

## 220: Discussion 1

Though a lot of 220 is teaching you key principles for writing code, *reading* code is just as important! In these problems you will have to consider some pretty complicated pieces of code - so the key is to reason about them systematically, keeping track of variable and argument values, as well as which functions are calling who.

### Problem 1

```
function foo(n) {
  console.log(n);
  if (n % 7 !== 0) {
    foo(n + 5);
  }
}
foo(6);
```

What does this code output?

### Problem 2

```
// Recall the map function from lecture:
// map<S,T>(f: (x : S) => T, a: S[]): T[]
function map(f, a) {
  let result = [];
  for (let i = 0; i < a.length; ++i) {
    result.push(f(a[i]));
  }
  return result;
}

function bar(str) {
  const q = [1, 4, 2, 6, 5, 9, 7, 4];
  const r = str.length;
  return q[r];
}
let x = map(bar, ['stack', 'random', 'tree', 'reverse']);
console.log(x);
```

What does this code output?

### Problem 3

```
function baz(a, b) {
  function qux() {
    return a + b;
  }
  if (a % 3 === 0) {
    return qux;
  } else {
    return baz(b + 1, a);
  }
}
let z = baz(2, 10);
let q = 10 + z();
console.log(q);
```

What does this code output?