18CSC305J – AI LAB 3

Constraint Satisfaction Problem

Date: 13-01-2022

Name: Vinoth S

Reg No: RA1911030010103

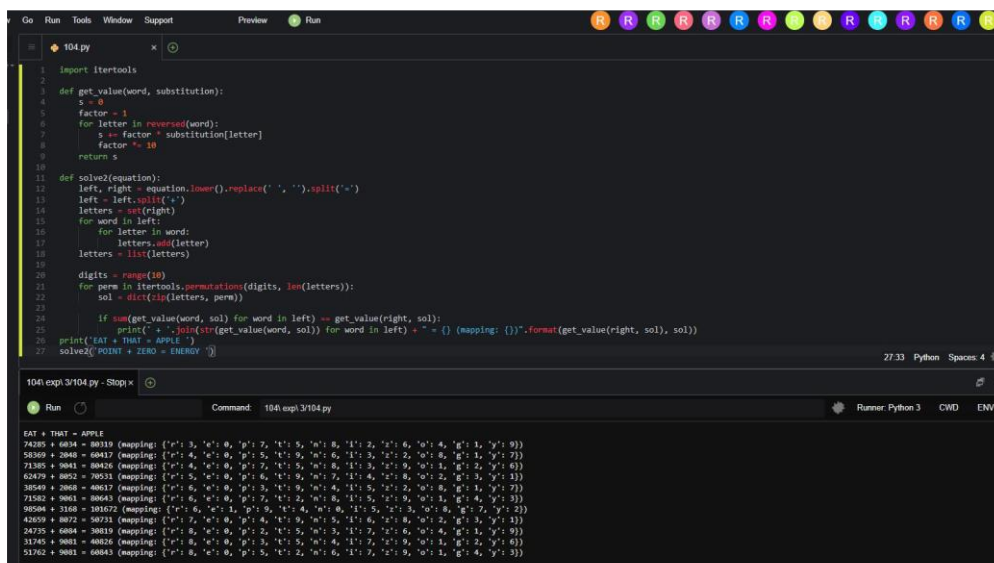**GitHub Link: https://github.com/vk1308**

Code:

```python
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))
```

```python
    return all_solutions


print(solutions())
```

Output:


```python
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 ')
```

```
EAT + THAT = APPLE
74285 + 6034 = 80319 (mapping: {'r': 3, 'e': 0, 'p': 7, 't': 5, 'n': 8, 'i': 2, 'z': 6, 'o': 4, 'g': 1, 'y': 9})
58369 + 2048 = 60417 (mapping: {'r': 4, 'e': 0, 'p': 5, 't': 9, 'n': 6, 'i': 3, 'z': 2, 'o': 8, 'g': 1, 'y': 7})
71385 + 9041 = 80426 (mapping: {'r': 4, 'e': 0, 'p': 7, 't': 5, 'n': 8, 'i': 3, 'z': 9, 'o': 1, 'g': 2, 'y': 6})
62479 + 8052 = 70531 (mapping: {'r': 5, 'e': 0, 'p': 6, 't': 9, 'n': 7, 'i': 4, 'z': 8, 'o': 2, 'g': 3, 'y': 1})
38549 + 2068 = 40617 (mapping: {'r': 6, 'e': 0, 'p': 3, 't': 9, 'n': 4, 'i': 5, 'z': 2, 'o': 8, 'g': 1, 'y': 7})
71582 + 9061 = 80643 (mapping: {'r': 6, 'e': 0, 'p': 7, 't': 2, 'n': 8, 'i': 5, 'z': 9, 'o': 1, 'g': 4, 'y': 3})
98504 + 3168 = 101672 (mapping: {'r': 6, 'e': 1, 'p': 9, 't': 4, 'n': 0, 'i': 5, 'z': 3, 'o': 8, 'g': 7, 'y': 2})
42659 + 8072 = 50731 (mapping: {'r': 7, 'e': 0, 'p': 4, 't': 9, 'n': 5, 'i': 6, 'z': 8, 'o': 2, 'g': 3, 'y': 1})
24735 + 6084 = 30819 (mapping: {'r': 8, 'e': 0, 'p': 2, 't': 5, 'n': 3, 'i': 7, 'z': 6, 'o': 4, 'g': 1, 'y': 9})
31745 + 9081 = 40826 (mapping: {'r': 8, 'e': 0, 'p': 3, 't': 5, 'n': 4, 'i': 7, 'z': 9, 'o': 1, 'g': 2, 'y': 6})
51762 + 9081 = 60843 (mapping: {'r': 8, 'e': 0, 'p': 5, 't': 2, 'n': 6, 'i': 7, 'z': 9, 'o': 1, 'g': 4, 'y': 3})
```