"Hello Everyone!"

Problem Solving

1. Factorial →
2. Nested loops → Concept
3. Count factors →
4. Pac - man
   → for-loop (Q1)

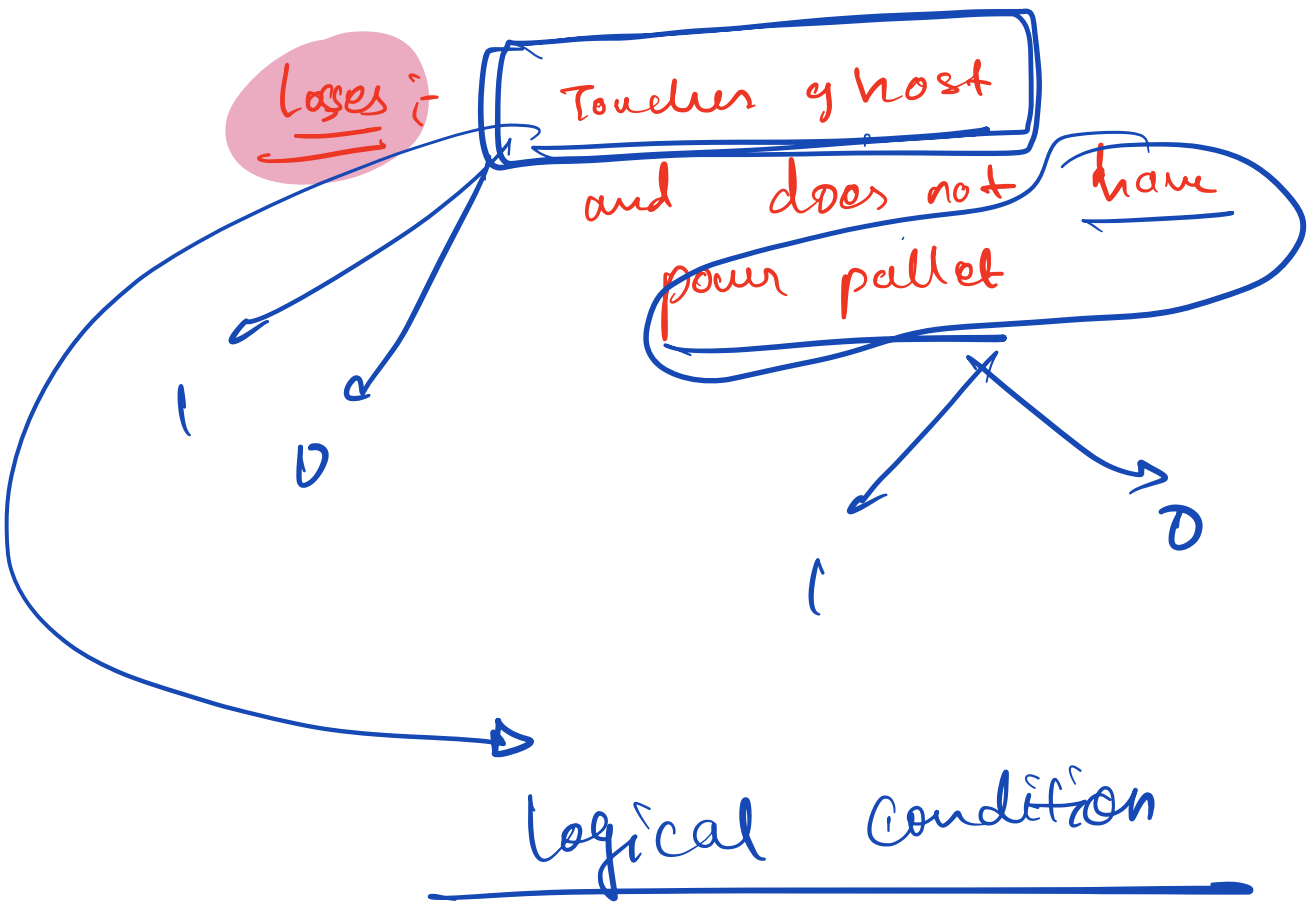
① Pac - man

① Power pallet ?

② Touching a ghost ?

1 → Yes

0 → No



Logical condition

(TG) and (not PP)

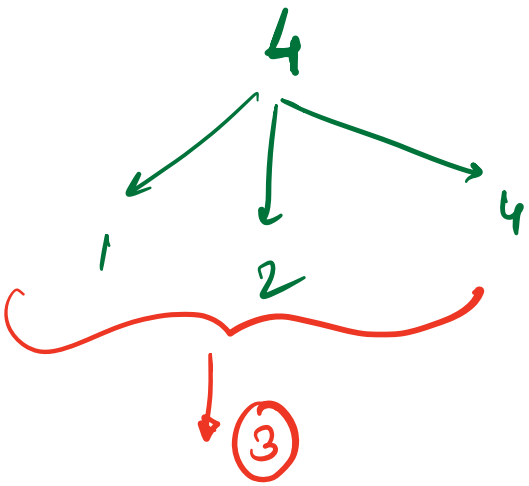
↳ True

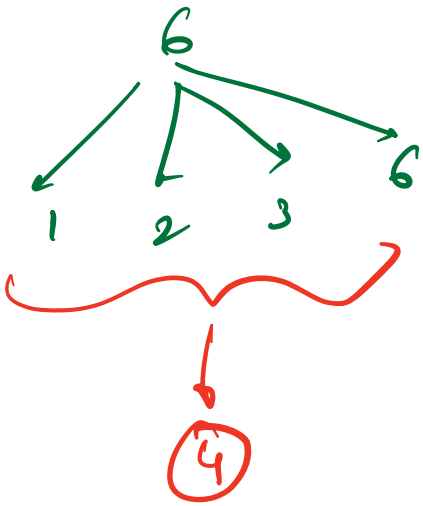
↳ '1'

