Dmitri Stanchevici
Unit 3, Module 2

---

Ex. 2.1

Place 1 in top middle:

| | | | | |
|---|---|---|---|---|
| | | 1 | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Place 2 to the Northeast of 2. Because this position is out of bounds, wrap around and place 2 to the right of 1 but in the bottom row:

| | | | | |
|---|---|---|---|---|
| | | 1 | | |
| | | | | |
| | | | | |
| | | | | |
| | | | 2 | |

Place 3 to the Northeast of 2:

| | | | | |
|---|---|---|---|---|
| | | 1 | | |
| | | | | |
| | | | | |
| | | | | 3 |
| | | | 2 | |

Place 4 to the Northeast of 3. Because this position is out of bounds, wrap around and place 4 above 3, but in the left-most column.

| | | | | |
|---|---|---|---|---|
| | | 1 | | |
| | | | | |
| 4 | | | | |
| | | | | 3 |
| | | | 2 | |

Place 5 to the Northeast of 5:

| | | | | |
|---|---|---|---|---|
| | | 1 | | |
| | 5 | | | |
| 4 | | | | |
| | | | | 3 |
| | | | 2 | |

Place 6 to the Northeast of 5. Because this position is occupied, place 6 directly below 5:

| | | | | |
|---|---|---|---|---|
| | | 1 | | |
| | 5 | | | |
| 4 | 6 | | | |
| | | | | 3 |
| | | | 2 | |

Place 7 to the Northeast of 6:

| | | | | |
|---|---|---|---|---|
| | | 1 | | |
| | 5 | 7 | | |
| 4 | 6 | | | |
| | | | | 3 |
| | | | 2 | |

Place 8 to the Northeast of 7:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | |
| | 5 | 7 | | |
| 4 | 6 | | | |
| | | | | 3 |
| | | | 2 | |

Place 9 to the Northeast of 8. Because this position is out of bounds, wrap around and place 9 to the right of 8 but in the bottom row:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | |
| | 5 | 7 | | |
| 4 | 6 | | | |
| | | | | 3 |
| | | | 2 | 9 |

Place 10 to the Northeast of 9. Because this position is out of bounds, wrap around and place 10 to the right of 9 but in the row above it:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | |
| | 5 | 7 | | |
| 4 | 6 | | | |
| 10 | | | | 3 |
| | | | 2 | 9 |

Place 11 to the Northeast of 10. Because this position is occupied, place 11 right below 10:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | |
| | 5 | 7 | | |
| 4 | 6 | | | |
| 10 | | | | 3 |
| 11 | | | 2 | 9 |

Place 12 to the Northeast of 11:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | |
| | 5 | 7 | | |
| 4 | 6 | | | |
| 10 | 12 | | | 3 |
| 11 | | | 2 | 9 |

Place 13 to the Northeast of 12:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | |
| | 5 | 7 | | |
| 4 | 6 | 13 | | |
| 10 | 12 | | | 3 |
| 11 | | | 2 | 9 |

Place 14 to the Northeast of 13:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | |
| | 5 | 7 | 14 | |
| 4 | 6 | 13 | | |
| 10 | 12 | | | 3 |
| 11 | | | 2 | 9 |

Place 15 to the Northeast of 14:

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | 15 |
| | 5 | 7 | 14 | |
| 4 | 6 | 13 | | |
| 10 | 12 | | | 3 |
| 11 | | | 2 | 9 |

Place 16 to the Northeast of 15. Because this place is out of bounds, we move 16 to the position in the bottom left corner. However, because this position is occupied, we should place it directly below the current number, which 15.

| | | | | |
|---|---|---|---|---|
| | | 1 | 8 | 15 |
| | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | | 22 |
| 10 | 12 | | | 3 |
| 11 | | | 2 | 9 |

Place 17 to the Northeast of 16. Because this place is out of bounds, we move 17 to the right of 16, but then we wrap around to place 17 in the top row:

| | | | | |
|---|---|---|---|---|
| 17 | | 1 | 8 | 15 |
| | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | | |
| 10 | 12 | | | 3 |
| 11 | | | 2 | 9 |

Place 18 to the Northeast of 17. Because this place is out of bounds, we move 18 to the right of 16, but then we wrap around to place 17 in the bottom row:

| | | | | |
|---|---|---|---|---|
| 17 | | 1 | 8 | 15 |
| | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | | |
| 10 | 12 | | | 3 |
| 11 | 18 | | 2 | 9 |

Place 19 to the Northeast of 18:

| | | | | |
|---|---|---|---|---|
| 17 | | 1 | 8 | 15 |
| | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | | |
| 10 | 12 | 19 | | 3 |
| 11 | 18 | | 2 | 9 |

Place 20 to the Northeast of 19:

| | | | | |
|---|---|---|---|---|
| 17 | | 1 | 8 | 15 |
| | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | 20 | |
| 10 | 12 | 19 | | 3 |
| 11 | 18 | | 2 | 9 |

Place 21 to the Northeast of 20. Because this position is occupied, place 21 directly under 20:

| | | | | |
|---|---|---|---|---|
| 17 | | 1 | 8 | 15 |
| | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | 20 | |
| 10 | 12 | 19 | 21 | 3 |
| 11 | 18 | | 2 | 9 |

