# CS 1: Intro to CS

Nested Lists/Arrays, Intro to MP8
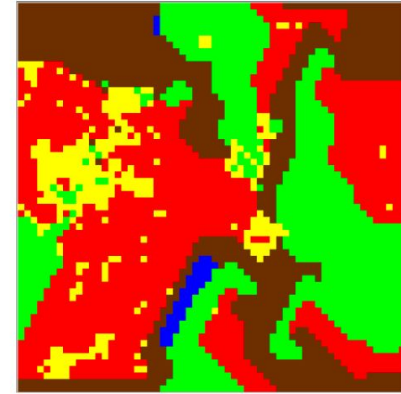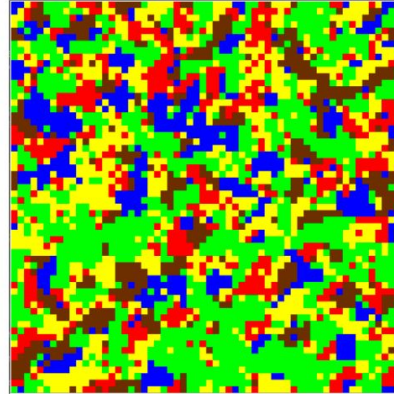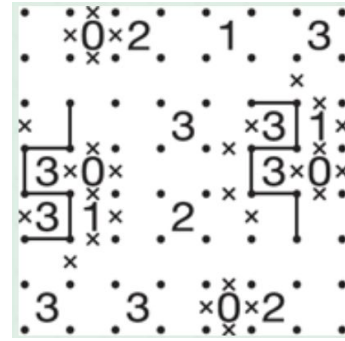
# Agenda

Introduce MP8

Random in Java

Arrays (for MP8)
- Accessing, loops, 2D arrays
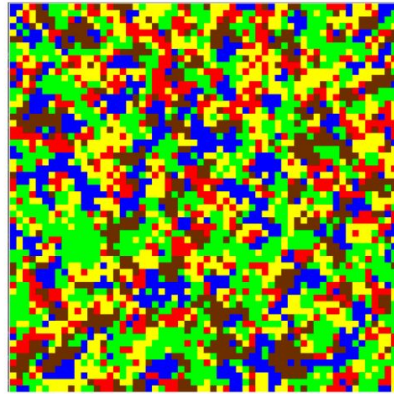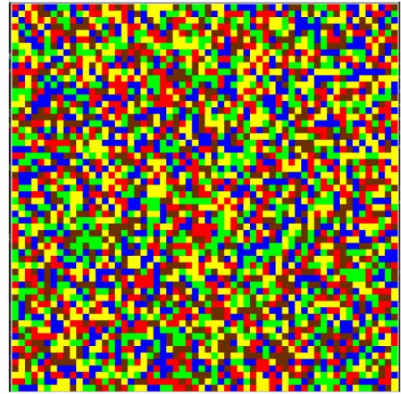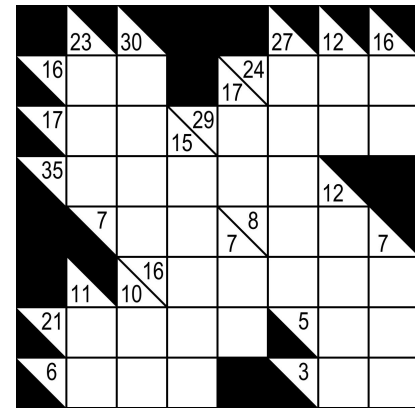
Please fill out this [Week 10 "Exploration Lecture" Interest Form](#) for upcoming lectures!

# MP8: PokePixel Simulator (Demo)

MP 8 (the final assignment before the exam) is out

On Friday, we'll wrap up and introduce the Final Exam overview/format

A few options for the rest of Friday's lecture (answer on Exit Ticket Q3!):
- Dedicated to working on small parts/strategies for MP 8 with students
- General QA for any review/extra material to explore
- Other things you'd like to review before next week?

# Review: Random in Java

To work with random numbers in Java, we use the **Random** object (requiring import **java.util.Random** at the top)

The two methods that are most commonly used are **r.nextInt(start, stop)** and **r.nextDouble()** (returns a random **double** between **0.0** and **1.0**) (**RandomDemo.java**)

```java
public static void randomDemo() {
    // in B.6., Random mutator = new Random();
    Random r = new Random();
    // Set a rate of success for a random coin flip
    double HEADS_RATE = 0.5;
    int heads = 0;
    // random digit between 0 and 9
    int randDigit = r.nextInt(10);
    int flips = randDigit;
    System.out.println("Flipping " + flips + " coins...");
    for (int i = 0; i < flips; i++) {
        // r.nextDouble() returns a random double between 0.0 and 1.0
        double coinFlip = r.nextDouble();
        // 50% chance of heads
        if (coinFlip < HEADS_RATE) {
            heads++;
        }
    }
    System.out.println(heads + " flips were heads!");

    // Some other examples
    String s = "hello world!";
    // random char index between 0 and length of string
    int randomIndex = r.nextInt(s.length());
    // access the random character by index
    char randomChar = s.charAt(randomIndex);
    System.out.println("Randomly selected " + randomChar +
                " at index " + randomIndex + " of " + s);
}
```

```
mehovik@Els-MacBook-Pro:~/eipsum.github.io/cs1/lectures/lec25$ javac Lec25.java
mehovik@Els-MacBook-Pro:~/eipsum.github.io/cs1/lectures/lec25$ java Lec25
Flipping 6 coins...
2 flips were heads!
Randomly selected   at index 5 of hello world!
mehovik@Els-MacBook-Pro:~/eipsum.github.io/cs1/lectures/lec25$ javac Lec25.java
mehovik@Els-MacBook-Pro:~/eipsum.github.io/cs1/lectures/lec25$ java Lec25
Flipping 7 coins...
4 flips were heads!
Randomly selected h at index 0 of hello world!
mehovik@Els-MacBook-Pro:~/eipsum.github.io/cs1/lectures/lec25$ []
```

# File IO in Java (From MP7)

In Java, we use the File and Scanner objects to process files (these do not need to be imported):

**File Processing in Python:**

```python
f = open('some_file.txt')
line_count = 0
char_count = 0
for line in f:
    line_count += 1
    char_count += len(line)
f.close()
print(f'Lines: {line_count}, Chars: {char_count}')
```

**File Processing in Java:**

```java
File f = new File("some_file.txt");
Scanner reader = new Scanner(f);
int lineCount = 0;
int charCount = 0;
while (reader.hasNextLine()) {
    String line = reader.nextLine();
    lineCount++;
    charCount += line.length();
}
reader.close();
System.out.println("Lines: " + lineCount +
                   ", Chars: " + charCount);
```

Note: If you are processing lines with numbers, you can use `Integer.parseInt(line)` or `Double.parseDouble(line)` in Java (where `line` is a String)

Practice: averageValueInFile

# 2D Lists and Java Arrays

*VSCode demo*

PythonTutor visualizer for reviewing nested lists and aliasing (see also Reading 8)

2dlists.py

ArrayDemo.java (practice from lecture, as well as additional exercises to try in either Python or Java)

CodeStepByStep Practice:
- list_tracing_2d (Python)
- list_mystery_2d (Python)
- arrayMystery2d (Java)