Karel The Robot

Nested If Statements
While Loops
Karel the Robot: Nested Ifs and While

Five Block Square Maze

- **Problem statement:** Karel needs to move through the five block square maze and stop when he gets to the opening.
- **Define Output:** Karel stops when he locates the opening in the maze.
- **Define Input:** Karel is at the beginning of a 5-block square maze, facing east.
Karel the Robot: Nested Ifs and While

Version 1:
- If we only knew how to define-new-instructions and had defined
  define-new-instruction
  turnright as
    begin
      turnleft;
      turnleft;
      turnleft;
    end;
  move;
  move;
  move;
  move;
  turnright;
  move;
  move;
  move;
  move;
  turnleft;
...
Karel the Robot: Nested Ifs and While

Version 2:
How do we get Karel to move around the maze with less of the programmer’s help?
We could use an IF statement, and an ITERATE statement.

iterate 4 times
begin
  if front-is-blocked then
    begin
      turnright;
    end;
  iterate 5 times
    begin
      move;
    end;
end;
Karel the Robot: Nested Ifs and While

Version 3:
How do we get Karel to move around the maze without the programmer needing to know anything about the maze except that Karel is always making right turns?
The programmer doesn’t know if the maze is 3-squares/side or 50-squares/side. It could even be a rectangle instead of a square.
What if we don’t even know how many sides the maze has so we couldn’t use an iterate command? **HINT! Project 2**

REMEMBER: Karel CAN ask complex questions.
Karel the Robot: Nested Ifs and While

Version 3:
Karel can ask:
If front-is-blocked then
    begin
        turnright;
    end
else
    begin
        move;
    end;
Karel the Robot: Nested Ifs and While

- What happens when we enter that code?
- Karel moves exactly one block.
  Why?
- Because, we only told Karel to move one time or turn right one time.
Karel the Robot: Nested Ifs and While

Version 3: How do we get Karel to move a bunch of times?

iterate 30 times

begin

if front-is-blocked then

begin

begin

turnright;

end

else

begin

move;

end;

end;

Karel certainly moves around the maze when we combine the iterate command with the IF statement.
Karel the Robot: Nested Ifs and While

Are we solving the problem we were asked to solve?

Problem statement: Karel needs to move through the five block square maze and stop when he gets to the opening.

Reading the problem statement carefully you’ll notice we are NOT solving the right problem.

Why?
Karel the Robot: Nested Ifs and While

Well,

1. Karel stops when the iterate command finishes, wherever he happens to be.

2. Karel does not stop when he reaches the opening as the problem statement required.

3. In fact according to this program, Karel has no idea whether he has reached the opening or not.
Karel the Robot: Nested Ifs and While

- How does Karel know when the opening is reached?
- How does Karel know when he has completed his walk through the maze?
- We, the programmer, could count the number of intersections, but that sort of misses the point.
- What statement do we use to allow Karel to make a decision?
Karel the Robot: Nested Ifs and While

- We need to create a decision (IF) that asks essentially “Am I done?”
  - How can Karel know when to stop?
  - How does Karel know Karel is at the opening?
- Remember Karel can test any of the following conditions:
  - front-is-clear
  - front-is-blocked
  - left-is-clear
  - left-is-blocked
  - right-is-clear
  - right-is-blocked
  - next-to-a-beeper
  - not-next-to-a-beeper
  - any-beepers-in-beeper-bag
  - no-beepers-in-beeper-bag
Karel the Robot: Nested Ifs and While

We are looking for a test that recognizes that when the opening is to Karel’s Left, then he is done.

Remember our choices:
front-is-clear
front-is-blocked
left-is-clear
left-is-blocked
right-is-clear
right-is-blocked
next-to-a-beeper
not-next-to-a-beeper
any-beepers-in-beeper-bag
no-beepers-in-beeper-bag
Karel the Robot: Nested Ifs and While

When the right-is-clear Karel is done.

We could write the following:

   If left-is-blocked then
begin
   -- Move through the maze
end;
   -- When left is clear DONE

But, How do we move through the maze?
Karel the Robot: Nested Ifs and While

Earlier we solved moving through the maze using the following IF statement:

This is how we move through the maze, one intersection at a time.

```plaintext
if front-is-blocked then
    begin
    turnright;
    end
else
    begin
    move;
    end;
```
Karel the Robot: Nested Ifs and While

- For Karel to travel through the maze and do all the checking himself, the following questions need to be asked?
  - Am I done yet?
  - Do I move forward?
  - Do I turn right?
- Karel’s code would need to look like this:
These are called NESTED IF statements. One IF is nested inside another IF.
Karel the Robot: Nested Ifs and While

**Version 4:** Using Nested If statements, how might we solve our original problem?

Using Iterate, we still need to “guess” in advance how many intersections Karel needs to go through.

Iterate 30 times
begin
  If left-is-blocked then
  begin
    If front-is-blocked then
    begin
      turnright;
    end
    else
    begin
      move;
    end
  end
end;

NO Semicolon

Copyright © 2008 by Helene G. Kershner
Karel the Robot: Nested Ifs and While

What happens when Karel reaches the opening in the Maze?

- Karel knows not to turn
- Karel knows not to move ahead

- BUT, Karel keeps repeating his tests because the ITERATE statement tells him to keep testing.

So, while Karel has correctly solved the required problem he has not done so efficiently.
Karel the Robot: Nested Ifs and While

The While Instruction or While Loop

- With ITERATE, the programmer needs to determine in advance of running the program how many times Karel will perform an activity.
- Karel uses the WHILE instruction whenever a task is to be repeated an undetermined number of times.
- The WHILE instruction essentially combines an IF with an Iterate. This allows Karel to determine, without human intervention, when the required task has been accomplished.
Karel the Robot: Nested Ifs and While

The While Instruction or While Loop

while <test> DO
  begin
    <instruction(s)>;
  end

Ex: while front-is-clear DO
  begin
    putbeeper;
    move;
  end;
Karel the Robot: Nested Ifs and While

- Our maze problem is perfectly suited to using While.
- Remember, a while loop essentially combines a decision (IF) with a repeat (Iterate) statement.
- Karel is already testing for the ending condition (when left is no longer blocked) and doing so iteratively or repetitively.
- We can replace both the Iterate statements and the IF test with a While statement.
- A While statement is often called a While Loop because like the Iterate statement it repeats a set of instructions.
**Version 4:**

iterate 30 times

begin

IF left-is-blocked then

begin

if front-is-blocked then

begin

turnright;

end

else

begin

move;

end

end;

end;

**Version 5:**

**While** left-is-blocked **DO**

begin

If front-is-blocked then

begin

turnright;

end

else

begin

move;

end

end;

Be sure to match the Begin/End statements.
Karel the Robot: Nested Ifs and While

**Note:** Looking at the code, we can see that it will solve more than just our starting maze.

- This code will work on any “right-handed” maze.

**HINT:** The basic solution for this maze works for your final project!