**Q.No.1.**

**Answer:**

Input: Sourpuss

Output: Ugwtrwuu

def main():

word = input("Please enter the text:").lower()

Value = int(input("Please enter the value of the key:"))

En\_or\_De = input("Press e(to encrypt) or d(to decrypt: ")

coded\_message = ""

for ch in word:

if En\_or\_De == "e":

coded\_message += chr((ord(ch) - 97 + Value)%26 + 97)

elif En\_or\_De =="d":

coded\_message += chr((ord(ch) - 97 - Value)%26 + 97)

print("The input:", word, "," "is coded as: ", coded\_message)

main()

**Q.No.2.**

**Answer: 4257**

lst = [4, 2, 5, 6]

i = [str(integer) for integer in lst]

a = "".join(i)

res = int(a)

a=1

sum=res+1

print(sum)

num = sum

print("The number is " + str(num))

res1 = [int(x) for x in str(num)]

print("List = " + str(res1))

**Q.No.3.**

**Answer: 1101**

def add\_binary\_nums(x,y):

max\_len = max(len(x), len(y))

x = x.zfill(max\_len)

y = y.zfill(max\_len)

result = ''

carryover = 0

for i in range(max\_len-1, -1, -1):

r = carryover

r += 1 if x[i] == '1' else 0

r += 1 if y[i] == '1' else 0

result = ('1' if r % 2 == 1 else '0') + result

carryover = 0 if r < 2 else 1

if carryover !=0 : result = '1' + result

return result.zfill(max\_len)

print(add\_binary\_nums('1011', '10'))

**Q.No.4.**

**Answer:**

**Input: 5**

**Output: 220**

**Input: 220**

**Output: 284**

**Input: 5000**

**Output: 5020**

import math

n = eval(input("The input is: "))

def div(ron):

sum = 0;

for i in range(1,ron):

if(ron%i==0):

sum=sum+i

return sum;

def amicable(iteration):

final = 0

for i in range(1,iteration):

currentDivs = div(int(i));

resultDivs = div(currentDivs);

if (i == resultDivs and currentDivs != resultDivs):

if(i > n):

print(i)

final = i

return i

break

if(final == 0):

amicable(iteration\*2)

def amc(n):

final\_result = amicable(1500);

amc(n)