**18CSC305J- ARTIFICIAL INTELLEGENCE**

**Experiment-3**

**Implementation of Constraint Satisfaction Problem - Cryptarithmetic puzzle**

**Team Ai 4 life**

Aayushi Goenka (RA1911031010151)

Kapuluru Srinivasulu (RA1911031010142)

Sonia Raja (RA1911031010152)

Sadekar Adesh (RA1911031010141)

Praneet Botke (RA1911031010149)

Sai Mohit Ambekar (RA1911031010137)

**Aim:**

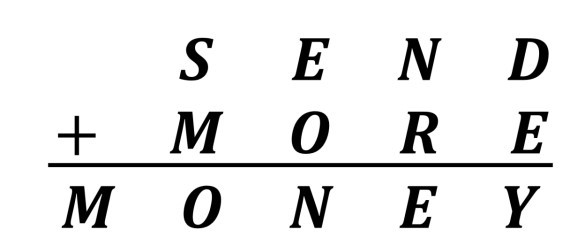
To implement the constraint satisfaction problem using cryptarithmetic puzzle (SEND + MORE=MONEY).

**Solution & Procedure:**

SEND + MORE=MONEY is an interesting Cryptarithmetic Problems

wherein the digits are replaced by letters of the alphabets. The objective

of this problem is to find a solution for



Therefore, we have to find a numerical value for {S, E, N, D, M, O, R, Y}

that satisfies the given question. The simplest way to solve this problem

will be to do permutation from 0-9 for all Letters and compute the above

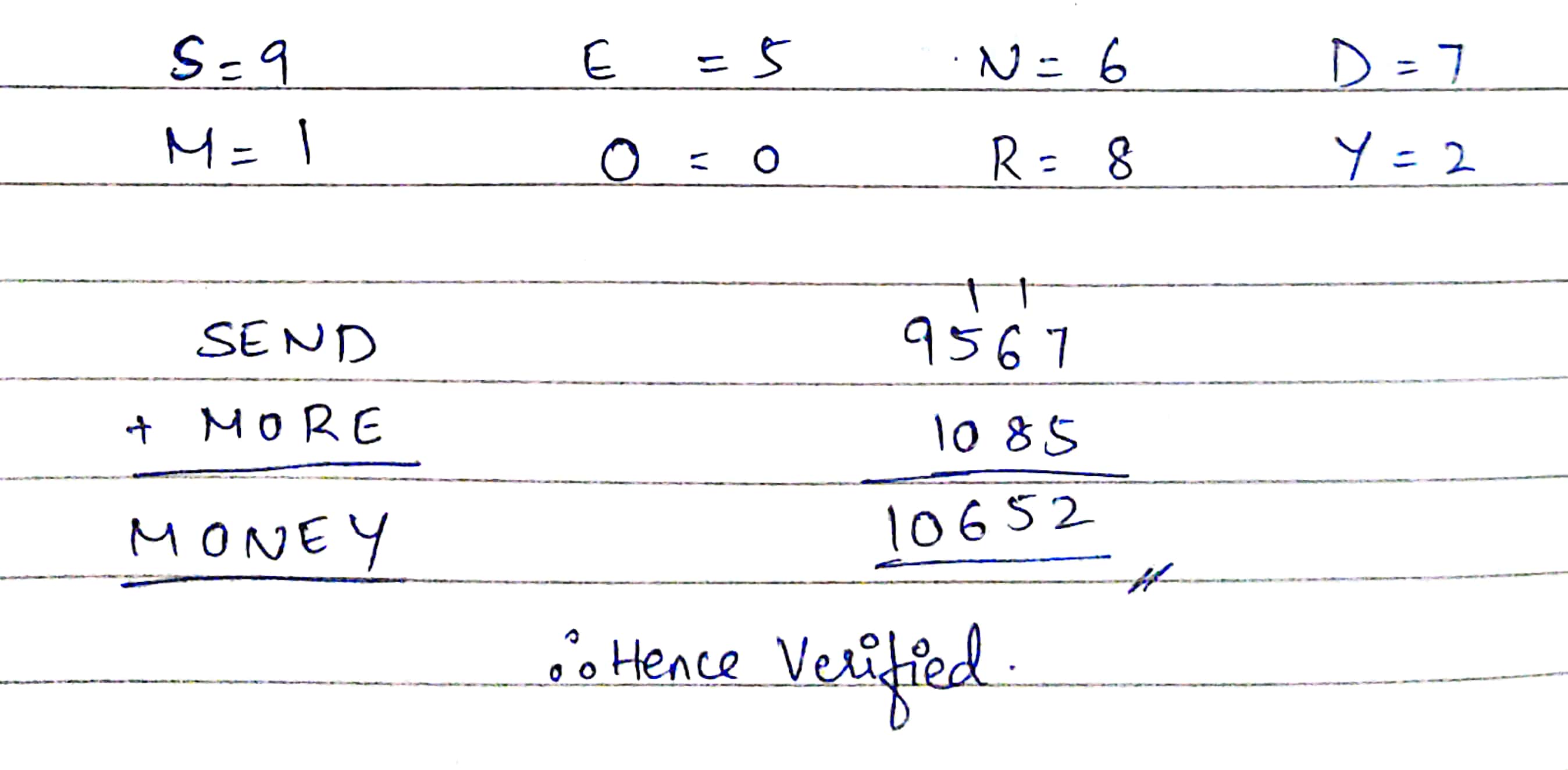
equation. However, this is definitely not optimizable as there will be a

significantly high time duration. We should also consider that S and M

not equal to 0, as they are the leading digits and leading digits cannot

be 0.

**Calculation:**

****

**Code:**

from itertools import combinations, permutations

def replacements():

for comb in combinations(range(10), 8):

for perm in permutations(comb):

if perm[0] \* perm[1] != 0:

yield dict(zip('SMENDORY', perm))

a, b, c = 'SEND', 'MORE', 'MONEY'

for replace in replacements():

f = lambda x: sum(replace[e] \* 10\*\*i for i, e in enumerate(x[::-1]))

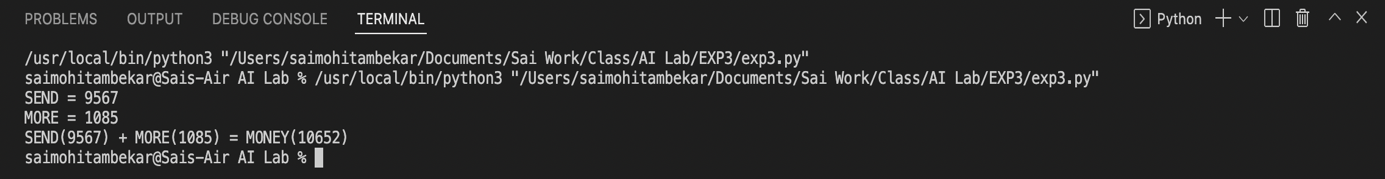
if f(a) + f(b) == f(c):

print("SEND = {}".format(f(a)))

print("MORE = {}".format(f(b)))

print("SEND({}) + MORE({}) = MONEY({})".format(f(a), f(b), f(c)))

**Output:**

****

**Result:**

Implemented constraint satisfaction problem using the Cryptarithmetic problem called SEND + MORE=MONEY.