IntGrid2D.java analysis — space complexity. Paxton 11/29/17 honor code upheld

* "get..." methods do not affect memory
* setpoint(); does not ~~add new~~ require new memory space to be allotted.

*** therefore the space complexity only depends on the IntGrid2D constructor, which contains ~~five constants~~ variables (uLX, uLY, lRX, lRY, and fill) assignments and the double loop that instantiates ~~the~~/fills the 2D array. The variable assignments always require constant time, so can be discounted.

```
for (int x = 0; x < i; x++) {
    for (int y = 0; y < j; y++) {
        grid[x][y] = fill;
    }
}
```

(x & y are counters, i & j are the dimensions of the 2D array)

* the inner loop executes j times, and the outer loop executes i times. & i and j are ~~given by the~~ determined by the inputs to the constructor, so they are the independent variable in the space complexity equation.

* the call "grid[x][y] = fill;" ~~takes constant time, so the time for the doo~~ assigns a value to one spot in memory, so the space function $S(i) = i \times j$

if $i \cong j$ then $S(i) \cong i^2$

if $i >> j$ then $S(i) \cong i$

so $S(i)$ is in $O(n^2)$ and $\Omega(i)$.

polynomial linear