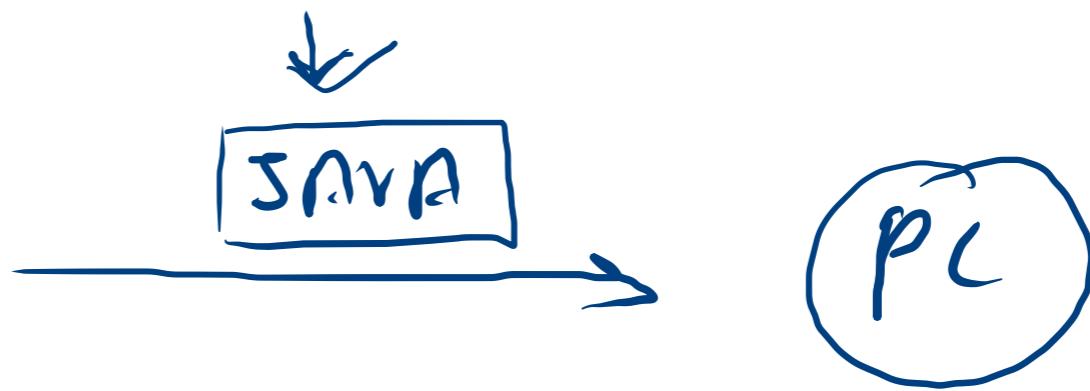




D.S. ALGO





Input

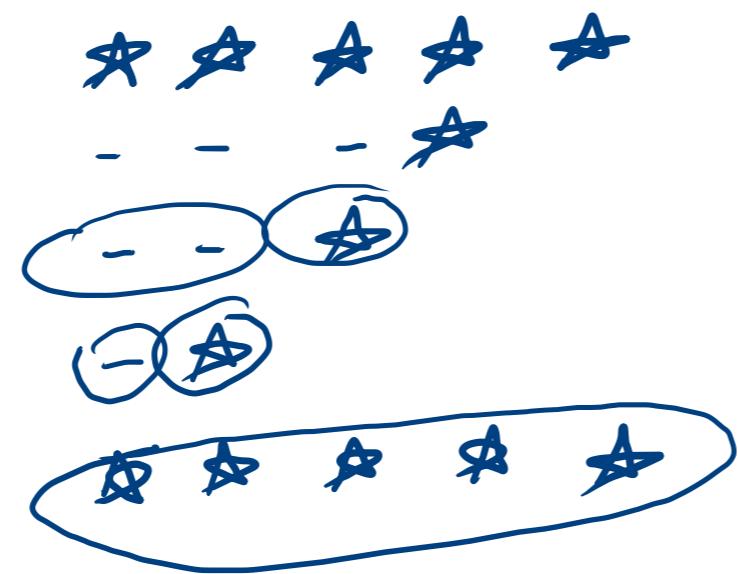


process



output

Z





`n`
10

JAVA

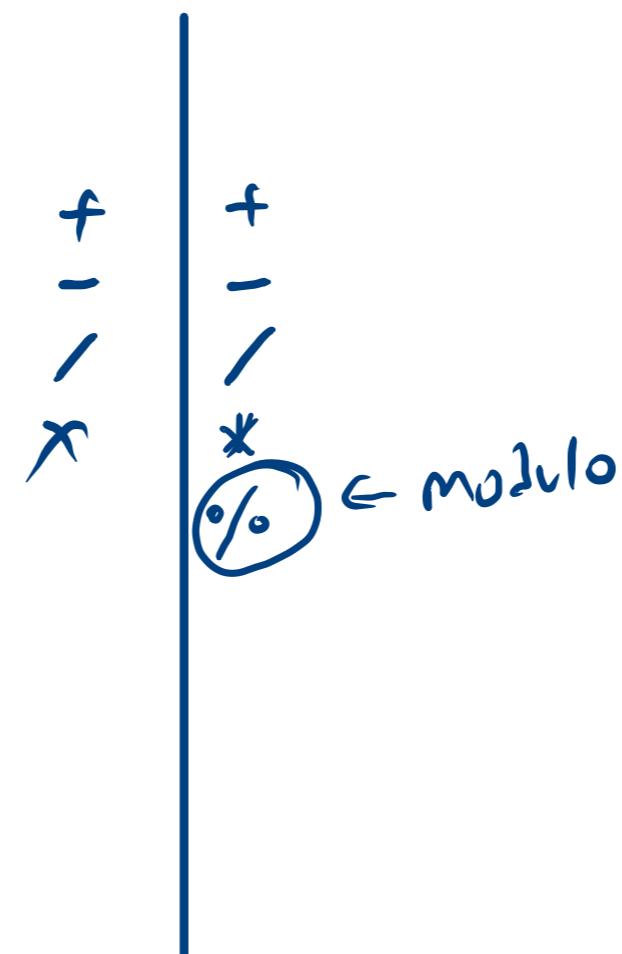
MUTATION

Let

`n = 10`

`10. 10`

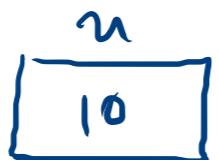
`n + 15`



`int`

`n = 10;`

`n + 15;`
25



```

public static void main(String[] args) {
    int x = 10;
    System.out.println(x);
    System.out.println(x+15);
    System.out.println(x-15);
    System.out.println(15/x); ←
    System.out.println(x*15);
    System.out.println(15%x);
    // System.out.println("hello");
}

```

- 10
- 25
- -5
- 1
- 150
- 5

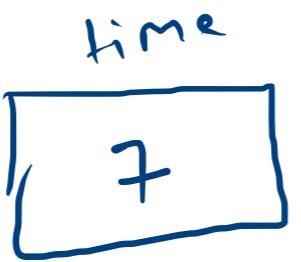
$$\begin{array}{r}
 & 1 \\
 & \swarrow \quad \searrow \\
 10 & 15 \\
 & \overline{10} \\
 & \underline{\quad \quad \quad} \\
