Time taken	24 mins 22 secs
Marks	5.00/5.00
Grade	**100.00** out of 100.00

Write an algorithm to find the discount value for the given total bill amount.

Constraints

1 <= orderValue< 10e100000

Input

The input consists of an integer orderValue, representing the total bill amount.

Output

Print an integer representing the discount value for the given total bill amount.

Example Input

578

Output

12

For example:

Test	Result
print(christmasDiscount(578))	12

Answer: (penalty regime: 0 %)

Reset answer

```
def is_prime_digit(digit):
   #Check if the digit is a prime number (2, 3, 5, or 7)
   return digit in {'2', '3', '5', '7'}
def christmasDiscount(n):
    # Convert the total bill amount to a string
    orderValue_str=str(n)

    discount=0

    #Iterate through each digit of the total bill amount

    for digit in orderValue_str:
        #Check if the digit is a prime number
        if is_prime_digit(digit):
             discount+=int(digit)

    return discount
```

	Test	Expected	Got	
✓	print(christmasDiscount(578))	12	12	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

complete the function which takes a number n as input and checks if it's an ugly number.

return ugly if it is ugly, else return not ugly

Hint:

An ugly number U can be expressed as: U = 2^a * 3^b * 5^c, where a, b and c are nonnegative integers.

For example:

Test	Result
`print(checkUgly(6))`	`ugly`
`print(checkUgly(21))`	`not ugly`

Answer: (penalty regime: 0 %)

Reset answer

```
def checkUgly(n):
    while n%2==0:
        n//=2
    while n%3==0:
        n//=3
    while n%5==0:
        n//=5
    return "ugly" if n == 1 else "not ugly"

```

	Test	Expected	Got	
✓	`print(checkUgly(6))`	`ugly`	`ugly`	✓
✓	`print(checkUgly(21))`	`not ugly`	`not ugly`	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Take an input integer from stdin.

Output Format:

Print TRUE or FALSE.

Example Input:

1256

Output:

TRUE

Example Input:

1595

Output:

FALSE

For example:

Test	Result
print(productDigits(1256))	True
print(productDigits(1595))	False

Answer: (penalty regime: 0 %)

Reset answer

```
def productDigits(n):
    s=str(n)
    l=len(s)
    e=1
    o=0
    for i in range(l):
        if ((i+1)%2==0):
            e*=int(s[i])
        else:
            o+=int(s[i])
    #print(e%o==0)
    return e%o==0

```

	Test	Expected	Got	
✓	print(productDigits(1256))	True	True	✓
✓	print(productDigits(1595))	False	False	✓

Passed all tests! ✓

Correct

of odd and even position digits.

Input Format:

Take a number in the form of String from stdin.

Output Format:

Print the difference between sum of even and odd digits

Example input:

1453

Output:

1

Explanation:

Here, sum of even digits is 4 + 3 = 7

sum of odd digits is 1 + 5 = 6.

Difference is 1.

Note that we are always taking absolute difference

Answer: (penalty regime: 0 %)

Reset answer

```
def differenceSum(n):
    n=str(n)
    a=0
    b=0
    for i, digit in enumerate(n):
        if i%2==0:
            a=a+int(digit)
        else:
            b=b+int(digit)
    return (abs(a-b))

```

	Test	Expected	Got	
✓	`print(differenceSum(1453))`	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

If it is an automorphic number display “Automorphic” else display “Not Automorphic”

Input Format:

Take a Integer from Stdin Output Format: Print Automorphic if given number is Automorphic number,otherwise Not Automorphic Example input: 5 Output: Automorphic Example input: 25 Output: Automorphic Example input: 7 Output: Not Automorphic

For example:

Test	Result
`print(automorphic(5))`	`Automorphic`

Answer: (penalty regime: 0 %)

Reset answer

```
def automorphic(n):
    c=len(str(n))
    r=(n**2)%(10**c)
    if(r==n):
        return 'Automorphic'
    else:
        return 'Not Automorphic'

```

	Test	Expected	Got	
✓	`print(automorphic(5))`	`Automorphic`	`Automorphic`	✓
✓	`print(automorphic(7))`	`Not Automorphic`	`Not Automorphic`	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ Week9_MCQ

Jump to...

Searching ►