### **Ex. No. : 9.1** Christmas Discount

**Program:** def is\_prime\_digit(digit): return digit in [2,3,5,7] def christmasDiscount(n):

s=discount=0

prime\_digitis=[2,3,5,7] for digit in str(n): digit=int(digit) if is\_prime\_digit(digit):

discount+=digit return discount

### **Ex. No. : 9.2** Check Product of Digits

**Program:**

def productDigits(n): a=n temp=[] list1=[] list2=[] rem=0 while a!=0: rem=a%10 temp.append(rem) a=a//10 for i in range(len(temp)): if(i+1)%2==0: list1.append(temp[i]) else:

list2.append(temp[i]) pro=1 sum=0

for i in list1: sum+=i for i in list2: pro\*=i

if pro%sum==0:

return True else:

return False

### **Ex. No. : 9.3** Abundant Number

**Program:**

def abundant(number):

d\_s=sum([divisor for divisor in range(1,number) if number % divisor == 0]) if d\_s>number:

return"Yes" else:

return "No"

### **Ex. No. : 9.4** Ugly number

**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"

### **Ex. No. : 9.5** Automorphic number or not

Automorphic

**Program:**

def automorphic(n):

if(n<0):

return "Invalid

input"

square = n \* n n\_s=str(n) s\_s=str(square)

if s\_s.endswith(n\_s): return "Automorphic"

else:

return "Not Automorphic"