2))
c) 2 + 2.0
d) 26 % 10 % 4 * 3
e) 813 % 100 / 3 + 2.4
f) 4 * 3 / 8 + 2.5 * 2
g) (5 * 7.0 / 2 - 2.5) / 5 * 2
h) 41 % 7 * 3 / 5 + 5 / 2 * 2.5
i) 89 % 10 / 4 * 2.0 / 5 + (1.5 + 1.0 / 2) * 2
```

4. **Nested For Loops.** What does the code below print out? (Note: unlike `println`, `print` does not add a newline).

```
for (int i = 1; i <= 10; i++) {
    for (int j = 1; j <= 10 - i; j++) {
        print(" ");
    }
    for (int j = 1; j <= 2 * i - 1; j++) {
        print("*");
    }
    println();
}
```

5. **InchesToCentimeters** *(Ex. 3.1, p91).* Write an interactive console program that prompts the user to read in two input values: a number of feet, followed on a separate line by a number of inches. The program should convert this amount to centimeters. Here is a sample run of the program (user input is underlined):

```
This program converts feet and inches to centimeters.
Enter number of feet: 5
Enter number of inches: 11
5ft 11in = 180.34cm
```

6. **CalculateLine** Write an interactive console program that calculates y coordinates on a line. First, it prompts the user for a slope, *m*, and an intercept term, *b* (remember that a line has an equation of the form $y = mx+b$). Then, the program prompts the user for *x* values until the user enters a **-1**. For each entered number, print the *y* value on that line for that entered *x* value. Here is a sample run of the program (user input is underlined):

```
This program calculates y coordinates for a line.
Enter slope (m): 2
Enter intercept (b): 4
Enter x: 5
f(5) = 14
Enter x: 1
f(1) = 6
Enter x: -1
```

Note: To give you extra practice problems throughout the quarter, we intentionally include more problems on each section handout than can be covered in one 50-minute section.

This document and its contents are copyright © Stanford University. All rights reserved.