

Chapter 1

A Gentle Introduction

1. In your own words, explain the meaning and purpose of a *programming microworld*.

A *microworld* is a simplified programming environment that allows students to learn the fundamental principles of programming without being overwhelmed by the details of a full-scale language.

2. Who created the Karel microworld?

Richard Pattis

3. What is the etymology of the name *Karel*?

Karel the Robot is named in honor of Karel Čapek whose play *R.U.R.* (*Rossum's Universal Robots*) introduced the word *robot* into English.

4. Define each of the following aspects of Karel's world: *street*, *avenue*, *corner*, *wall*, and *beeper*.

A *street* is an east-west sequence of corners running horizontally.

An *avenue* is a north-south sequence of corners running vertically.

A *corner* is the intersection of a street and an avenue.

A *wall* is a barrier between corners that blocks Karel's movement.

A *beeper* is an object that Karel can pick up and put down on a corner, which Pattis describes as a "plastic cone that emits a beeping sound."

5. What are the four predefined Karel functions?

**move
turnLeft
pickBeeper
putBeeper**

6. What are the two functions included in the Karel library named `turns`?

**turnRight
turnAround**

7. What is the meant by the strategy of *stepwise refinement*?

***Stepwise refinement* is the process of solving problems by starting with the problem as a whole and then breaking that problem into smaller subproblems, which are in turn broken down into successively smaller subproblems, if necessary.**

8. What control statement do you use to execute statements only if some condition applies? What are the two forms of this statement?

The two forms of the `if` statement are illustrated in the following syntax boxes:

```
if (condition) {  
    statements  
}
```

```
if (condition) {  
    statements  
} else {  
    statements  
}
```

9. What two statements does Karel offer for repeating a group of statements?

The `repeat` statement and the `for` statement, as follows:

```
repeat (count) {  
    statements  
}
```

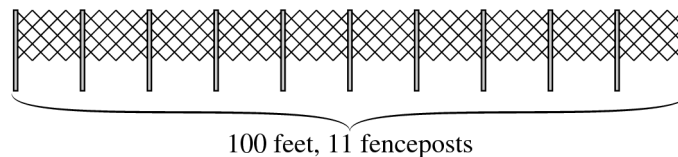
```
while (condition) {  
    statements  
}
```

10. What condition would you use to test whether Karel can move forward from its current position? What condition would you use to test whether there are any beepers on the current corner?

The `frontIsClear` and the `beepersPresent` conditions, respectively

11. What is a *fencepost error*?

A *fencepost error* occurs when a programmer fails to take account of the fact that the number of intervals bounded by a set of dividers is one less than the number of those dividers. This situation is illustrated by the following diagram, which illustrates the fencepost metaphor:



12. What are *preconditions* and *postconditions*?

A *precondition* is a true-or-false statement that is true before executing a function; a *postcondition* is true after executing a function.

13. The `collectLineOfBeepers` function in Figure 1-7 includes an `if` statement that checks the `frontIsClear` condition before moving. Why is it important to make this test?

This statement is necessary to ensure that Karel does not move forward if a line of beeper extends all the way to a wall.