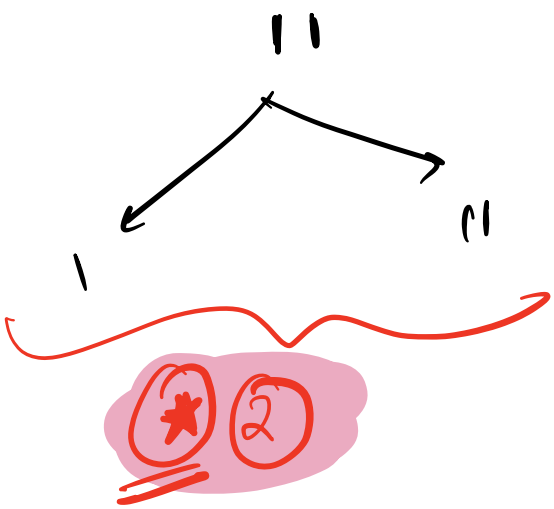
else

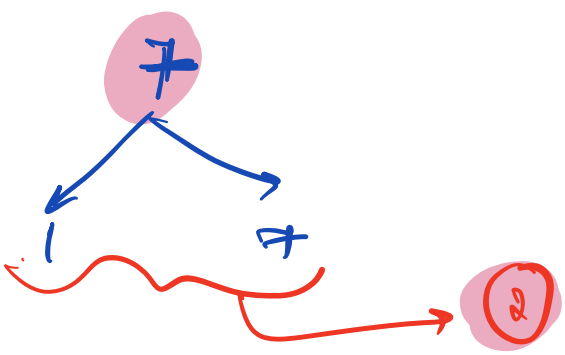
↳ '0'

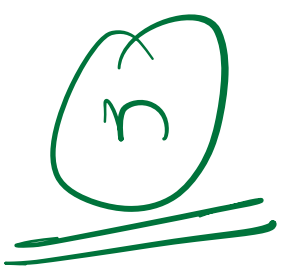
* factors





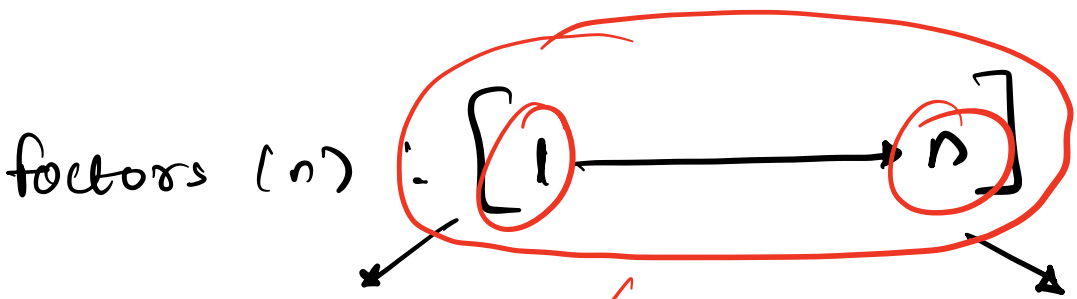






1 ⟶ always

n ⟶ always



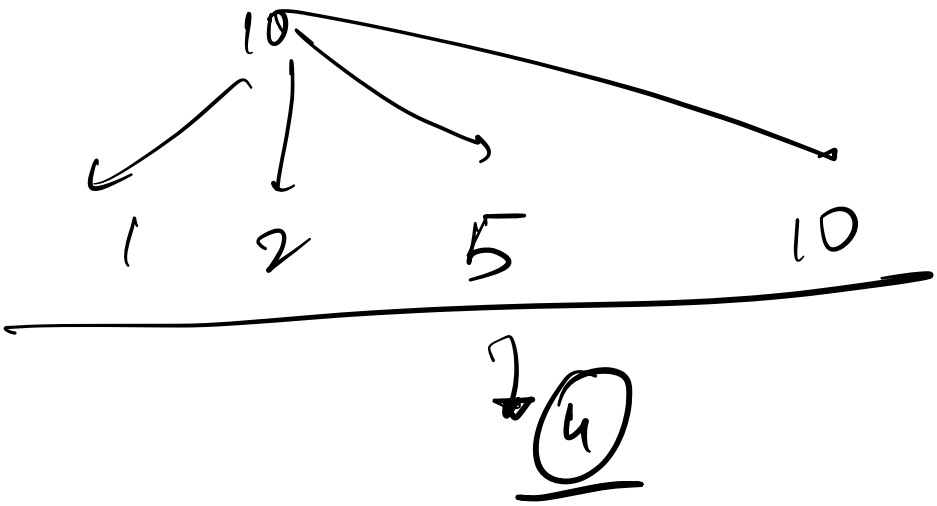
count = 0

```
for i in range(1, n+1)
    if n % i == 0
        count = count+1
    count = 1
```

