

## Basic Maths

### ① Count Digits

- 1- get a user input number
- 2- keep a variable count assigned to 0
- 3- now with a while condition & execute the while only if  $n > 0$
- 4- if the while cond is true then in the loop  
do  $n = n / 10$  (removes the last digit ex: ~~27~~  $27 / 10 = 2$ )
- 5- and increment the counter.
- 6- repeat the steps till while is false & return the counter value this tells total digits.

get a number and keep a counter to count the digits  
check if the number is greater than 0 keep dividing  
the number by 10 to remove the last digits & everytime  
a <sup>digit</sup> is removed counter increases till all the  
digits are removed & it becomes 0 then return the  
count value.



② check if each digit of  $N$  is divisible by  $N$   
ie if  $(N \% \text{digit} = 0)$

1- get a number

2- keep a counter to count how many digits are divisible

3- keep a copy of  $N$  & a digit variable

4- with while condition  $N > 0$

4.1- extract the last digit & store in the digit variable

4.2- and then remove the last digit.

4.3- if any of the digits is 0 skip its division  
bcoz  $N/0 = \text{run time error}$

4.4- if the digits divide  $N$  perfectly i.e.  $N \% \text{digit} = 0$   
then increase count  
and finally return count which tells how  
digits are divisible.

get a number fetch & remove the last  
digits if a digit is 0 skip division  
or else divide  $N$  by each digit & if its divisible  
then return the count.



### ③ reverse the number

- 1- get a number.
- 2- rev variable - to store the reversed number
- 3- digit variable - to store digits
- 4- check with while condition till the num is 0
- 4.1- get the digits
- 4.2-  $rev = rev \times 10 + digit$   
 $123 = 0 \times 10 + 3 = 3$   
 $12 = 3 \times 10 + 2 = 32$   
 $1 = 32 \times 10 + 1 = 321$
- 4.3- remove the last digit then return rev

### ④ Palindrome

Get a number and create a copy of the number  
do the same steps as above on the copy of the  
number then compare the rev with original  
number if its same then palindrome else no



⑤ Armstrong number (153 - dig=3,  $1^3 + 5^3 + 3^3 = 153$ )

- 1- get a number
- 2- two copies of number (one for digit count, one for total sum)
- 3- get the count of digits
- 4- now ~~can~~ extract each digit and raise it to the power of count and add each digit  
$$\text{Sum} = \text{Sum} + \text{pow}(\text{digit}, \text{count})$$
- ~~5-~~ and remove the digits one by one
- 5- now check if the original num is equal to Sum

⑥

Factors

i	n/i
1	36
2	18
3	12
4	9
6	6

- 1- get a number
- 2- run the for loop from 1 to  $\sqrt{n}$
- 3- if  $n \% i == 0$  then store that in vector & if  $n/i == i$  then store that in vector then sort that vector
- 4- print out the vector



### ⑦ Prime

Just like the previous one check for divisors  
if it has only 2 divisors 1 & itself then prime.

### ⑧ GCD (Euclidean algo)

- 1- get two numbers  $a$  &  $b$
- 2- while both are  $> 0$
- 3- & whichever is greater do mod of that  
if  $a > b \rightarrow a = a \% b$   $\left\lfloor \frac{\log(\min(a,b))}{(\log b)} \right\rfloor$   $a = a - b$   $TC = \min(a,b)$   
if  $b > a \rightarrow b = b \% a$   $b = b - a$
- 4- keep doing that till ~~one~~  $a$  or  $b$  becomes 0  
& return the non zero number i.e the GCD.