

Manas Jha

RA191100310643

Date

### LAB-3

Aim: To implement the constraint Satisfaction Problem based on the given constraints.

SEND + MORE = MONEY

$$\begin{array}{r} 5 \quad 4 \quad 3 \quad 2 \quad 1 \\ \quad S \quad E \quad N \quad D \\ \quad M \quad O \quad R \quad E \\ \hline + \quad C_3 \quad C_2 \quad C_1 \\ \hline M \quad O \quad N \quad E \quad Y \end{array}$$

$$\begin{array}{r} 9 \quad 5 \quad 6 \quad 7 \\ + 1 \quad 0 \quad 8 \quad 5 \\ \hline 10 \quad 6 \quad 5 \quad 2 \end{array}$$

S	9
E	5
N	6
D	7

M	1
O	0
R	8
E	5

### Algorithm:

- From column 5,  $M=1$ , since it is only carry-over possible from sum of 2 digits number in column 4.
- To produce a carry from column 4 to column 5 ' $S+M$ ' is atleast 9 so ' $S=8$  or ' $9$ ' so ' $S+M=9$  or ' $10$ ' & so ' $O=0$  or ' $1$ '. But  $M=1$ , so ' $O=0$ '
- If there is a carry from column 3 to 4 then ' $E=9$ ' & ' $N=0$ '. But ' $O=0$ ' so there is no carry & ' $S=9$ ' & ' $C_3=0$ '.

- If there is no carry from column 2 to 3 then ' $E = N$ ' which is impossible, therefore there is carry & ' $N = E + 1$ ' & ' $c_2 = 1$ '
- If there is a carry from column 1 to 2 then ' $N + R = E \text{ mod } 10$ ' & ' $N = E + 1$ ' so ' $E + 1 + R = E \text{ mod } 10$ ' so ' $R = 9$ ' but ' $S = 9$ ', so there must be carry from column 1 to 2. Therefore ' $c_1 = 1$ ' & ' $R = 8$ '.
- To produce carry ' $c_1 = 1$ ' from column ~~1~~ 1 to 2, we must have ' $D + E = 10 + Y$ '
- 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.  
 $\therefore 'E = 5' \text{ \& } 'N = 6'$
- $D + E$  is at least 12 for that we get ' $D = 7$ ' & ' $Y = 2$ '.

Manas Jha

RA1911003010643

```
import itertools
```

```
def get_value(word, substitution):
```

```
    s = 0
```

```
    factor = 1
```

```
    for letter in reversed(word):
```

```
        s += factor * substitution[letter]
```

```
        factor *= 10
```

```
    return s
```

```
def solve2(equation):
```

```
    left, right = equation.lower().replace(' ', '').split('=')
```

```
    left = left.split('+')
```

```
    letters = set(right)
```

```
    for word in left:
```

```
        for letter in word:
```

```
            letters.add(letter)
```

```
    letters = list(letters)
```

```
    digits = range(10)
```

```
    for perm in itertools.permutations(digits, len(letters)):
```

```
        sol = dict(zip(letters, perm))
```

```
    if sum(get_value(word, sol) for word in left) == get_value(right, sol):
```

```
print(' + '.join(str(get_value(word, sol)) for word in left) + " = {} (mapping: {})"
.format(get_value(right, sol), sol))
```

```
print('EAT + THAT = APPLE ')
```

```
solve2('POINT + ZERO = ENERGY ')
```

```
bash - "ip-172-31-11-126" x Immediate x RA1911003010643/Al\ lab3.py x
Run Command: RA1911003010643/Al\ lab3.py
74285 + 6034 = 80319 (mapping: {'e': 0, 't': 5, 'r': 3, 'o': 4, 'i': 2, 'y': 9, 'z': 6, 'n': 8, 'p': 7, 'g': 1})
71385 + 9041 = 80426 (mapping: {'e': 0, 't': 5, 'r': 4, 'o': 1, 'i': 3, 'y': 6, 'z': 9, 'n': 8, 'p': 7, 'g': 2})
31745 + 9081 = 40826 (mapping: {'e': 0, 't': 5, 'r': 8, 'o': 1, 'i': 7, 'y': 6, 'z': 9, 'n': 4, 'p': 3, 'g': 2})
24735 + 6084 = 30819 (mapping: {'e': 0, 't': 5, 'r': 8, 'o': 4, 'i': 7, 'y': 9, 'z': 6, 'n': 3, 'p': 2, 'g': 1})
58369 + 2048 = 60417 (mapping: {'e': 0, 't': 9, 'r': 4, 'o': 8, 'i': 3, 'y': 7, 'z': 2, 'n': 6, 'p': 5, 'g': 1})
62479 + 8052 = 70531 (mapping: {'e': 0, 't': 9, 'r': 5, 'o': 2, 'i': 4, 'y': 1, 'z': 8, 'n': 7, 'p': 6, 'g': 3})
38549 + 2068 = 40617 (mapping: {'e': 0, 't': 9, 'r': 6, 'o': 8, 'i': 5, 'y': 7, 'z': 2, 'n': 4, 'p': 3, 'g': 1})
42659 + 8072 = 50731 (mapping: {'e': 0, 't': 9, 'r': 7, 'o': 2, 'i': 6, 'y': 1, 'z': 8, 'n': 5, 'p': 4, 'g': 3})
98504 + 3168 = 101672 (mapping: {'e': 1, 't': 4, 'r': 6, 'o': 8, 'i': 5, 'y': 2, 'z': 3, 'n': 0, 'p': 9, 'g': 7})
Process exited with code: 0
```