1) (5 pts) DSN (Recursive Coding)

Mathematically, given a function f, we recursively define $f^k(n)$ as follows: if k = 1, $f^1(n) = f(n)$. Otherwise, for k > 1, $f^k(n) = f(f^{k-1}(n))$. Assume that a function, f, which takes in a single integer and returns an integer already exists. Write a ***recursive*** function fcomp, which takes in both n and k (k > 0), and returns $f^k(n)$.

```
int f(int n);
```

Solution #1

```
int fcomp(int n, int k) {
    if (k == 1) return f(n);
    return f(fcomp(n, k-1));
}
```

Solution #2

```
int fcomp(int n, int k) {
    if (k == 1) return f(n);
    return fcomp(f(n), k-1);
}
```

Grading: 2 pts for the base case.
3 pts for the recursive case.