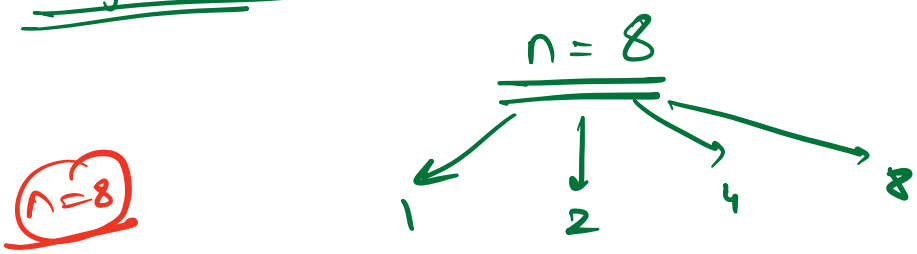
i divides n completely

i is a factor of n

i is "divisor"



Dry Run :



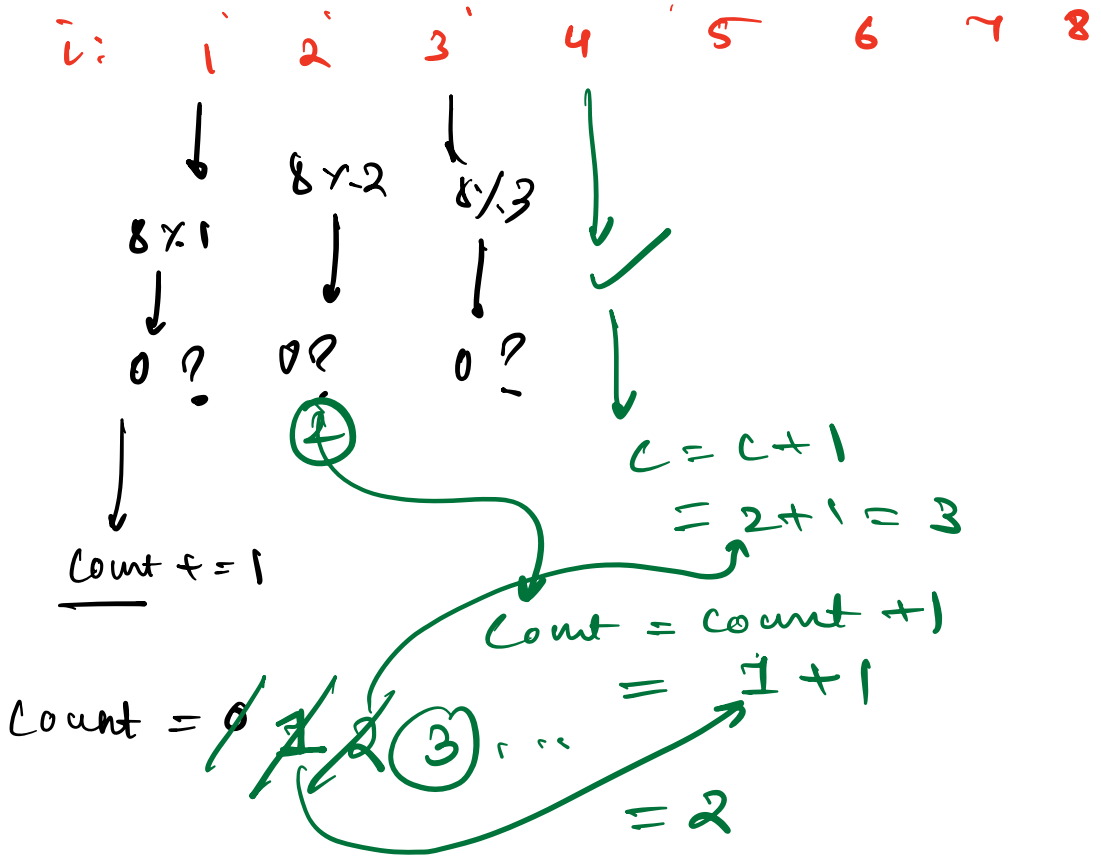
n=8

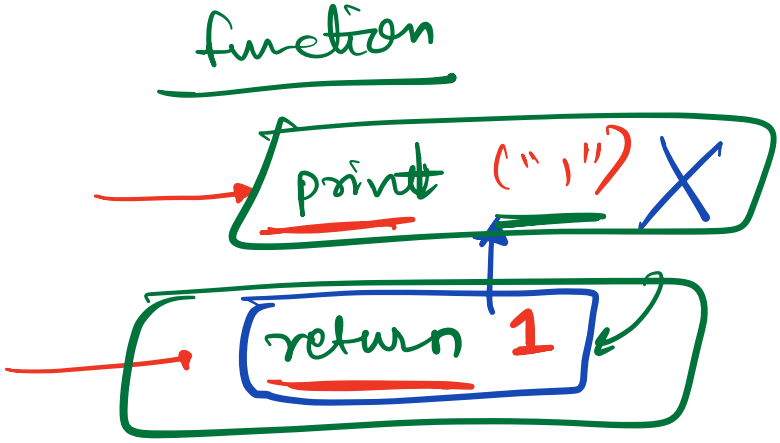
count = 0

for i in range (1, n+1) :

