$f_n = 0$

$s_n = 1$

nsum($f_n + s_n$ = newnumb

$f_s : s_n$

$s_n : n_s$

for($i = 0$; $i <= n$; $i++$) { (n = 10)

sout(f_n);

$nn = f_n + s_n$; → $f_n = s_n$; → $s_n = nn$

}

1 1 2 3 5 ——

0 1 1 2 3 5

$i = 0$ / $i = 1$ | $i = 2$ | $i = 3$

$nn = f_n + s_n$ $i = 4$

$nn = 2 + 3$

$f_n = s_n$

$s_n = nn$

```
int fn=0, sn=1, next=0
for(i=0; i<=n; i++){
    Sout(fn);
    next= fn+sn;
    fn=sn;
    sn = next;
}
```

n= 321 4256

Reverse,

Sum of digit

find largest digit = ?

find Smallest digit = ?

n Sq = n × n →

n² = Abcdeg

deg

eg

Q. digit count

621

2

3

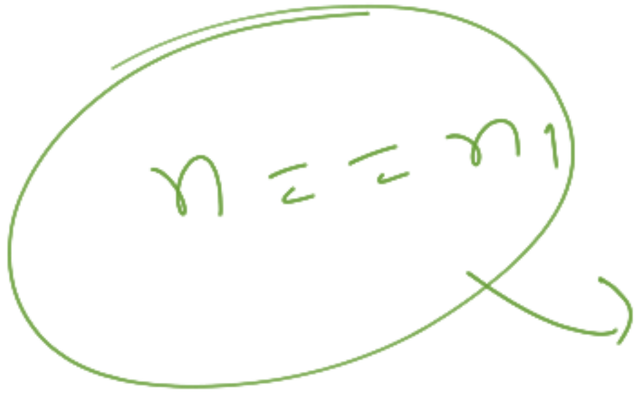
2

21.

