Binary to Decimal

Problem Description: Given a binary number as an integer N, you need to convert it into decimal and print it.
For example, if the given binary number is 1100, then the decimal number corresponding to this number is 12, so, print it as output.

How to approach?
A binary number can be converted into a decimal number by picking up the last digit and then multiplying each digit with 2 raised to the power of its place value and then adding them in a continuous manner.
For example, in the case of 1100, start picking up the last digit multiplied by 2's respective powers and add them up.So,
1100=0*2^(0)+0*2^(1)+1*2^(2)+1*2^(3)=0+0+4+8=12

1. Take the number as input from the user.
2. Now, initialize the decimal number (this will be our running sum) by 0 and take another variable pow for powers of 2 and initialize it with 1.
3. Now, run a loop until the number is greater than 0.
4. In each iteration, take the last digit by taking modulo 10 of the number. Find the decimal number by maintaining a running sum of multiplication of last digit and pow.
5. In each iteration multiply pow by 2 and divide num by 10.

Pseudo Code for this problem:

```
Input = number
decimal=0, pow=1
While number is greater than 0:
        last = number % 10
        decimal += last * pow
        pow *= 2
        number = number / 10
print(decimal)
```

❑ Let us dry run the code:
number=1100

- decimal=0, pow=1
- last=0
  decimal=0*1=0
  pow=1*2=2
  number=110.
- last=0
  decimal=0+0*2=0
  pow=2*2=4
  number=11.
- last=1
  decimal=0+1*4=4
  pow=4*2=8
  number=1.
- last=1
  decimal=4+1*8=12
  pow=8*2=16
  number=0.
- Now, number=0, so print decimal which is 12.
- Final Output:
  12