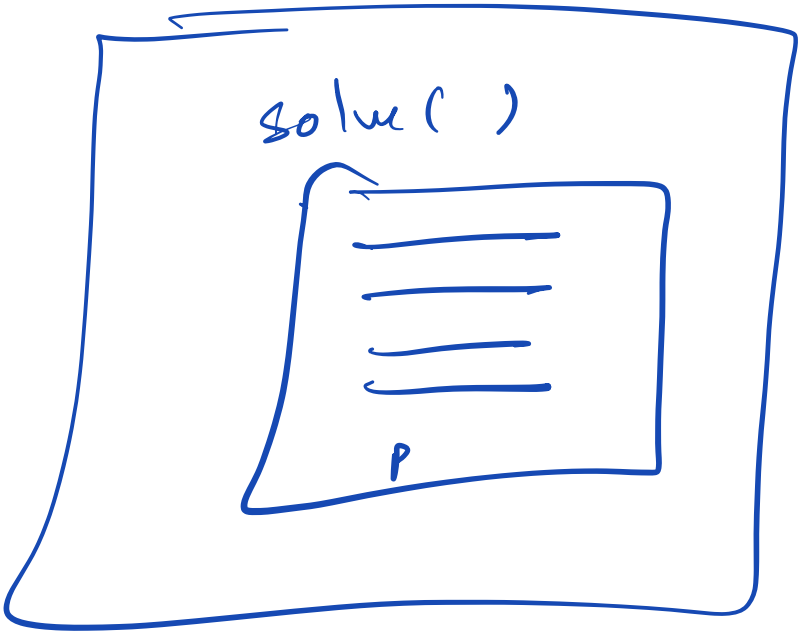
```
if n % i == 0 :
    count += 1
```

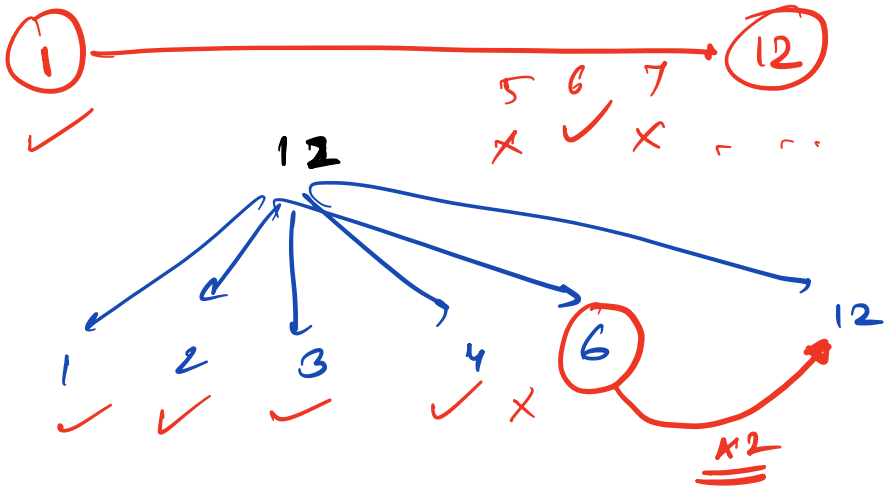
1 ⟶ 8

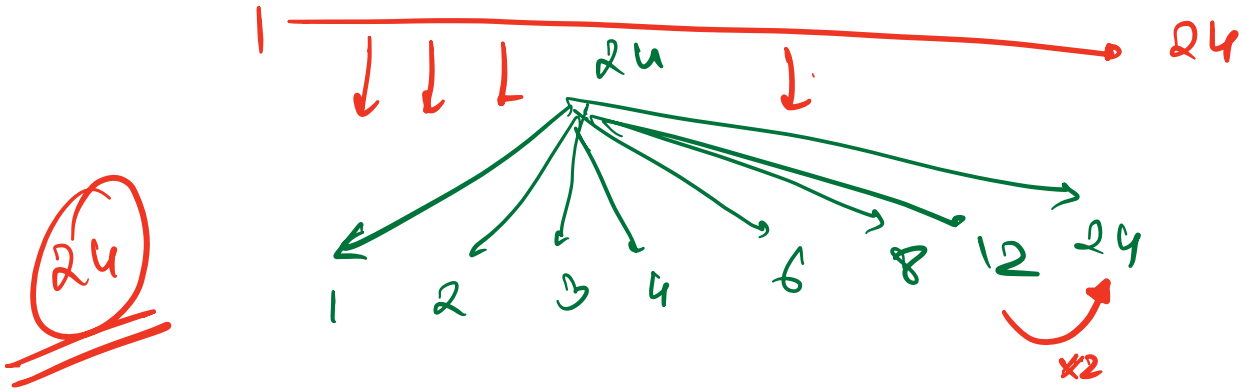

i: 1 2 3 4 5 6 7 8
8%1
8%2
8%3
0 ?
0 ?
0 ?
count += 1
c = c + 1
= 2 + 1 = 3
count = count + 1
= 1 + 1
= 2
Count = 0 1 2 3 ...