 & 5
 \end{array}
 \text{ rema}$$

$$25 = \frac{n+15}{10+15}$$

$$\begin{array}{l}
 n-15 \\
 10-15 \\
 -5
 \end{array}$$

$$\begin{array}{ll}
 15/n & n \neq 15 \\
 15/10 & 10 * 15 \\
 1 & 15^0
 \end{array}$$

$$\begin{array}{l}
 15 \neq 10/10 \\
 5
 \end{array}$$



- int time = 7;
if(time <= 12){
 - System.out.println("Good Morning");
- }else{
 - System.out.println("Good after noon");
- }

time 18

18

```
• int time = 18;
  18 <= 12 ✓ false
• if( time <= 12){
    System.out.println("Good Morning");
}else if(Time < 19){ 18 < 19 ✓ true
    System.out.println("Good after noon");
}else{
    System.out.println("Good Evening");
}
```

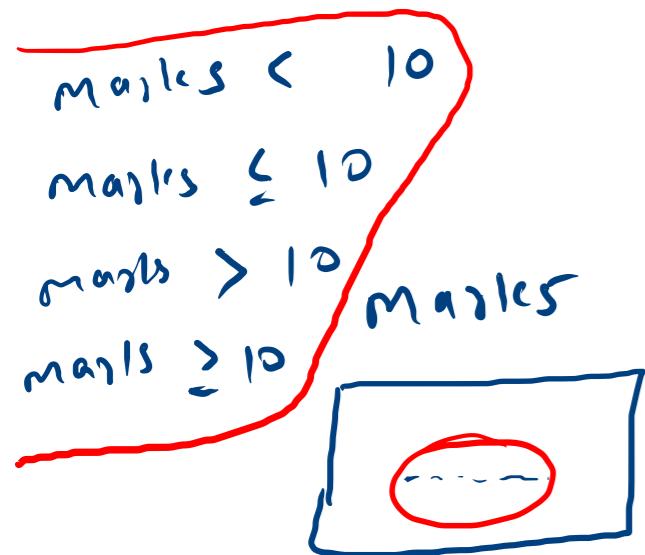
2.1 for marks above 90, print excellent.

2.2 for marks above 80 and less than equal to 90, print good.

