Nick Bowman, Sonja Johnson-Yu, Kylie Jue
CS106AP

Section #1
June 26, 2019

Section Handout #1: Karel the Robot

This week in section, your first priority is to meet your section leader and to discover what sections in CS106AP are all about. Your section leader will therefore spend the first part of this week's session on introductions and on telling you the things you need to know. Afterwards, they will move on to cover some of the important material from class in a setting that is small enough for you to go over practice problems and ask questions. This week, your goal is to solve Karel problems that involve decomposition, functions, and control flow.

1. United Nations Karel

As part of its plans to help reconstruct infrastructure worldwide, the United Nations—that's right, the UN is using Karel—established a new program with the mission of dispatching house-building robots to repair flood-damaged areas. Your job is to program those robots.

Each robot begins at the west end of a street that might look like this:

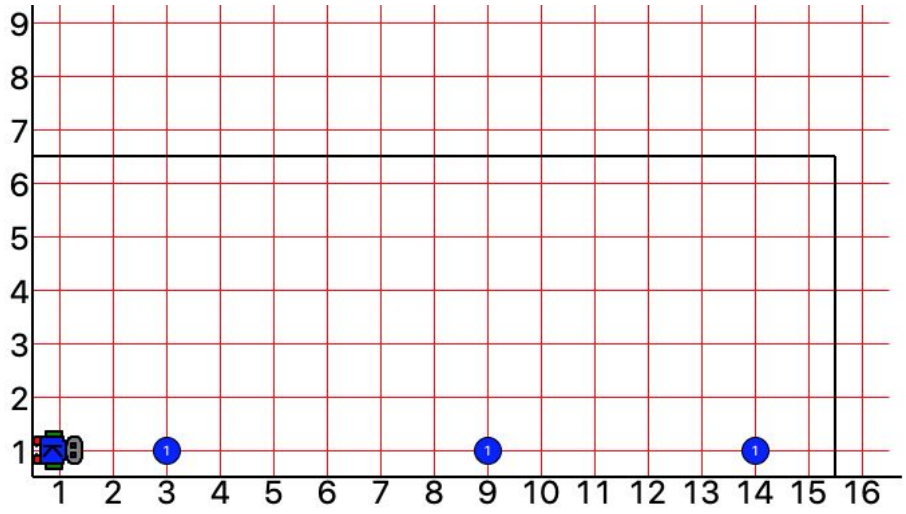
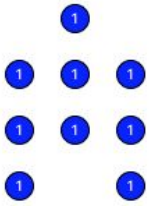


Figure 1: The starting state for United Nations Karel, where each beeper represents a pile of debris.



Each beeper in the figure represents a pile of debris. Karel's job is to walk along the street and build a new house in the places marked by each beeper. Those houses also need to be raised on stilts to avoid damage from the next flood. Each house, in fact, should look exactly like the picture to the right:

The new house should be centered at the column where the debris was left, which means that the first house in Figure 1 would be constructed with its left edge

along 2nd Avenue. At the end of the run, Karel should be at the east end of the street having created a set of houses as shown in Figure 2.

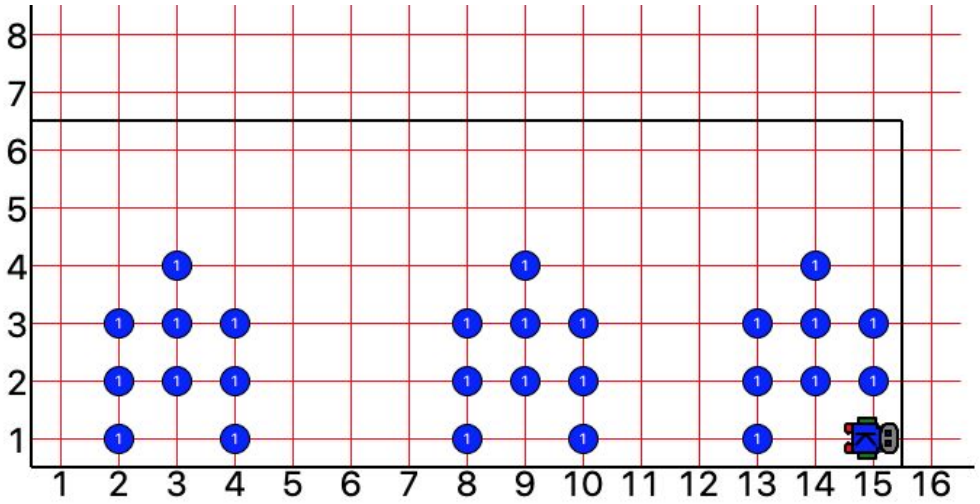


Figure 2: The end state for Karel given the beginning state shown in Figure 1.

Keep in mind the following information about the world:

- Karel starts facing east at (1, 1).
- The beepers indicating the positions at which houses should be built will be spaced so that there is room to build the houses without overlapping walls.
- 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.

Write a program to implement the United Nations Karel project. Remember that your program should work for any world that meets the above conditions.

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 the day!

In Karel's world, a ballot consists of a series of columns that a voter can "punch out." Karel starts on the left of a ballot and should progress through each column. In this world, a lack of beepers represents ballot entries that have been punched out (i.e. holes in the paper). If a column contains a beeper in the center row, the voter **did not intend** to vote in that column, and Karel should move on to the next column. No columns will have punch outs on the top or bottom rows if the middle row is not punched out.

However, if a column contains **no** beeper in the center row, Karel must make sure that there is no hanging chad (i.e. hanging pieces of paper). In other words, Karel should check the corners above and below and remove any beepers. A corner may contain **any number** of beepers. Karel must finish facing east at the rightmost edge of the ballot.

An example initial world is shown on the left in Figure 3. The world on the right shows what Karel's final world should look like (given the initial world on the left).

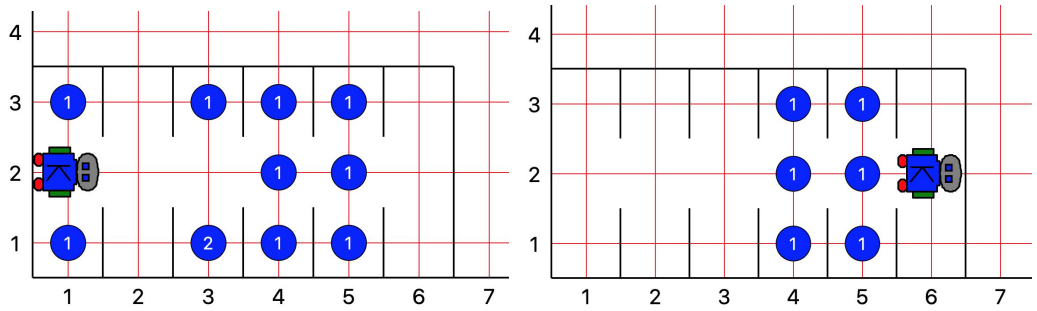


Figure 3: An example starting world (left) and the resulting end world (right) for Karel Defends Democracy.

Keep in mind the following information about the world:

- Karel starts facing east at (1, 2).
- Karel must end up facing east at the end of 2nd Street.
- The world consists of an arbitrary number of 3-height columns only; Karel can travel along the middle row without hitting a wall.

Your program should work for any world that meets the above conditions.