More recursive functions...

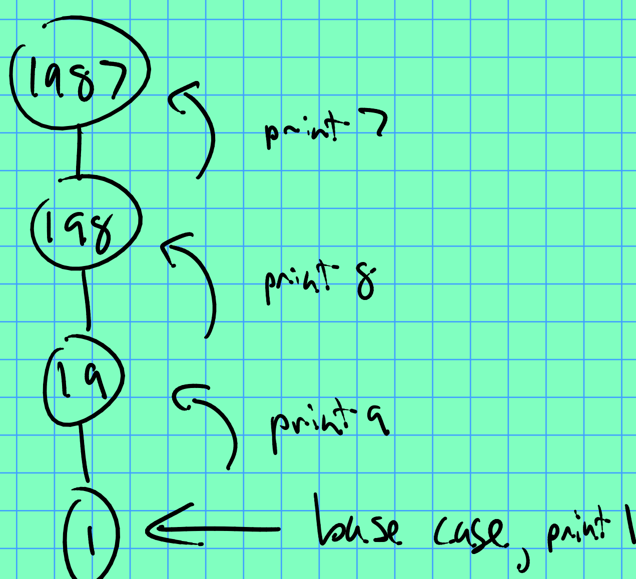
void PrintVertical (int n) :

PrintVertical (1987) prints
1
9
8
7

Recursive breakdown :

$\left.\begin{array}{l}\text{print first k-1 digits } (n/10) \\ \text{print last digit } (n \% 10)\end{array}\right]$

```
void printVertical(unsigned int n) {
    if(n < 10) { // one digit.
        cout << n << endl;
        return;
    }
    printVertical(n/10);
    cout << n%10 << endl;
}
```



Example: greatest common divisor (gcd)

Where is the recursive structure??

How to solve gcd(a, b) in terms of gcd(a', b') for smaller inputs?

<u>Claim:</u> common divisors of (a, b) are the same as the common divisors of (b, r) where $r = a \% b$.

Recall "division algorithm":

$$a = q \cdot b + r \quad \text{where } 0 \le r \lneq b$$

Suppose $d \mid a$ & $d \mid b$. Then $a = md$, $b = m'd$.

Then $d \mid r$: $r = a - q \cdot b$
$$= md - qm'd = (m - qm')d \quad \checkmark$$

other direction: $d \mid b$ & $d \mid r \implies d \mid a$.
just factor out the d. ✓

Two things to notice:

① $\gcd(a,b) = \gcd(b,r)$

② $r \lneq b$.

In C++:

```
int gcd(int a, int b) {
    if(a%b == 0) return b;
    return gcd(b, a%b);
}
```