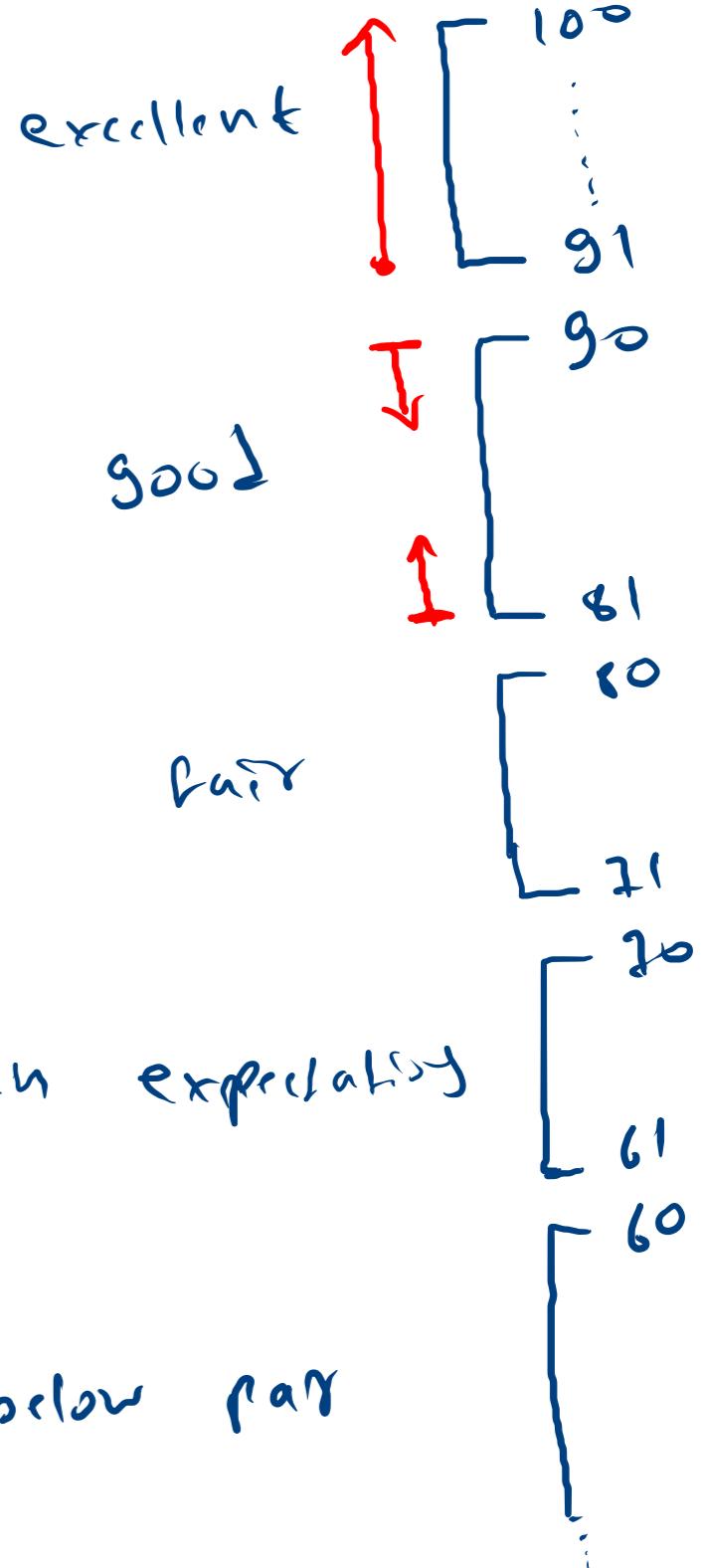
2.3 for marks above 70 and less than equal to 80, print fair.

2.4 for marks above 60 and less than equal to 70, print meets expectations.

2.5 for marks less than equal to 60, print below par.



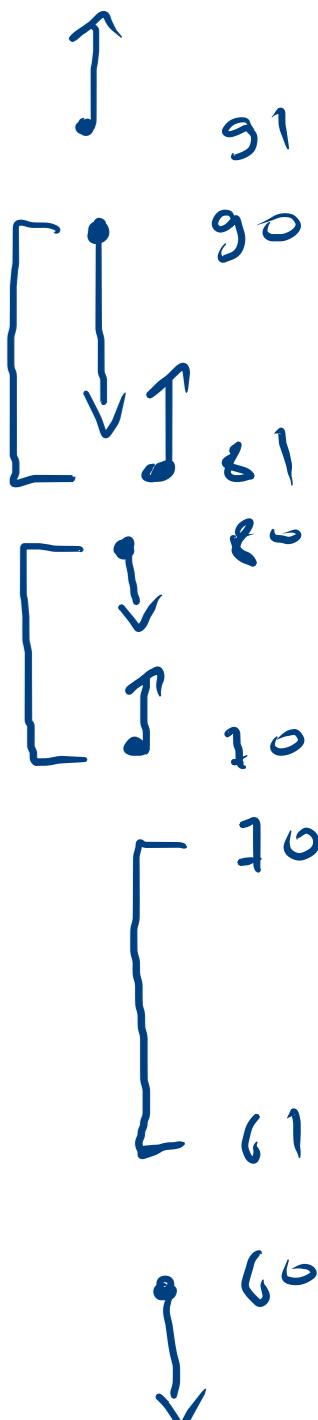
```
if (marks > 90) {  
    S.o.p ("excell")  
} else if (marks > 80) {  
    } else if ( ) {  
    } else {  
    S.o.p  
}
```



marks 90



excellent



```
int marks = scn.nextInt();
// input - don't change this code

// code here
if( marks > 90 ){
    System.out.println("excellent");
} else if( marks > 80 ){
    System.out.println("good");
} else if(marks > 70){
    System.out.println("fair");
} else if(marks > 60){
    System.out.println("meets expectations");
} else{
    System.out.println("below par");
}
```

```
    ← S.O.P ("*");  
[  
  ← S.O.P ("*");  
  ← S.O.P ("*");
```

