Complete Beginner
For loop-1

Target: Hamand have diary, there are T numbers written in it. Hamend is confused how can he find last digit of every number?

Task → help hamand to find last digit of Every number.

Priyesh → How many numbers are there in diary?
↳ T differencent number.

[Iterate T times to find last digit of Every number.
ask from hamand for next number.
[print last digit from that number

T → different test cases
Iterate for every test case and solve the problem.

T- different numbers, we have to print all the digits for every different number?

all the digits

Ex → 3 → 1437 → 7, 3, 4, 1,
2012 → 2, 1, 0, 2,
176 → 6, 7, 1,

num = 1437

[Algorithm1]
↓
Print all the digit of a number.

```
while (num > 0) {
    ld = num % 10;
    SOP(ld + ", ");
    num = num / 10;
}
```

```
t = scn.nextInt();

while (t > 0) {
    sop(t);

    t--;
}
```

t= 5

5
4
3
2
1

Initialisation	condition	Loop work	update
t=5 t			
5	✓	✓	✓(4)
4	✓	✓	✓(3)
3	✓	✓	✓(2)
2	✓	✓	✓(1)
1	✓	✓	✓(0)
$\boxed{0}$ break point the loop			

```
t= scn.nextInt();

→ while (t >0) {

      int num= scn.nextInt();
      // print all the digits of "num" → Apply Algorithm!
      while ( num > 0) {
        int l.d = num %10;
            sop(ld + ", ");
            num = num /10;
      }
      SOP ln();

      t--;

}
```

Nested loop.

t	num	
3 →	1 4 3 7	7, 3, 4, 1,
2 →	2 0 1 2	2, 1, 0, 2,
1 →	1 7 6	6, 7, 1,
$\boxed{0}$ → Break point		

Problem: Reverse a number.

→ Reverse a number so that we can perform some Arithmatic operations on it

N → 61 23 ⟶ 3 21 6

N → 71 2 ⟶ 2 17

N → 10 ⟶ 1

N → 270 ⟶ 0 7 2 = 72

add 'd' digit to back of 'm' digit

'7' 2.

2 ⟶ 2*10 + 7 = 20+7 = 27

Result = 0

6 1 3 4

0*10 +4 = 04

4*10+3 = 43

43*10+1 = 431

431*10+6 = 4316

Repititive task

① take the last digit in effect of reverse number.

② Reducing the value of N.

N = 7834.

rev = 0

rev = rev*10 + (7834 % 10)

rev = 0*10 + 4 = 4

N = 7834 / 10 = 783

→ rev = rev * 10 + (783 % 10)

rev = 4* 10 + 3 = 43

→ N = 783 / 10 = 78

rev = rev * 10 + (78 % 10)

rev = 43*10 + 8 = 438

N = 78 / 10 = 7

rev = rev * 10 + (7 % 10)

rev = 438*10 + 7 = 4387

N = 7/10 = 0 → Break point

```
int num = 7348;
int rev = 0;

while (num > 0) {
    ld = num % 10   → (4)
    // add last digit after rev
    rev = rev * 10 + ld;   → 0*10 + 8 = 8
                             8*10 + 4 = 84
    num = num / 10;
}
                (7)
SOP(rev);
```

$\frac{7348}{10} = 734 + \frac{8}{10}$ → Reminder

734 → completed

7348 % 10 → Remainde

Gives us Remainde

```
        734
10 ) 7348 (
     70↓
     ---
      34
      30
     ---
       48
       40
      ---
        8
```

734 = 734/10 = 73

843

Structure of while loop

```
// 1.
while ( // 2. ) {        cond
    // 3.
    // 4.
}
```