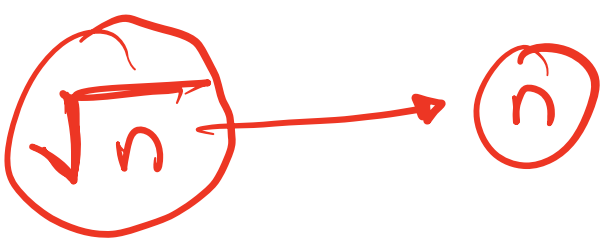
return ans





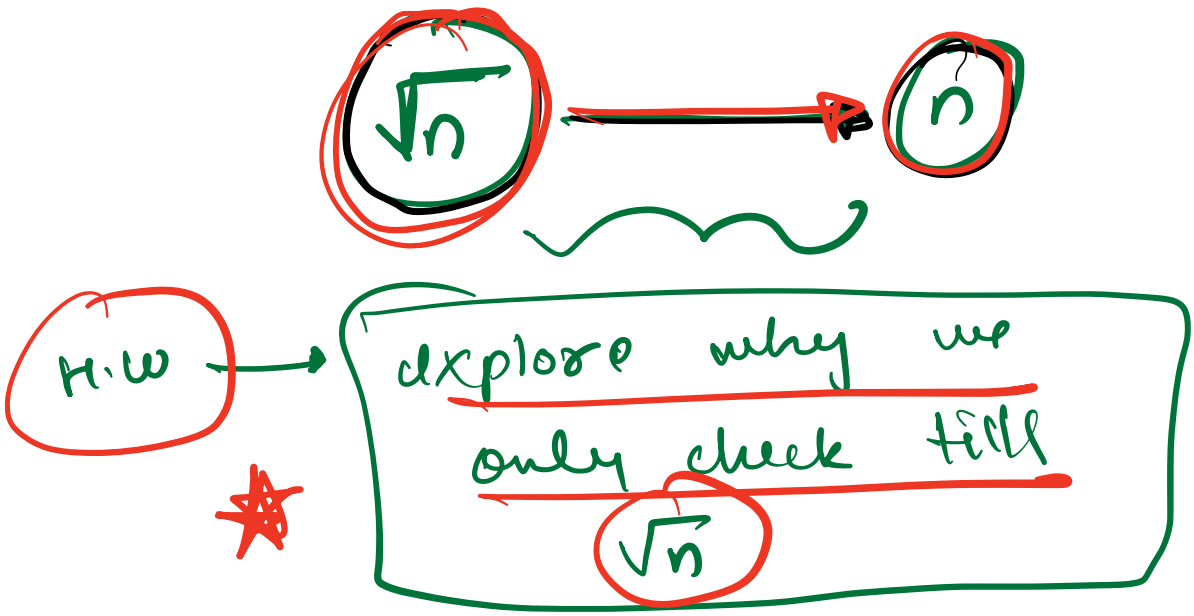


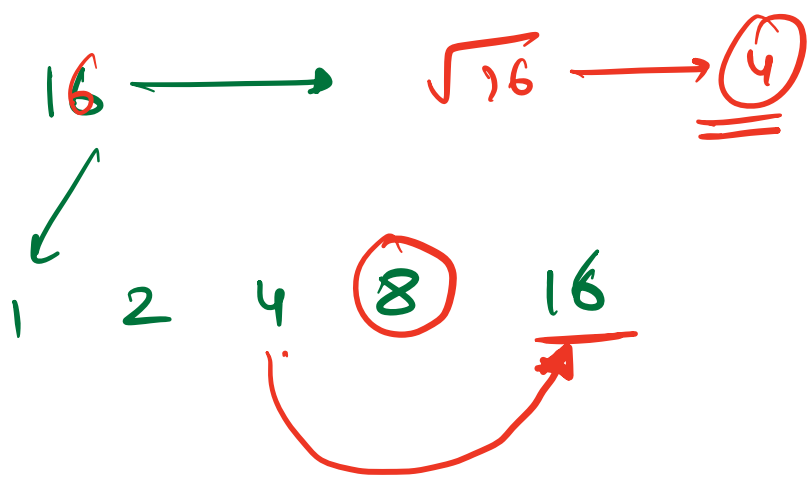
★ 100,00000



Imp

factors





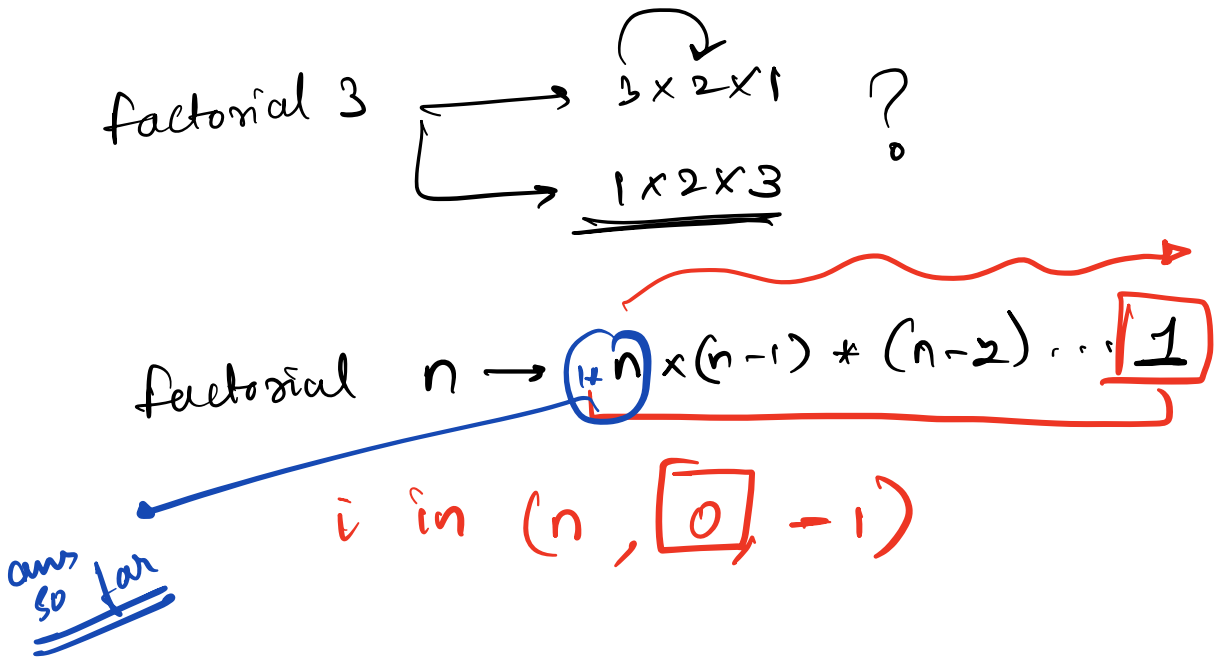