Place 22 to the Northeast of 21:

| | | | | |
|---|---|---|---|---|
| 17 | | 1 | 8 | 15 |
| | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | 20 | 22 |
| 10 | 12 | 19 | 21 | 3 |
| 11 | 18 | | 2 | 9 |

Place 23 to the Northeast of 22. This position is out of bounds, so move 23 up but to the left all the way to the first column:

| | | | | |
|---|---|---|---|---|
| 17 | | 1 | 8 | 15 |
| 23 | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | 20 | 22 |
| 10 | 12 | 19 | 21 | 3 |
| 11 | 18 | | 2 | 9 |

Place 24 to the Northeast of 23:

| | | | | |
|---|---|---|---|---|
| 17 | 24 | 1 | 8 | 15 |
| 23 | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | 20 | 22 |
| 10 | 12 | 19 | 21 | 3 |
| 11 | 18 | | 2 | 9 |

Place 25 in the remaining spot:

| | | | | |
|---|---|---|---|---|
| 17 | 24 | 1 | 8 | 15 |
| 23 | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | 20 | 22 |
| 10 | 12 | 19 | 21 | 3 |
| 11 | 18 | 25 | 2 | 9 |

Ex. 2.2

For N=3
With row = 0
With col = N/2 = 1

| n | In computeNext(N) | In generateSquare(N) |
|---|---|---|
| 2 | nextRow = N-1=2<br>nextCol = col+1=2 | A[2][2] = 2<br>row = 2<br>col = 2 |
| 3 | nextRow = row-1=1<br>nextCol = 0 | A[1][0] = 3<br>row = 1<br>col = 0 |
| 4 | nextRow = row-1=0<br>nextCol = col+1=1 | A[0][1] != 0<br>nextRow = row+1 = 2<br>nextCol = col = 0<br>A[2][0]=4<br>row=2<br>col=0 |
| 5 | nextRow = row-1=1<br>nextCol = col+1=1 | A[1][1] = 5<br>row=1<br>col=1 |
| 6 | nextRow = row-1=0<br>nextCol = col+1=2 | A[0][2]=6<br>row=0<br>col=2 |

---

Ex. 2.8

Just by reading System.out.println:

(A.length) should print 2.
(A[0].length) should print 3.
(A[0][0].length) should print 4.

Ex. 2.15
multiplier=numRows/numTargetRows=600/(600*0.2)=5

| i | j | originalRow=i*multiplier | originalCol=j*multplier |
|---|---|---|---|
| 0 | | | |
| | 0 | 0 | 0 |
| | 1 | 0 | 5 |
| | 2 | 0 | 10 |
| | 3 | 0 | 15 |
| 1 | | | |
| | 0 | 5 | 0 |
| | 1 | 5 | 5 |
| | 2 | 5 | 10 |
| | 3 | 5 | 15 |
| 2 | | | |
| | 0 | 10 | 0 |
| | 1 | 10 | 5 |
| | 2 | 10 | 10 |
| | 3 | 10 | 15 |
| 3 | | | |
| | 0 | 15 | 0 |
| | 1 | 15 | 5 |
| | 2 | 15 | 10 |
| | 3 | 15 | 15 |

The program prints two images. The original is crisper because it is based on 5-times as many pixels as the shrunk image. Obviously, the transition from pixel to pixel is 5 times smoother in the original.

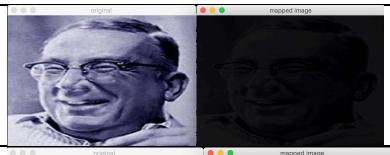
With System.out.println (shrunkPixels[i][j]), the program prints the addresses of the rows multiplied by columns of the 3D array shrunkPixels. In turn these addresses should point to the values in the third dimension of this array, the values specifying the color scheme of each of the pixels created by this array.

---

Ex. 2.19
Albert Einstein Memorial outside of the National Academy of Sciences Building in Washington, D.C.

Ex. 2.21

I can’t recognize the person in the picture.

| | |
|---|---|
| a=0.1, b=0.0 |  |
| a=1.5, b=0.0 |  |
| a=3.0, b=0.0 |  |
| | |
| a=1.0, b=25.0 |  |
| a=1.0, b=70.0 |  |
| a=1.0, b=100.0 |  |
| | |
| a=-1.0, b=255 |  |

---

Ex. 2.23

BEFORE: A=[1, 2] BEFORE: B=[3, 4]
BEFORE: X=[1, 2] BEFORE: Y=[3, 4]
AFTER: X=[3, 4] AFTER: Y=[1, 2]
AFTER: A=[1, 2] AFTER: B=[3, 4]

A and B are passed into X and Y. The swapping of X for Y inside swap() does not affect A and B. The swapping statements exchange the addresses/pointers labeled X and Y. It is not obvious, however, whether A and B are passed to X and Y by value or by pointer.

---

Ex. 2.24

BEFORE: A=[1, 2] BEFORE: B=[3, 4]
BEFORE: X=[1, 2] BEFORE: Y=[3, 4]
AFTER: X=[3, 4] AFTER: Y=[1, 2]
AFTER: A=[3, 4] AFTER: B=[1, 2]

The swapping of values in X and Y affects the values of A and B. Therefore, it is obvious that A and B are passed to X and Y by pointer, not by value. In other words after the passing, A and X point to the same values (or the same ONE array). Changing values through one of these two labels, A or X, will become obvious if we access the array through the other label. The formula here is ONE ARRAY + TWO LABELS/POINTERS.