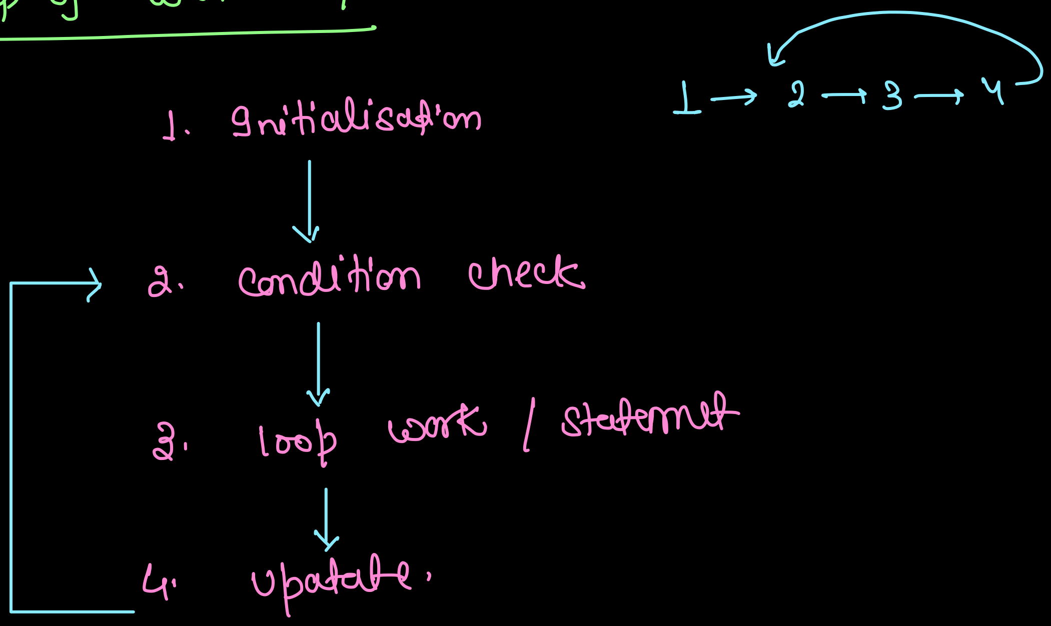
1. → Initialisation
2. → condition check
3. → loop work / loop statement
4. → Update

Two suggestion

① Enter in full screen mode

② Decrease your screen size at 80%

Execution step of while loop.



Is there , any different structure is available to perform same task → for loop

Structure of for loop →

```
for( initialisation ; condition ; update) {
                       check
        loop work / statement

}
```

Print all ther numbers from 1 to 10. →

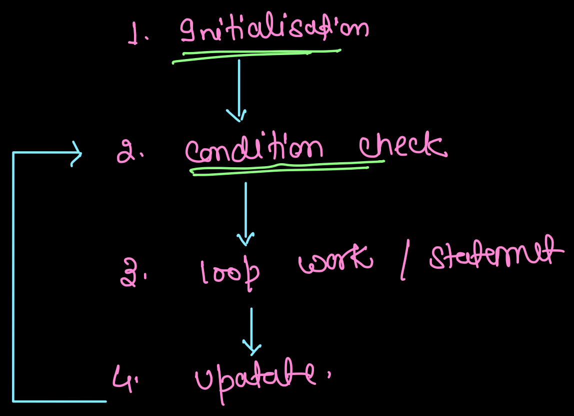
```
                                  ⇒ i=i+1
for(int i=1; i<=10; i++) {
      SOP(i);
}
```

$\boxed{N=10}$

```
for(i=1; i >= N; i++)
   SOP(i);
}
```

→ Nothing print

Execution steps for for-loop:



```
            initialisation          update
for( int i= 1; i <=10; i++) {
                condition check
  SOP(i);
   └→ loop work/ statement
}
```

Initialisation i=1	condition Check	loop work	Update
1			
1	✓	✓	✓ (2)
2	✓	✓	✓ (3)
3	✓	✓	✓ (4)
4	✓	✓	✓ (5)
5	✓	✓	✓ (6)
6	✓	✓	✓ (7)
7	✓	✓	✓ (8)
8	✓	✓	✓ (9)
9	✓	✓	✓ (10)
10	✓	✓	✓ (11)
11	X		

└→ Break point of loop

① we can declare a variable when we are initialising it.

② initialisation, condition check and update. → we can do it in single line

Printing → 1 2 3 4 5 6 7

8 9 10

```
for( int i=1; i<=10; i= i+2){
    SOP(i);
}
```

i=1

Initialisation	condition check	loop work	update
i			
1	✓	✓	✓(3)
3	✓	✓	✓(5)
5	✓	✓	✓(7)
7	✓	✓	✓(9)
9	✓	✓	✓(11)
11	11<=10 ↳ Break point		

Print

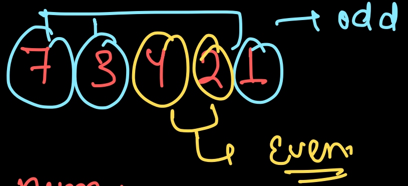
1
3
5
7
9

} all odd numbers from 1 to 10

Problem:

Print the odd number from 1 to N.

```
odd = 0
Even=0
while (i<=N){
        if (i%2 = =0) {
                Even += i;
        } else {
                odd += i;
        }
}
SOP (Odd + even)
```



```
num ;
while( num > 0) {
    ld = num %10;
    if( ld %2 = =0) {
        even += ld;
    } else {
        oddd += ld;
    }
    num = num/10
}
```