*
*
*
*:
100 times

S.O.P ('*')
"
"
","

0

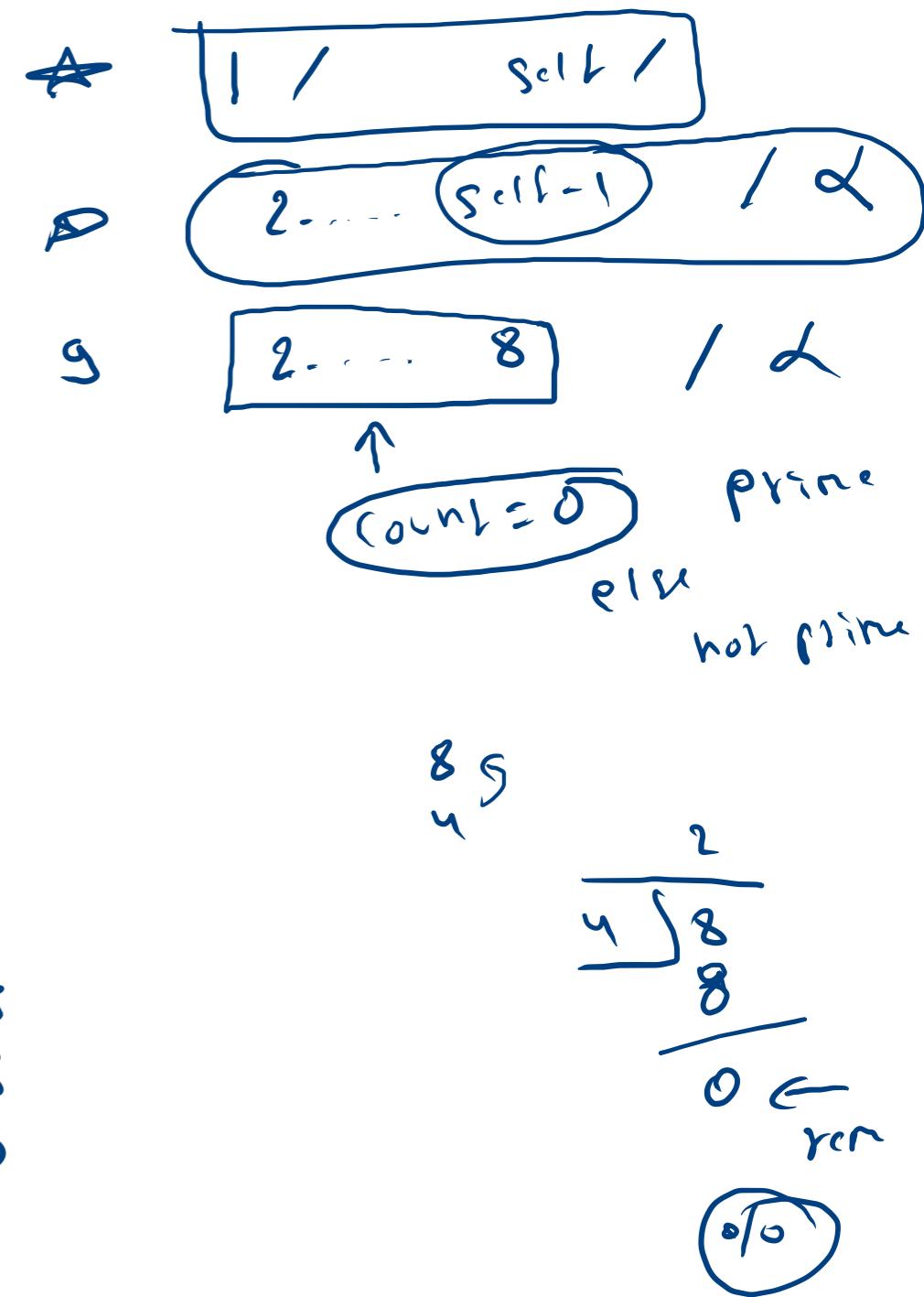
if (-- >) {
 // true
}

int n = 5;

while (n > 0) {
 System.out.println ("*");
 n = n - 1;

n = n - 1
n = 4 - 1
n = 3

1. You've to check whether a given number is prime or not.
2. Take a number "t" as input representing count of input numbers to be tested.
3. Take a number "n" as input "t" number of times.
4. For each input value of n, print "prime" if the number is prime and "not prime" otherwise.



