

PROGRAM:

```
a=str(n)  
s,p=0,1  
for i in range(0,len(a),2):  
    s+=int(a[i])  
for i in range(1,len(a),2):  
    p*=int(a[i])  
if(p%s==0):  
    return("True")  
else:  
    return("False")
```

OUTPUT:

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

Passed all tests! ✓

Correct
Marks for this submission: 1.00/1.00.

Ex. No. : 7.4

Date:

Register No: 231501049

Name: GNAANESH B B

Christmas Discount

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 python code to find the discount value for the given total bill amount.

Constraints

$1 \leq \text{orderValue} < 10^{100000}$

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
<code>print(christmasDiscount(578))</code>	12

PROGRAM:

```
def christmasDiscount(n):
    res=0
    while n!=0:
        rem=n%10
        flag=0
        for i in range(1,rem+1):
```

```
if rem%i==0:  
    flag+=1  
  
if flag==2:  
    res=res+rem  
  
n=n//10  
  
return res
```

OUTPUT:

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

Passed all tests! ✓

Correct
Marks for this submission: 1.00/1.00.

Ex. No. : 7.5

Date:

Register No: 231501049

Name: GNAANESH B B

Coin Change

complete function to implement coin change making problem i.e. finding the minimum number of coins of certain denominations that add up to given amount of money.

The only available coins are of values 1, 2, 3, 4

Input Format:

Integer input from stdin.

Output Format:

return the minimum number of coins required to meet the given target.

Example Input:

16

Output:

4

Explanation:

We need only 4 coins of value 4 each

Example Input:

25

Output:

7

Explanation:

We need 6 coins of 4 value, and 1 coin of 1 value

PROGRAM:

```
def coinChange(amount):
    # Available coin denominations
    coins = [1, 2, 3, 4]
    # Initialize a list to store the minimum number of coins for each amount
    # from 0 to the target amount
```

```

dp = [float('inf')] * (amount + 1)

dp[0] = 0 # Base case: 0 coins needed to make amount 0

# Iterate through all amounts from 1 to the target amount

for i in range(1, amount + 1):

    # Iterate through all available coin denominations

    for coin in coins:

        # If the current coin denomination is less than or equal to the current
        # amount

        if coin <= i:

            # Update dp[i] to be the minimum between its current value and
            # dp[i - coin] + 1

            dp[i] = min(dp[i], dp[i - coin] + 1)

# The result is stored at dp[amount]

return dp[amount]

amount = int(input())

print(coinChange(amount))

```

	Test	Expected	Got
✓	print(coinChange(16))	4	4

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Ex. No. : **7.6**

Date:

Register No: **231501049**

Name: **GNAANESH B B**

Difference Sum

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(n):
    a=[]
    b=[]
    k=str(n)
    for i in range(len(k)):
        if int(i)%2==0:
            a.append(int(k[i]))
```

```
else:  
    b.append(int(k[i]))  
  
s=sum(b)  
  
r=sum(a)  
  
j=s-r  
  
return j
```

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

Passed all tests! ✓

Correct
Marks for this submission: 1.00/1.00.

