07- Functions

Ex.No.   :   7.1   **Date: 11-04-2024**

Register No.: 2116231501072   **Name: Kanishka P**

ʹ

AbundantNumber

Anabundantnumberisanumberforwhichthesum   ofitsproperdivisorsisgreater thanthenumberitself.Properdivisorsofthenumberarethosethatarestrictly lesser than the number.

InputFormat:
Takeinputanintegerfromstdin
OutputFormat:
ReturnYesifgivennumberisAbundant.Otherwise,print No
Exampleinput:
12
Output:
Yes
Explanation
The properdivisors of 12 are: 1,2, 3, 4,6, whose sum is1 +2 + 3 + 4 + 6 = 16.Since sumofproperdivisorsisgreaterthanthegivennumber,12isanabundantnumber.
Example input:
13
Output:
No
Explanation
The proper divisors of 13is: 1, whose sumis 1. Since sumof proper divisors is not greater than the given number, 13 is not an abundant number.

For example:

Test	Result
print(abundant(12))	Yes
print(abundant(13))	No

Program:

```
defabundant(n):

    l,s=[],0

    for i in range(1,int(n//2)+1):

        if(n%i==0):
```

```
        l.append(i)fo

r i in l:

    s+=i

if(s>n):

    return("Yes")

else:

    return("No")
```

Output:

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

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

<u>Automorphicnumberornot</u>

An automorphic number is a number whose square ends with the number itself. For example, 5 is an automorphic number because 5*5 =25. The last digit is 5 which sameas the given number.

Ifthenumberisnot valid,itshould display"Invalidinput".
Ifitisanautomorphicnumberdisplay"Automorphic"elsedisplay"NotAutomorphic".

InputFormat:
TakeaIntegerfromStdin
Output Format:
PrintAutomorphicifgivennumberisAutomorphicnumber,otherwiseNot Automorphic
Exampleinput:5Output:AutomorphicExampleinput:25Output:Automorphic Example
input: 7 Output: Not Automorphic
For example:

Test	Result
print(automorphic(5))	Automorphic

Program:

```
defautomorphic(n):
    a=str(n*n)
    if(int(a[-1])==n):
        return("Automorphic")
    else:
        return("NotAutomorphic")
```

Output:

	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.

CheckProductofDigits

Writeacodetocheckwhetherproductofdigitsatevenplacesisdivisiblebysum of digits at odd place of a positive integer.
InputFormat:

Takeaninputintegerfromstdin.

Output Format:

Print TRUE or FALSE.

Example Input:

1256

Output:

TRUE

ExampleInput:

1595

Output:

FALSE
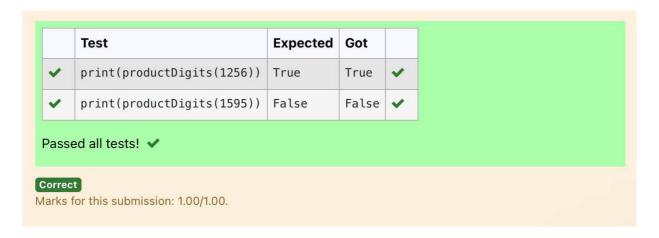
Forexample:

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

Program:

defproductDigits(n):

```python
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.

Christmas Discount

An e-commerce company plans to give their customers a special discount for Christmas.
Theyareplanningtoofferaflatdiscount.Thediscountvalueiscalculatedasthe sum of all the prime digits in the total bill amount.

Writeanpythoncodetofindthediscountvalueforthegiventotalbillamount.

Constraints

$1 <= orderValue < 10e^{100000}$ Input

TheinputconsistsofanintegerorderValue,representingthetotalbillamount. Output

Printanintegerrepresentingthediscountvalueforthegiventotalbillamount. Example

Input

578

Output

12

Forexample:

Test	Result
print(christmasDiscount(578))	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


    returnres
```

Output:

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

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

CoinChange

complete function to implement coin change making problem i.e. finding the minimum numberofcoinsofcertain denominationsthatadduptogivenamountofmoney.

Theonlyavailablecoinsareofvalues1,2,3,4 Input

Format:

Integerinputfromstdin.

Output Format:

returntheminimumnumberofcoinsrequiredtomeetthegiventarget.

Example Input:

16

Output:

4

Explanation:

Weneedonly4coinsofvalue4each

Example Input:

25

Output:

7

Explanation:

Weneed6coinsof4value,and1coinof1value

Program:

defcoinChange(amount):

   # Available coin

   denominationscoins = [1, 2, 3, 4]

   # Initialize a list to store the minimum number of coins for each amountfrom 0 to the target amount

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

```python
    dp[0]=0#Basecase:0coinsneededtomakeamount0


    # 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
thecurrent amount

            ifcoin<=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))
```

Output:

	Test	Expected	Go
✔	`print(coinChange(16))`	4	4

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

DifferenceSum

Given a number with maximum of 100 digits as input,find the difference betweenthe sum of odd and even position digits.

InputFormat:

TakeanumberintheformofStringfromstdin.

Output Format:

Print the difference between sum of even and odd digits

Example input:

1453

Output:

1

Explanation:

Here,sumofevendigitsis4+3=7 sum of

odd digits is 1 + 5 = 6.

Differenceis1.

Notethatwearealwaystakingabsolutedifference

Program:

```
def differenceSum(n):
    a=[]
    b=[]
    k=str(n)
    foriinrange(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

Output:

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

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

Uglynumber

Anumberisconsideredtobeuglyifitsonlyprimefactorsare2,3or5. [1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, …] is the sequence of ugly numbers.
Task:
completethefunctionwhichtakesanumbernasinputandchecksifit'sanugly number.
return ugly if it is ugly, else return not ugly
Hint:
AnuglynumberUcanbeexpressedas:U=2^a*3^b*5^c,wherea,bandcare nonnegative integers.
Forexample:

Test	Result
print(checkUgly(6))	ugly
print(checkUgly(21))	notugly

Program:

defcheckUgly(n):

    foriinrange(n):

    forjinrange(n):

    forkinrange(n):

        if(n==(2i)+(3**j)+(5**k)):**

        return("ugly")

    return("notugly")

Output:

	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.