5 13 2 3 4 5

2...5n

100

```
int t = scn.nextInt();
while( t > 0){
    int n = scn.nextInt();
    // code to check if n is prime

    int div = 2;
    int count = 0;

    while(div < n){
        if(n%div == 0){
            count = count+1;
        }
        div = div+1;
    }

    if(count == 0){
        System.out.println("prime");
    }else{
        System.out.println("not prime");
    }

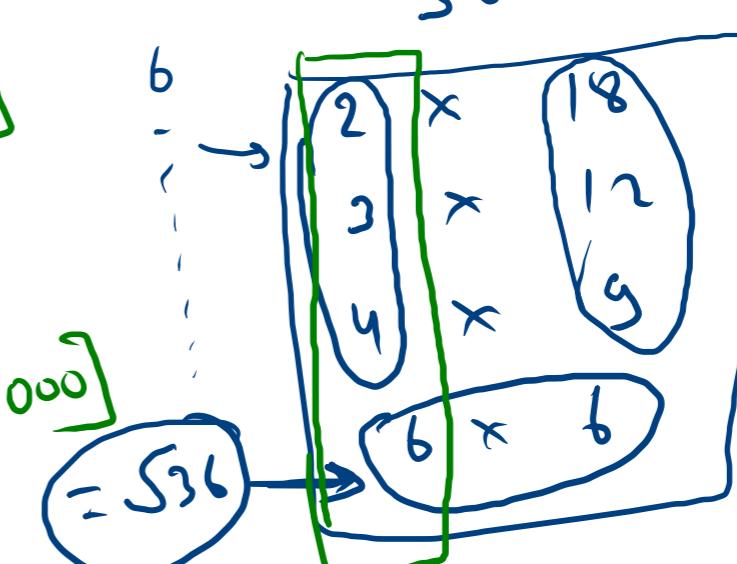
    t = t-1;
}
```

13 → prime
2 → prime
3 → prime
n → not prime

t = 5 * 3 * 2 * 0

div <= \sqrt{n}
(div^2) <= (n)^2
div * div <= n

10000
[2, 10000] [2, 100]
100000000
[2, 100000000] [2, 10000]



a * b
a < S36
b > S36
a = S36
b = S36

(a * b) = n

a > S36

b > S36

~~N.o~~ Print All Primes Till N

low = 5

high = 13

[5, 13]

6 ← low
24 ← high

[6, 24]

[5
7
11
13]

Sample :

7
11
13
17
19
23

input

$n=10$

fibonacci sequence

0 1 1 2 3 5 8 13 21 34

$$n \in \mathbb{N} = (n-1) + (n-2)$$

0 1 1 2 3 5 8 13 ...

0 1 1 2 3 5 8 13 21 34

c

a/b
 a b

$$\begin{aligned} a &= 0 \\ b &= 1 \end{aligned}$$

$$\begin{cases} c = a+b \\ a = b \\ b = c \end{cases}$$

```

Scanner scn = new Scanner(System.in);
int n = scn.nextInt();
int a=0;
int b=1; 3>0
while( n > 0){
    // code
    int c = a+b;
    a=b;
    b=c;
    n = n-1;
}

```

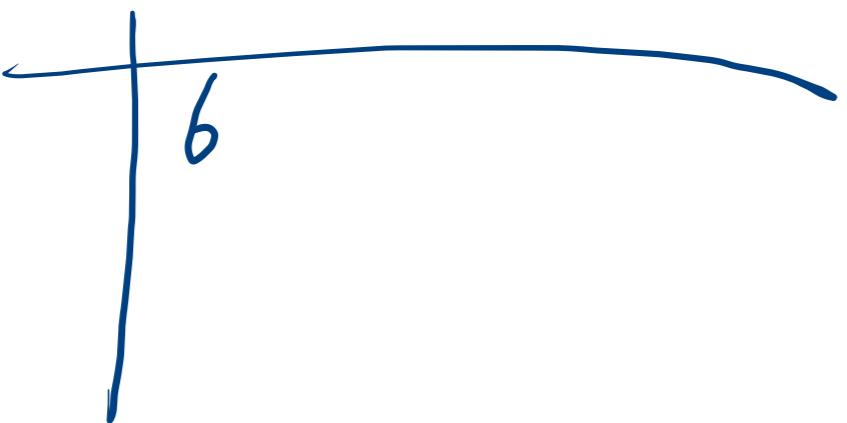
$n = 3 \times 3$
 $= 9$
 $a = 1$
 $b = 1$

