vl dharshini
231501038
aiml a

WEEK 9

1) An e-commerce company plans to give their customers a special discount for Christmas.

They are planning to offer a flat discount. The discount value is calculated as the sum of all

the prime digits in the total bill amount.

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

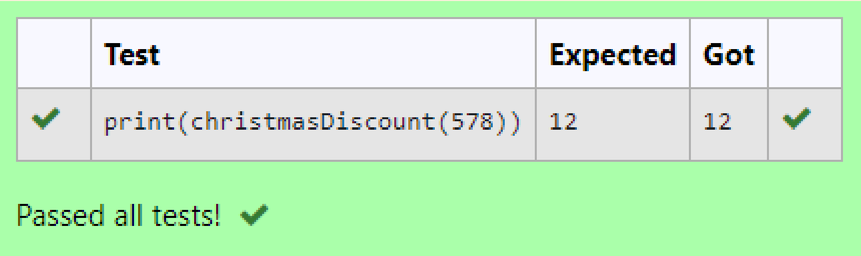
12

PROGRAM:

```
def is_prime_digit(digit):
   return digit in [2,3,5,7]
def christmasDiscount(n):
   discount = 0
   print_digits = [2,3,5,7]
   for digit in str(n):
      digit = int(digit)
      if is_prime_digit(digit):
         discount += digit

   return discount
```

OUTPUT:



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

Passed all tests! ✓

2) An abundant number is a number for which the sum of its proper divisors is greater than

the number itself. Proper divisors of the number are those that are strictly lesser than the number.

Input Format:

Take input an integer from stdin

Output Format:

Return Yes if given number is Abundant. Otherwise, print No

Example input:

12

Output:

Yes

Explanation

The proper divisors of 12 are: 1, 2, 3, 4, 6, whose sum is 1 + 2 + 3 + 4 + 6 = 16. Since sum of

proper divisors is greater than the given number, 12 is an abundant number.

Example input:

13

Output:

No

Explanation

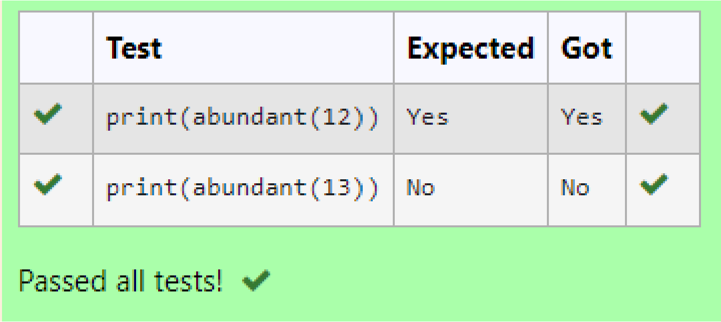
The proper divisors of 13 is: 1, whose sum is 1. Since sum of proper divisors is not greater

than the given number, 13 is not an abundant number.

PROGRAM:

```
def abundant(number):
    divisor_sum=sum([divisor for divisor in range(1,number) if number % divisor==0])
    if divisor_sum>number:
        return "Yes"
    else:
        return "No"
```

OUTPUT:



	Test	Expected	Got	
✓	`print(abundant(12))`	Yes	Yes	✓
✓	`print(abundant(13))`	No	No	✓

Passed all tests! ✓

3) Given a number with maximum of 100 digits as input, find the difference between the sum

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.

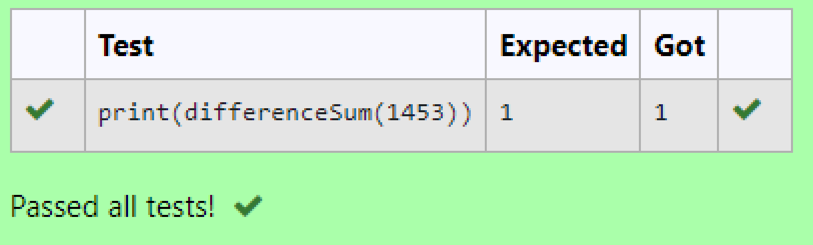
Difference is 1.

Note that we are always taking absolute difference

PROGRAM:

```
def differenceSum(number_str):
   number_str=str(number_str)
   odd_sum = 0
   even_sum = 0
   for index, char in enumerate(number_str):
      digit = int(char)
      if (index + 1) % 2 == 0:
         even_sum += digit
      else:
         odd_sum += digit
   difference = abs(even_sum - odd_sum)
   return difference
   print(differenceSum(number_str))
```

OUTPUT:



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

Passed all tests! ✓

4) Write a code to check whether product of digits at even places is divisible by sum of digits

at odd place of a positive integer.

Input Format:

Take an input integer from stdin.

Output Format:

Print TRUE or FALSE.

Example Input:

1256

Output:

TRUE

Example Input:

1595

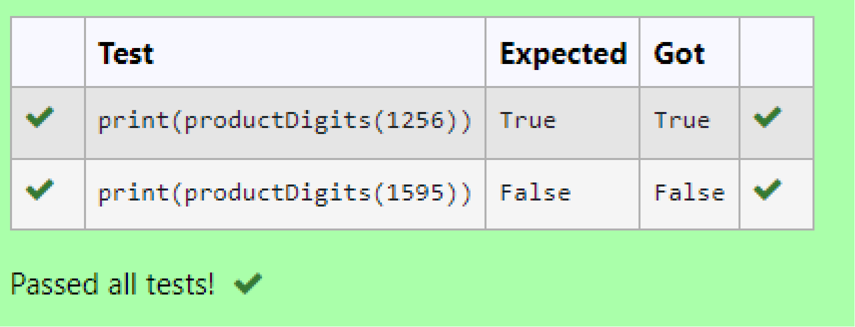
Output:

FALSE

PROGRAM:

```
def productDigits(number):
    number_str = str(number)
    product_even = 1
    sum_odd = 0
    for i, digit_char in enumerate(number_str):
        digit = int(digit_char)
        if (i+1)%2 == 0:
            product_even *= digit
        else:
            sum_odd += digit
    return product_even % sum_odd == 0
```

OUTPUT:



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

Passed all tests! ✓

5) A number is considered to be ugly if its only prime factors are 2, 3 or 5.

[1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, ...] is the sequence of ugly numbers.

Task:

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.

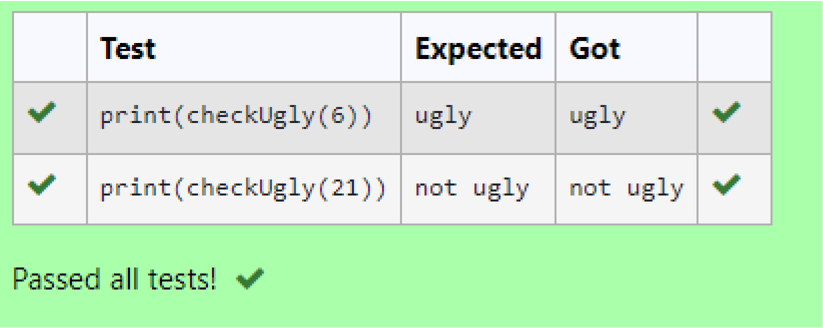
PROGRAM:

```
def checkUgly(n):
    if n <= 0:
        return "not ugly"

    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"
```

OUTPUT:



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

Passed all tests! ✓