→ 84: Operators

↳ String Concatenation

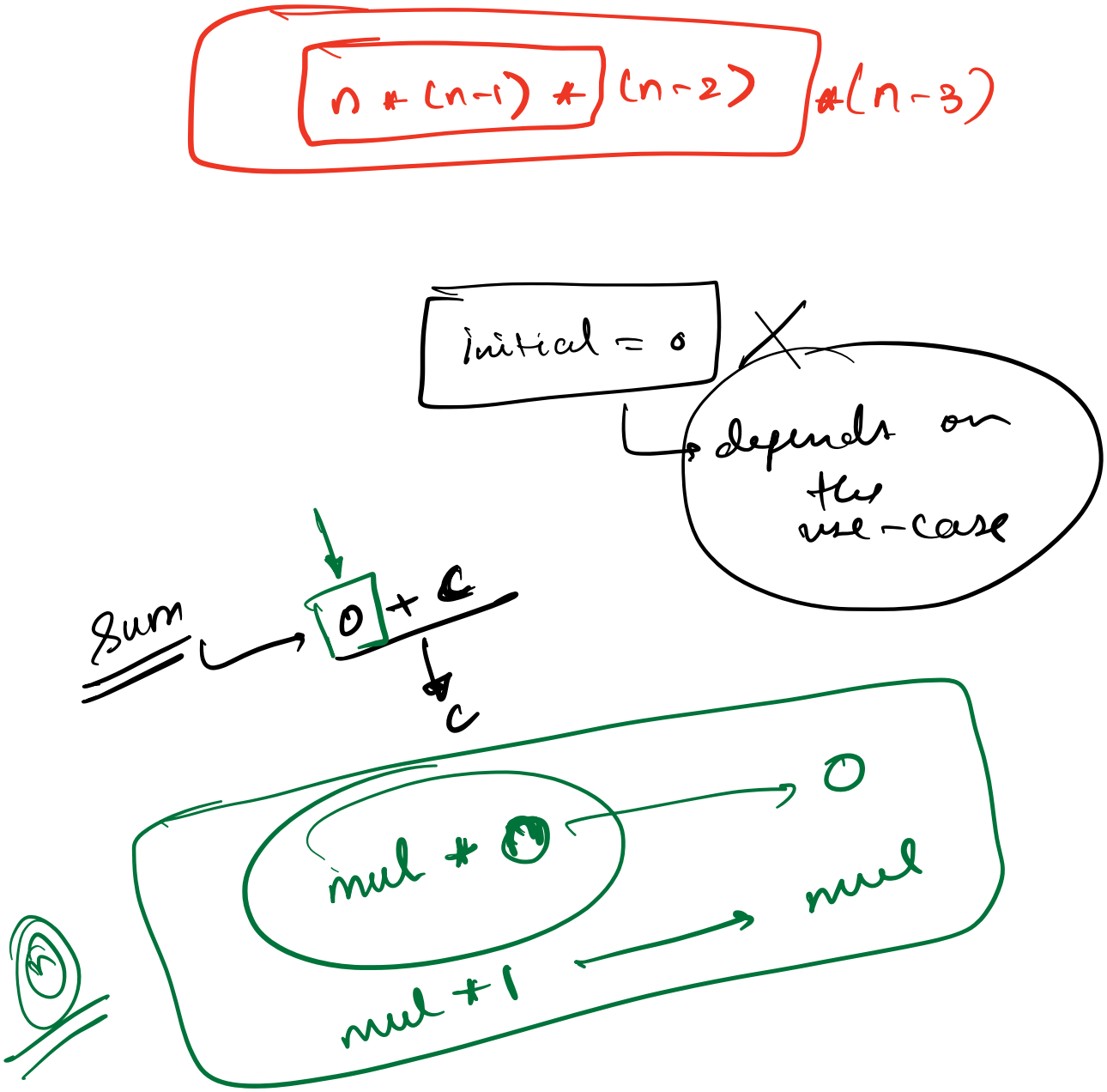
↓

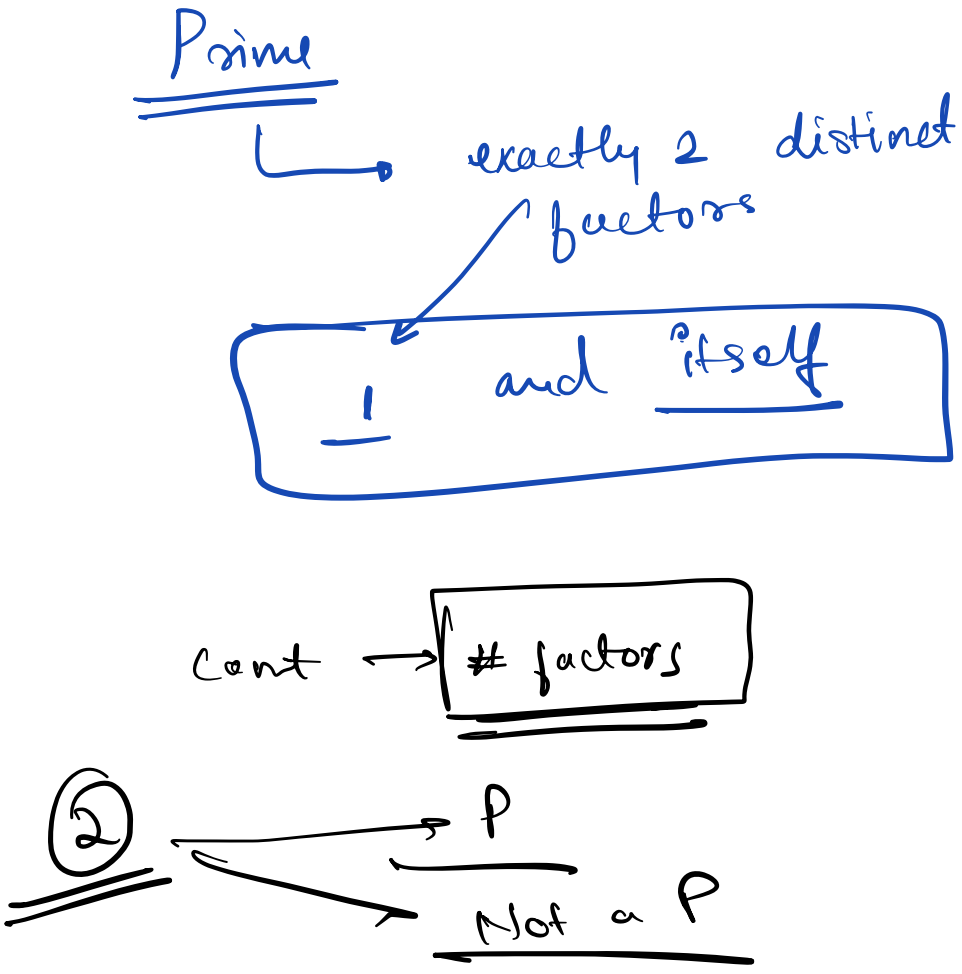
After Lict

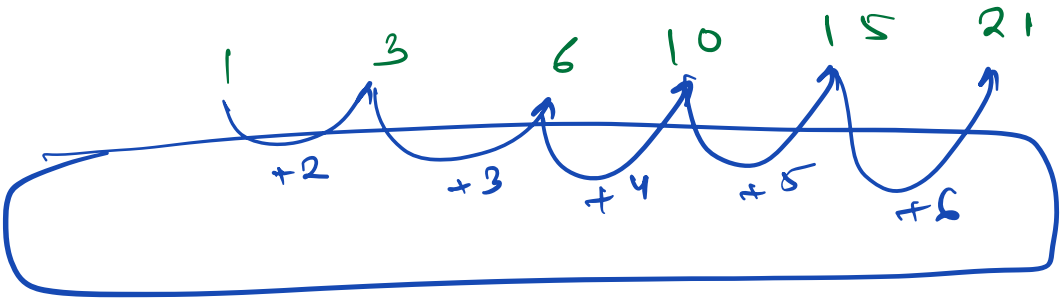
factorial


factorial 3
3 x 2 x 1
1 x 2 x 3
?
factorial n → n x (n-1) * (n-2) ... 1
i in (n, 0, -1)
ans so far


count += 1
+1
+1
+1


n * (n-1) * (n-2) * (n-3)
initial = 0
depends on the use-case
sum
0 + c
c
mul * 0
0
mul
mul + 1