~~a~~ . 1 1 2
~~b~~ ~~a~~
a b

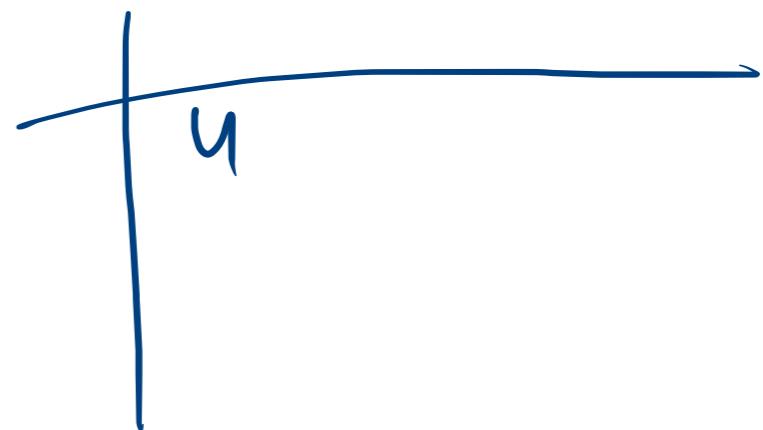
✓ ✓ ✓ ✓
0 1 1 2
a b

h

358262

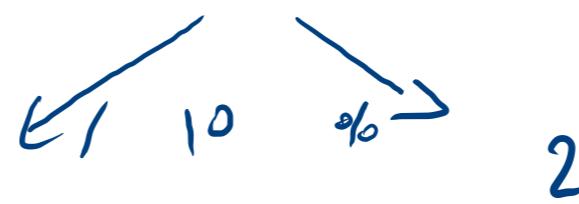


5462

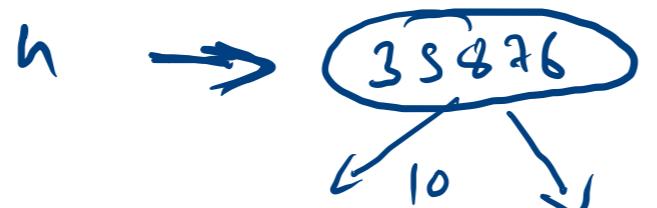


$n \rightarrow 358262$

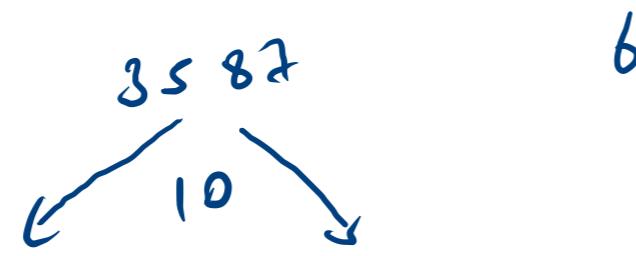
Count 0



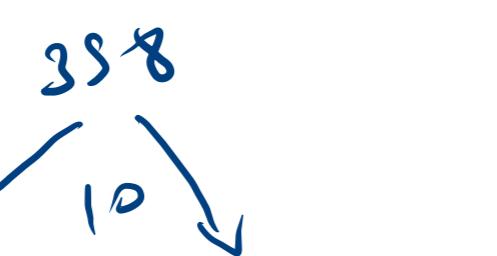
1



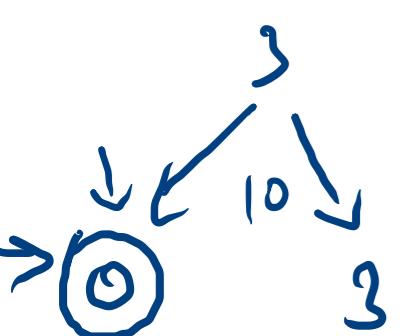
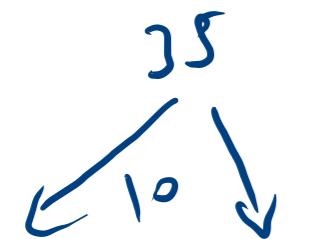
6



