Chapter 3: Variables (Solutions)

Solution to Task 21:

public void act()

{

int count = 0;

while (! treeFront())

{

move();

if (onLeaf())

{

count = count + 1;

}

}

System.out.println("The result is: " + count);

}

Solution to Task 22:

public class MyKara extends Kara

{

boolean goingRight = true;

boolean finished = false;

public void act()

{

// process the first line

processLine();

while (!finished)

{

if (goingRight)

{

if (!treeRight())

{

turnRight();

move();

turnRight();

// we have turned and now go left

goingRight = false;

processLine();

}

else

{

// we are in the bottom right corner

finished = true;

}

}

else

{

if (!treeLeft())

{

turnLeft();

move();

turnLeft();

Continued on next page…

// we have turned and now go right

goingRight = true;

processLine();

}

else

{

// we are in the bottom left corner

finished = true;

}

}

}

}

public void processLine()

{

while (!treeFront())

{

invertField();

move();

}

// invert the last field in the corner

invertField();

}

public void invertField()

{

if (onLeaf())

{

removeLeaf();

}

else

{

putLeaf();

}

}

}

Solution to Task 23:

public class MyKara extends Kara

{

boolean goingRight = true;

boolean finished = false;

boolean havePutLeaf = false;

public void act()

{

// process first row

processRow();

while (!finished)

{

if (goingRight)

{

if (!treeRight())

{

turnRight();

move();

turnRight();

Continued on next page…

// we have turned and now go left

goingRight = false;

processRow();

}

else

{

// we are in the bottom right corner

finished = true;

}

}

else

{

if (!treeLeft())

{

turnLeft();

move();

turnLeft();

// we have turned and now go right

goingRight = true;

processRow();

}

else

{

// we are in the bottom left corner

finished = true;

}

}

}

}

public void processRow()

{

while (!treeFront())

{

processCell();

move();

}

// process the last cell

processCell();

}

public void processCell()

{

if (!havePutLeaf)

{

putLeaf();

}

havePutLeaf = !havePutLeaf;

}

}

Solution to Task 24:

public class MyKara extends Kara

{

int longestRow = 0;

public void act()

{

while (! onLeaf())

{

if (treeFront())

{

countRow();

}

else

{

move();

}

}

System.out.println("The longest tree line is " + longestRow + " trees long");

}

public void countRow()

{

int currentRow = 0;

turnLeft();

while (treeRight())

{

currentRow = currentRow + 1;

move();

}

// go around tree line

turnRight();

move();

move();

turnRight();

// go back down

int i = 0;

while (i < currentRow)

{

move();

i = i + 1;

}

turnLeft();

// test whether the current row is longer

if (currentRow > longestRow)

{

longestRow = currentRow;

}

}

}