**Csp problem- constraint satisfaction problems**

**LAB 3**

**AMRIT AANAND**

**RA1911003010917**

Lab 3: Implementation of constraint satisfaction problems  
(examples: SEND + MORE = MONEY, BASE + BALL = GAMES, TWO + TWO = FOUR, CROSS + ROADS = DANGER).

**1) SEND + MORE = MONEY**

5 4 3 2 1

S E N D

+ M O R E

c3 c2 c1

M O N E Y

1. From Column 5, M=1, since it is only carry-over possible from sum of 2 single digit number in column 4.
2. To produce a carry from column 4 to column 5 'S + M' is at least 9 so 'S=8or9' so 'S+M=9or10' & so 'O = 0 or 1'. But 'M=1', so 'O = 0'.
3. If there is c+rry from Column 3 to 4 then 'E=9' & so 'N=0'. But 'O = 0' so there is no carry & 'S=9' & 'c3=0'.
4. If there is no carry from column 2 to 3 then 'E=N' which is impossible, therefore there is carry & 'N=E+1' & 'c2=1'.
5. If there is carry from column 1 to 2 then 'N+R=E mod 10' & 'N=E+1' so 'E+1+R=E mod 10', so 'R=9' but 'S=9', so there must be c+rry from column 1 to 2. Therefore 'c1=1' & 'R=8'.
6. To produce carry 'c1=1' from column 1 to 2, we must h+ve 'D+E=10+Y' as Y cannot be 0/1 so D+E is at least 12. As D is at most 7 & E is

At least 5 (D cannot be 8 or 9 as it is already assigned). N is at most 7

& 'N=E+1' so 'E=5or6'.

1. If E were 6 & D+E at least 12 then D would be 7, but 'N=E+1' & N would also be 7 which is impossible. Therefore 'E=5' & 'N=6'.
2. D+E is at least 12 for that we get 'D=7' & 'Y=2'.

## SOLUTION:

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | 5 | 6 | 7 |
| + 1 | 0 | 8 | 5 |

1 0 6 5 2

CODE:

def solutions():

# letters = ('s', 'e', 'n', 'd', 'm', 'o', 'r', 'y')

all\_solutions = list()

for s in range(9, -1, -1):

for e in range(9, -1, -1):

for n in range(9, -1, -1):

for d in range(9, -1, -1):

for m in range(9, 0, -1):

for o in range(9, -1, -1):

for r in range(9, -1, -1):

for y in range(9, -1, -1):

if len(set([s, e, n, d, m, o, r, y])) == 8:

send = 1000 \* s + 100 \* e + 10 \* n + d

more = 1000 \* m + 100 \* o + 10 \* r + e

money = 10000 \* m + 1000 \* o + 100 \* n + 10 \* e + y

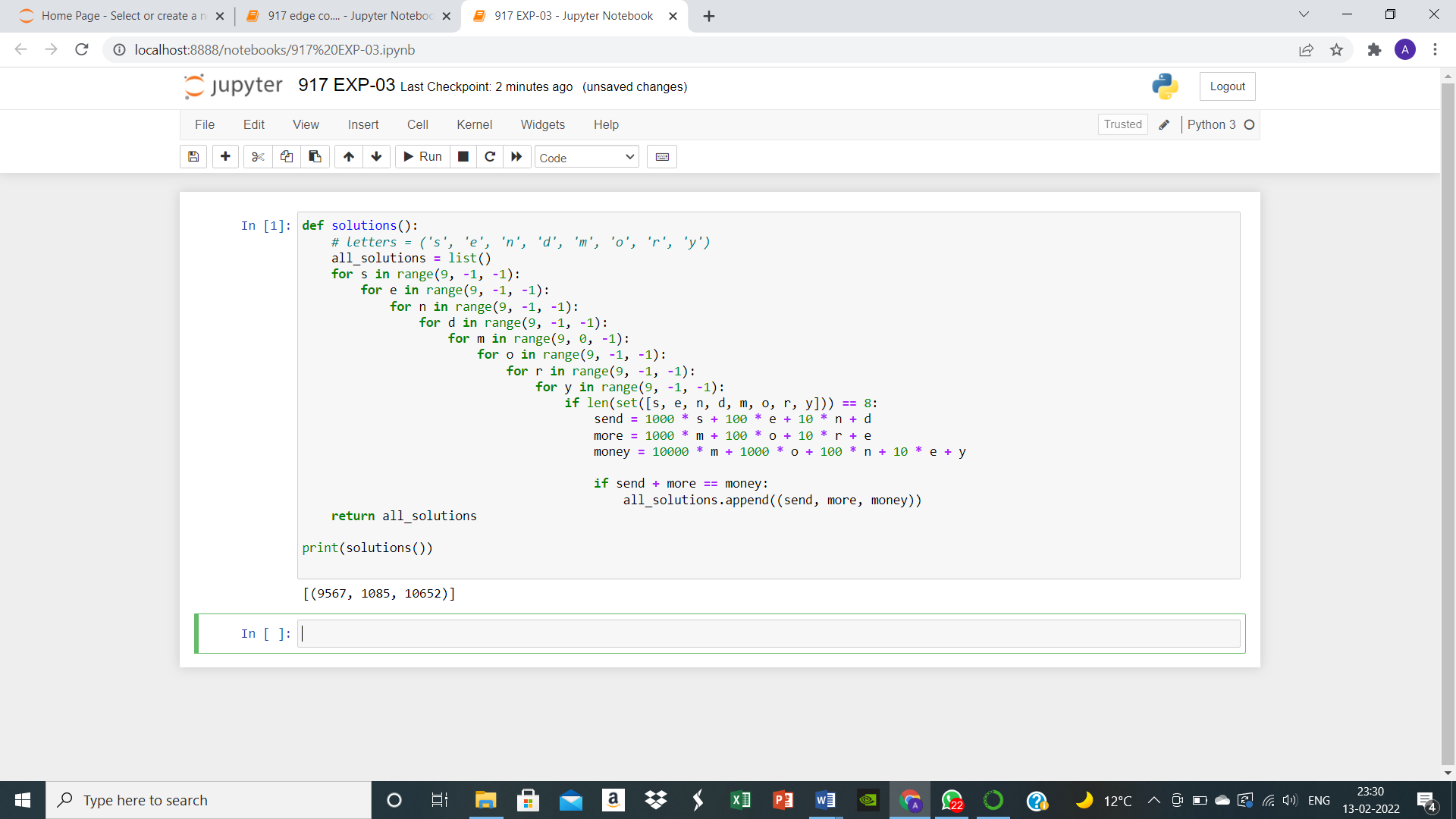
if send + more == money:

all\_solutions.append((send, more, money))

return all\_solutions

print(solutions())

OUTPUT



RESULT: The Constraint Satisfaction Problem was implemented successfully where the possible solutions were displayed based on user input.