7



8



```

Scanner scn = new Scanner(System.in);
int n = scn.nextInt();
int count = 0;
while( n > 0){  

    int q = n/10;  

    int r = n%10;  

    // count = count+1;  

    count++;  

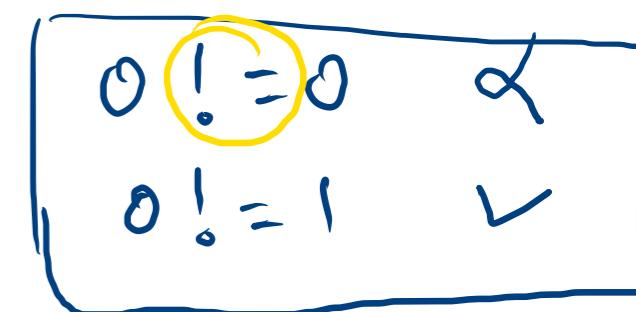
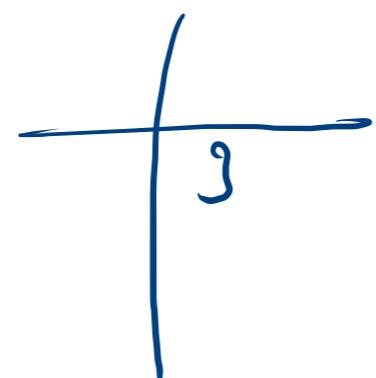
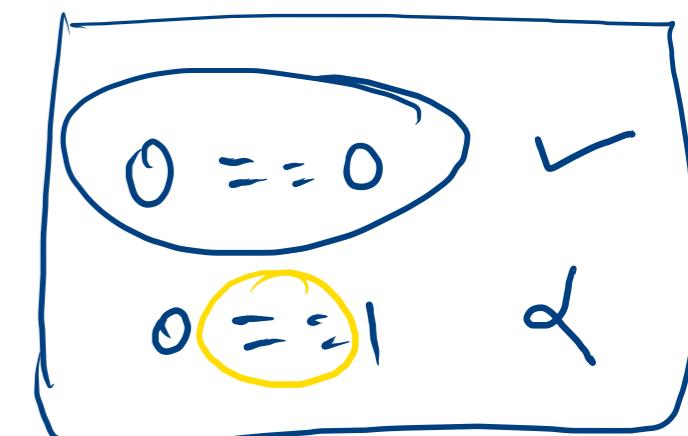
    n = q;  

}
System.out.println(count);

```

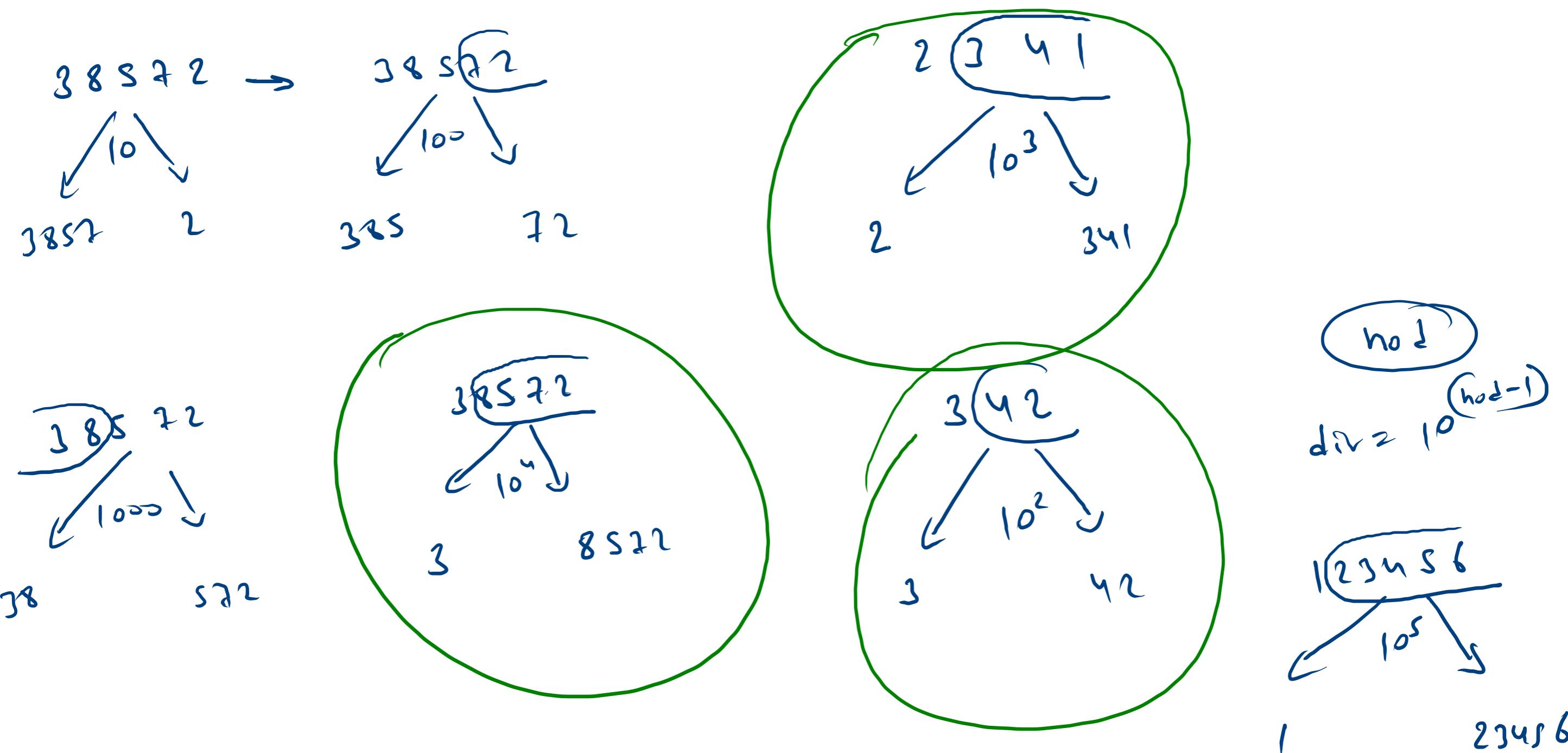
$n = \cancel{3} \cancel{4} \cancel{2} \quad \cancel{3} \cancel{4} \cdot \cancel{2} \cdot 0$
 $\text{count} = 0 + 2 + 3$