Prime
exactly 2 distinct factors
1 and itself
Cont
factors
2
P
Not a P

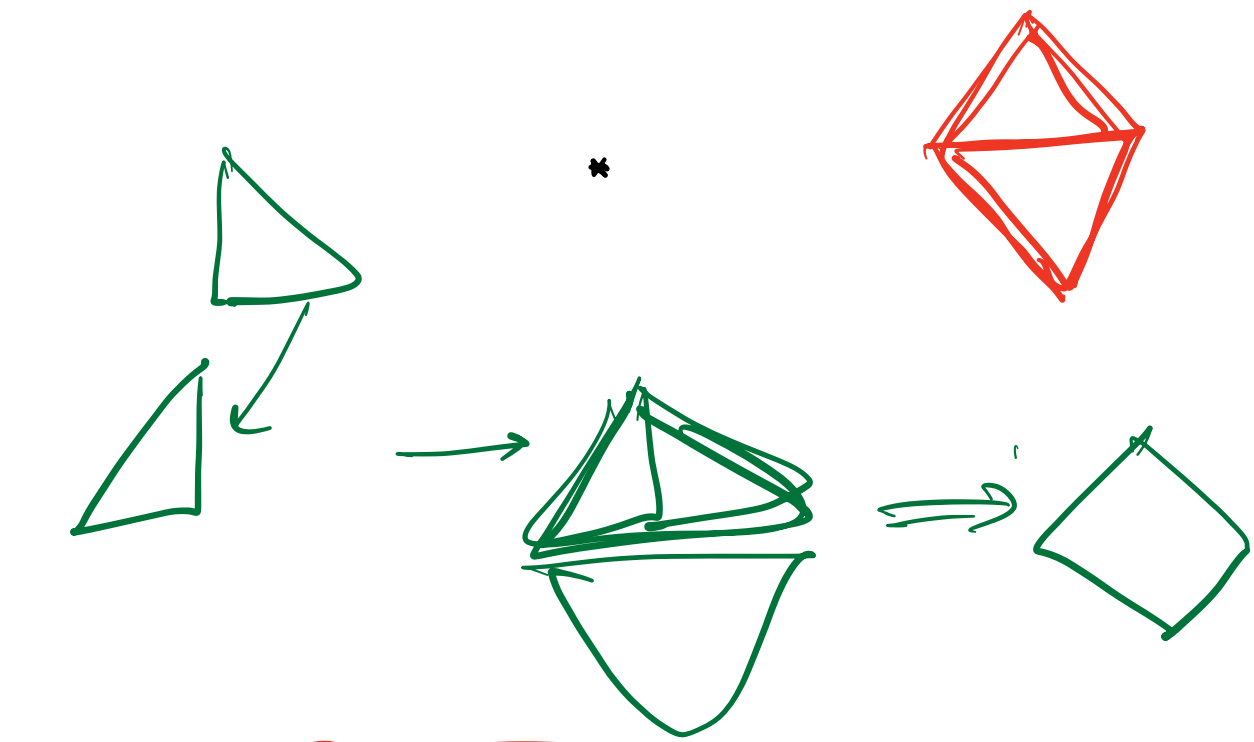


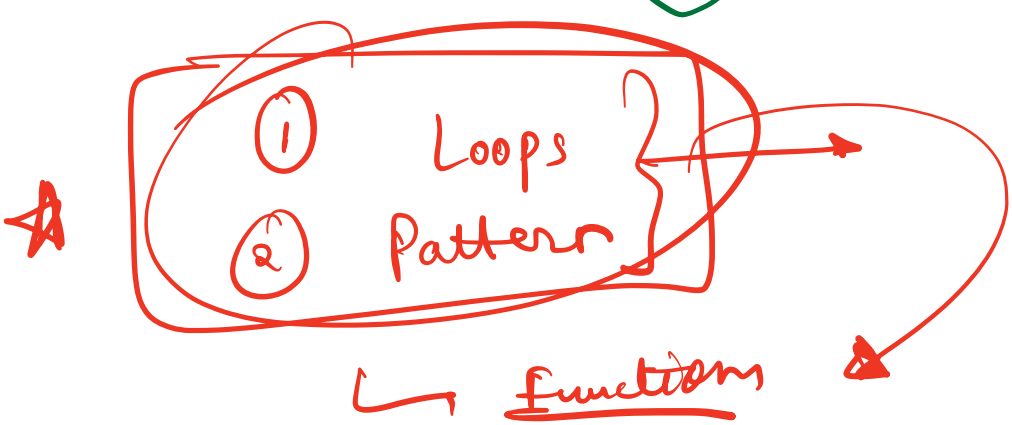
h.w → P5-2 → Q1

Pattern Printing

↳ Q4 → ★

n = 5





↳ Functions

Sunday EOD