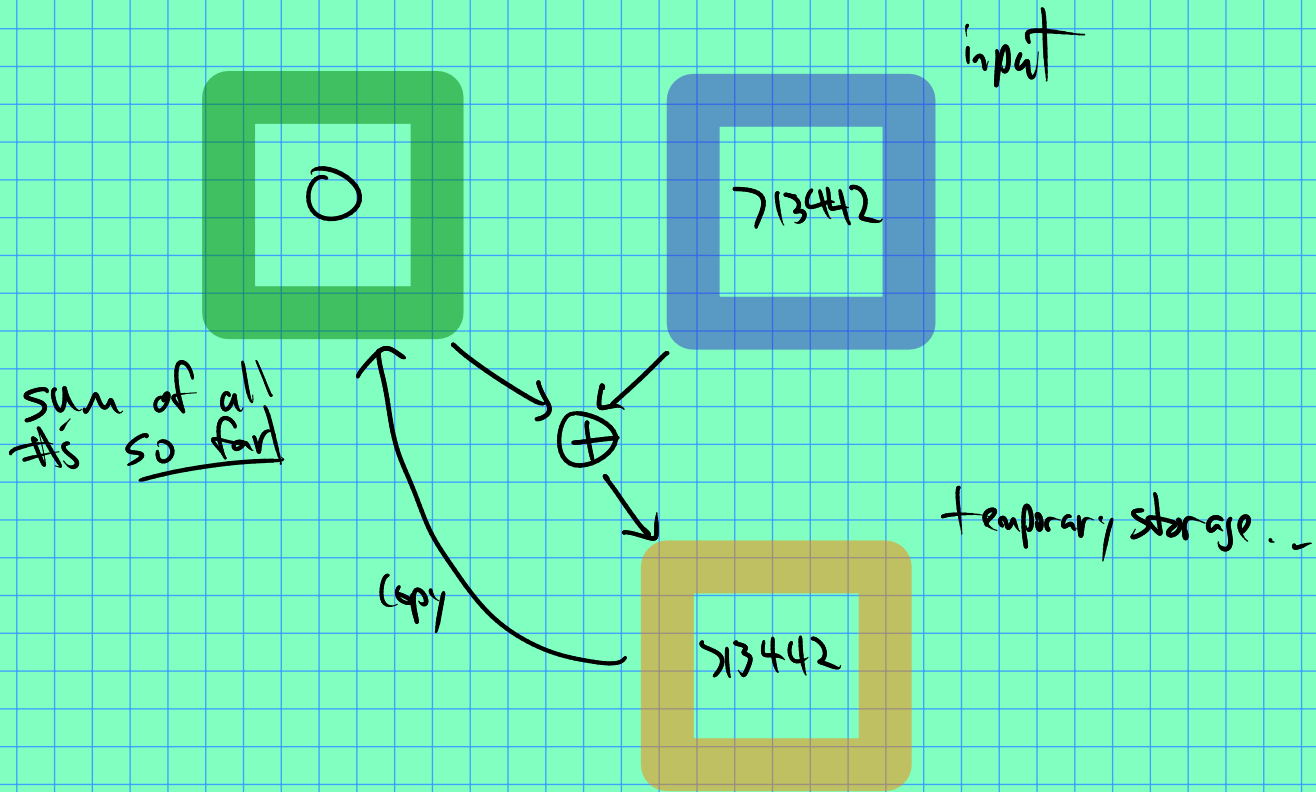


Problem: compute the sum of a list (arbitrary length!) of integers.



- ① Get 3 pieces of paper (blue, green, orange)
- ② Write 0 on the green
- ③ listen for #, write on blue paper
- ④ Add blue + green, write result in orange.
- ⑤ Copy # on orange to green.
- ⑥ repeat from ③ until out of #s.
- ⑦ Answer is on green note.

Note: the above looks a lot like our max program from the first lecture. General pattern:
Want to compute

$$x_0 \star x_1 \star x_2 \star \dots \star x_n.$$

If \star has a "neutral" element e , such that $x \star e = x$, then the following will always work:

```
a = e;  
while (cin >> x) {  
    a = a  $\star$  x;  
}  
// answer is in a.
```

New problem: compute the gcd (greatest common divisor) of 2 given (positive) integers.

E.g., if $a=12$, $b=18$, then
 $\text{gcd}(a,b) = 6$.

Idea: we can "brute force" the answer.

I.e., we can try all possibilities and check.

$$\text{gcd}(9, 15) = 3$$

How to check if some value d is a common divisor of a & b ?

$$\underline{a \% d == 0 \quad \text{and} \quad b \% d == 0.}$$

Upper bound for $\gcd(a, b)$? $\min\{a, b\}$.

List of candidates:

$[1, 2, \dots, \min\{a, b\}]$

d

$x \neq$
 $y \dots$

$\text{int } n = \min(a, b);$

$\text{int } d = 1;$

$\text{while}(d \leq n) \{$

$d = d + 1;$

$\}$