~~$q = 0$~~
 ~~$r = 3$~~



$$\begin{array}{r} \overbrace{65784383}^{\rightarrow} \\ \times 3834 \\ \hline 200 \\ 00 \\ \hline 2 \end{array}$$

6
5
7
8
4
3
8
2



$$\begin{array}{l} \delta_{en} = 0 \\ \delta_{iv} = 0 \end{array}$$

7
5
3
4
6

$$\frac{10^4}{10} = 10^3$$

$$\frac{10^3}{10} = 10^2$$

$$q = 2$$

$$n = 2 \rightarrow 346$$

$$\delta = 5 \rightarrow 346$$

$$h_{02} = 5$$

$$\delta = 5 \rightarrow 346$$

$$10^3$$

$$s$$

$$346$$

$$10^2$$

$$3$$

$$46$$

$$10^1$$

$$4$$

$$6$$

$$10^0$$

$$6$$

$$0$$

$$0$$

$$0$$

$$0$$

$$0$$

$$0$$

$$0$$

$$0$$

$$0$$

$$0$$

$$h_{02} = 4$$

$$h_{02} = 3$$

$$h_{02} = 2$$

$$h_{02} = 1$$

$$h_{02} = 0$$

$$\begin{array}{c} 100 \\ 10^2 \\ - \\ 0 \\ 0 \\ 10^1 \\ 100 \\ 0 \end{array}$$

$$0$$

```

Scanner scn = new Scanner(System.in);

int n = scn.nextInt();

int nod = 0;
int copyN = n; 050 2
while(copyN>0){ 050 2
    copyN = copyN/10;
    nod++;
}

int div = (int)Math.pow(10, nod-1); 0!>0

while(div!=0){ 0!>0
    int q = n/div;
    int r = n%div;

    System.out.println(q);
    n=r;
    div = div/10;
}

```

$$h = \cancel{3}8\cancel{2}0 \quad \cancel{8}\cancel{2}0 \quad \cancel{2}0 \quad \cancel{0} \quad 0$$

$$h_{10} = 0 + 2 \times 10^4$$

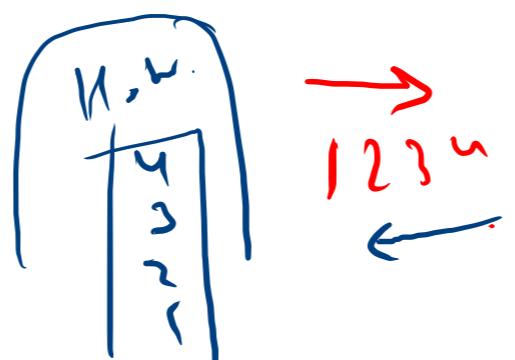
$$\text{copyN} = \cancel{3}8\cancel{2}0 \quad \cancel{3}8\cancel{2} \quad \cancel{2}0 \quad \cancel{0} \quad 0$$

$$\text{div} = -10^3 \quad 10^2 \quad 10^1 \quad 10^0 \quad 0$$

$$q = 0$$

$$r = 0$$

$$\frac{1}{10} > 0.1 \times = 0$$



1
2
3

3
8
7
0