3 & 69

n1 = 369 ←

n = last 3 digit = 369



True — Auto ✓

for not ∩

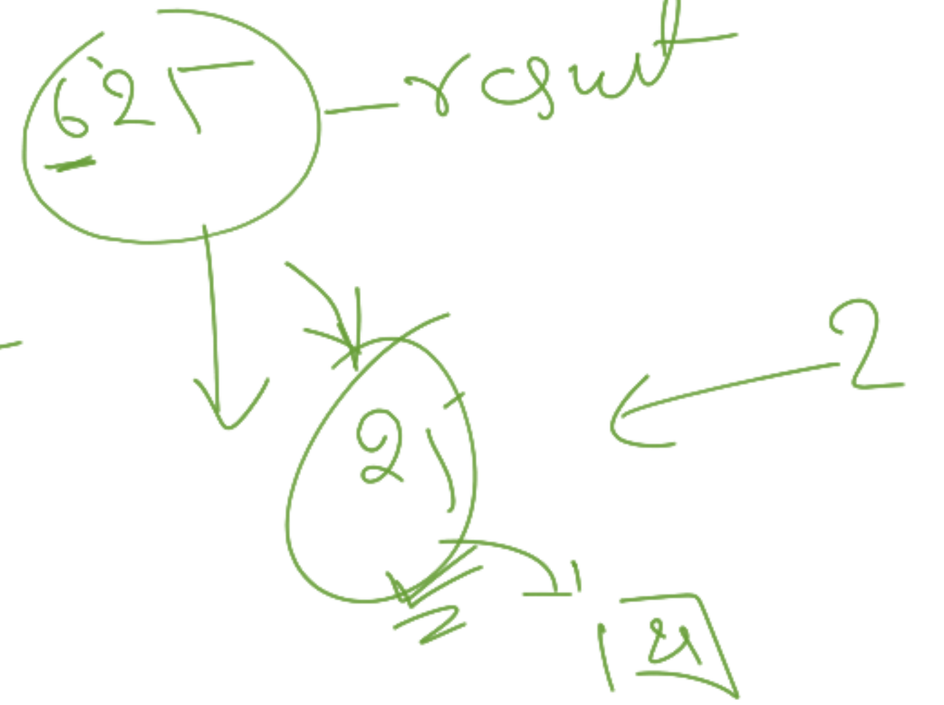
n = 25 Autor

25^2 = 625 — result

n: A

? no of digit

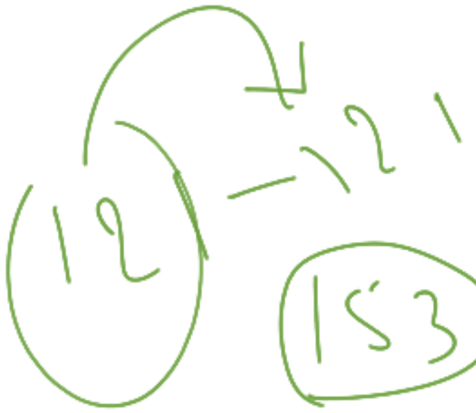
2



21 = 21' ✓

21 = 25' ✗

Armstrong



no of digit = ? = 3

digit sum = ?

$1^3 + 5^3 + 3^3 =$?

$1 + 125 + 27$

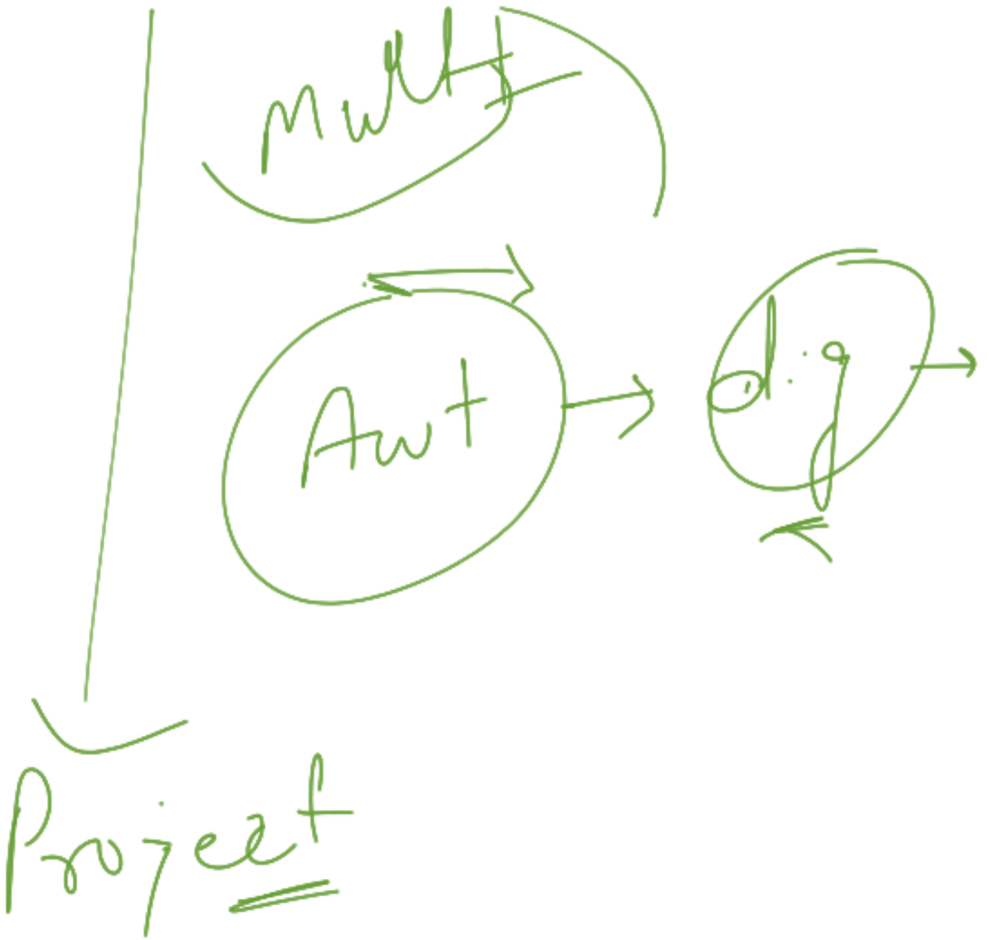
Exception hand

try

catch

multiple cat

final



Armstrong

Palindrom

Automorphic

fibonacci

find largest digit / smallest digit in a number

eg → 2416 → largest — 6

smallest → 1