**WEEK 12-Modules**

**1.** Given an integer n, print *true* if it is a power of four. Otherwise, print *false*.

An integer n is a power of four, if there exists an integer x such that n == 4x.

**PROGRAM:**

n = int(input())

if n <= 0:

print('False')

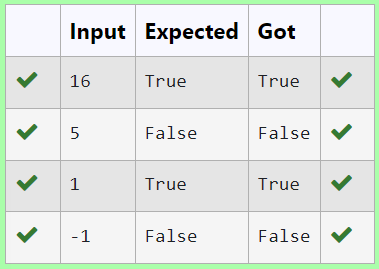
else:

while n % 4 == 0:

n //= 4

print(n == 1)

**OUTPUT:**

****

**2**. Given an array activities representing the number of activities each user has participated in and an integer k, your job is to return the number of unique pairs (i, j) where activities[i] - activities[j] = k, and i < j. The absolute difference between the activities should be exactly k.

**PROGRAM:**

def count\_unique\_pairs(n,activities,k):

unique\_pairs=set()

for i in range(n):

for j in range(i+1,n):

if abs(activities[i]-activities[j])==k:

unique\_pairs.add((i,j))

return len(unique\_pairs)

def main():

n=int(input())

activities=list(map(int,input().split()))

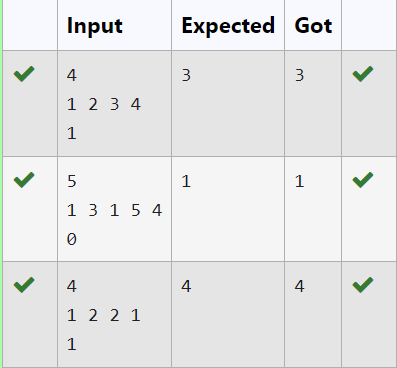
k=int(input())

result=count\_unique\_pairs(n,activities,k)

print(result)

main()

**OUTPUT:**

****

3. Create a Python-based solution that can parse input data representing a list of students with their respective marks and other details, and compute the average marks. The input may present these details in any order, so the solution must be adaptable to this variability.

**PROGRAM:**

def calculate\_avg\_marks(N,columns,std\_data):

total\_marks=0

num\_stud=0

marks\_index=columns.index("MARKS")

for student in std\_data:

marks=int(student[marks\_index])

total\_marks+=marks

num\_stud+=1

average\_marks=total\_marks/num\_stud if num\_stud>0 else 0

return average\_marks

N=int(input())

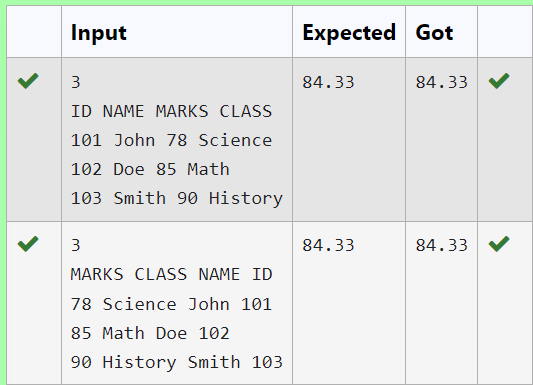
columns=input().split()

std\_data=[input().split() for \_ in range(N)]

average\_marks=calculate\_avg\_marks(N,columns,std\_data)

print("{:.2f}".format(average\_marks))

**OUTPUT:**

****

4. Develop a Python program that reads a series of book titles and their corresponding genres from user input, categorizes the books by genre using a dictionary, and outputs the list of books under each genre in a formatted manner.

**PROGRAM:**

d = {}

while True:

try:

book = input().split(',')

if len(book) < 2:

continue

book\_name = book[0].strip()

category = book[1].strip()

if category in d:

d[category].append(book\_name)

else:

d[category] = [book\_name]

except EOFError:

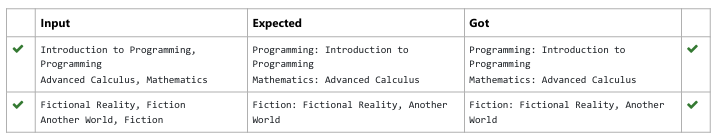
break

for k, v in d.items():

print(f"{k}: ", end='')

print(', '.join(v))

**OUTPUT:**



5. Develop a Python program that manages shoe inventory and processes sales transactions to determine the total revenue generated. The program should handle inputs of shoe sizes available in the shop, track the number of each size, and match these with customer purchase requests. Each transaction should only proceed if the desired shoe size is in stock, and the inventory should update accordingly after each sale

**PROGRAM:**

n = int(input())

shoes = list(map(int,input().split()))

N = int(input())

t = 0

for i in range(N):

a = list(map(int,input().split()))

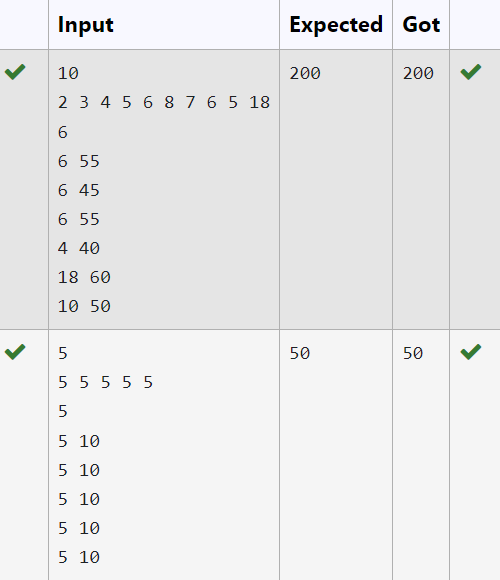
if a[0] in shoes:

shoes.remove(a[0])

t +=a[1]

print(t)

**OUTPUT:**

****