

LAB 3 – IMPLEMENTATION OD CSP

AI LAB

RAJAT KUMAR

RA1911003010652

ALGORITHM:

Step 1: Start

Step 2: Accept a expression 'SEND+MORE=MONEY'.

Step 3: Extract the words SEND,MORE and MONEY.

Step 4: Permute for different combination of values for S,E,N,D,M,O,R,Y.

Step5: And check if the sum of left value i.e, SEND+MORE is equal to right sum i.e, MONEY or not. If the sum value matches print the mapping.

Step 6: Continue for other permutations as well.

Step 7: Stop.

CODE:

```
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))
```

```
if __name__ == '__main__':
```

```
    solve2('SEND + MORE = MONEY')
```

OUTPUT:

```
bash - "ip-172-31-6-77" x RA1911003010652/lab3_ x +
Run Command: RA1911003010652/lab3_csp.py

9567 + 1085 = 10652 (mapping: {'o': 0, 'd': 7, 'y': 2, 'e': 5, 'r': 8, 'm': 1, 's': 9, 'n': 6})
2817 + 368 = 3185 (mapping: {'o': 3, 'd': 7, 'y': 5, 'e': 8, 'r': 6, 'm': 0, 's': 2, 'n': 1})
2819 + 368 = 3187 (mapping: {'o': 3, 'd': 9, 'y': 7, 'e': 8, 'r': 6, 'm': 0, 's': 2, 'n': 1})
3821 + 468 = 4289 (mapping: {'o': 4, 'd': 1, 'y': 9, 'e': 8, 'r': 6, 'm': 0, 's': 3, 'n': 2})
3712 + 467 = 4179 (mapping: {'o': 4, 'd': 2, 'y': 9, 'e': 7, 'r': 6, 'm': 0, 's': 3, 'n': 1})
3719 + 457 = 4176 (mapping: {'o': 4, 'd': 9, 'y': 6, 'e': 7, 'r': 5, 'm': 0, 's': 3, 'n': 1})
3829 + 458 = 4287 (mapping: {'o': 4, 'd': 9, 'y': 7, 'e': 8, 'r': 5, 'm': 0, 's': 3, 'n': 2})
5731 + 647 = 6378 (mapping: {'o': 6, 'd': 1, 'y': 8, 'e': 7, 'r': 4, 'm': 0, 's': 5, 'n': 3})
5732 + 647 = 6379 (mapping: {'o': 6, 'd': 2, 'y': 9, 'e': 7, 'r': 4, 'm': 0, 's': 5, 'n': 3})
5849 + 638 = 6487 (mapping: {'o': 6, 'd': 9, 'y': 7, 'e': 8, 'r': 3, 'm': 0, 's': 5, 'n': 4})
6851 + 738 = 7589 (mapping: {'o': 7, 'd': 1, 'y': 9, 'e': 8, 'r': 3, 'm': 0, 's': 6, 'n': 5})
6853 + 728 = 7581 (mapping: {'o': 7, 'd': 3, 'y': 1, 'e': 8, 'r': 2, 'm': 0, 's': 6, 'n': 5})
6524 + 735 = 7259 (mapping: {'o': 7, 'd': 4, 'y': 9, 'e': 5, 'r': 3, 'm': 0, 's': 6, 'n': 2})
6415 + 734 = 7149 (mapping: {'o': 7, 'd': 5, 'y': 9, 'e': 4, 'r': 3, 'm': 0, 's': 6, 'n': 1})
6419 + 724 = 7143 (mapping: {'o': 7, 'd': 9, 'y': 3, 'e': 4, 'r': 2, 'm': 0, 's': 6, 'n': 1})
7531 + 825 = 8356 (mapping: {'o': 8, 'd': 1, 'y': 6, 'e': 5, 'r': 2, 'm': 0, 's': 7, 'n': 3})
7643 + 826 = 8469 (mapping: {'o': 8, 'd': 3, 'y': 9, 'e': 6, 'r': 2, 'm': 0, 's': 7, 'n': 4})
7534 + 825 = 8359 (mapping: {'o': 8, 'd': 4, 'y': 9, 'e': 5, 'r': 2, 'm': 0, 's': 7, 'n': 3})
7316 + 823 = 8139 (mapping: {'o': 8, 'd': 6, 'y': 9, 'e': 3, 'r': 2, 'm': 0, 's': 7, 'n': 1})
7429 + 814 = 8243 (mapping: {'o': 8, 'd': 9, 'y': 3, 'e': 4, 'r': 1, 'm': 0, 's': 7, 'n': 2})
7539 + 815 = 8354 (mapping: {'o': 8, 'd': 9, 'y': 4, 'e': 5, 'r': 1, 'm': 0, 's': 7, 'n': 3})
7649 + 816 = 8465 (mapping: {'o': 8, 'd': 9, 'y': 5, 'e': 6, 'r': 1, 'm': 0, 's': 7, 'n': 4})
8432 + 914 = 9346 (mapping: {'o': 9, 'd': 2, 'y': 6, 'e': 4, 'r': 1, 'm': 0, 's': 8, 'n': 3})
8542 + 915 = 9457 (mapping: {'o': 9, 'd': 2, 'y': 7, 'e': 5, 'r': 1, 'm': 0, 's': 8, 'n': 4})
8324 + 913 = 9237 (mapping: {'o': 9, 'd': 4, 'y': 7, 'e': 3, 'r': 1, 'm': 0, 's': 8, 'n': 2})
```

RESULT: Hence, the implementation of CSP is done successfully.

AI LAB - 3

AIM :- Implementation of Constraint Satisfaction Problem - Cryptarithmic problem (SEND + MORE MONEY).

Rules

1. There should be a unique digit to be replaced with a unique alphabet.
2. The result should satisfy the predefined arithmetic rules i.e., $2+2=4$ nothing else.
3. Digits should be from 0-9 only.

Given a cryptarithmic problem
 $SEND + MORE = MONEY$

Let's assign $S \rightarrow 9$ $M \rightarrow 1$

$$\begin{array}{r} S \\ + M \\ \hline MO \end{array} \longrightarrow \begin{array}{r} 9 \\ + 1 \\ \hline 10 \end{array}$$

$$\begin{array}{r} E \\ + 0 \\ \hline N \end{array} \xrightarrow{\text{X}} \begin{array}{r} 5 \\ + 0 \\ \hline 5 \end{array}$$

We assume that C_2 (carry) = 1

$$\begin{array}{r} E \\ + 0 \\ \hline N \end{array} \longrightarrow \begin{array}{r} \textcircled{1} \\ 5 \\ + 0 \\ \hline 6 \end{array}$$

Further adding next two term N and R.

$$\begin{array}{r} N \\ + R \\ \hline E \end{array} \quad \xrightarrow{\text{X}} \quad \begin{array}{r} 6 \\ + 8 \\ \hline 14 \end{array}$$

consider $c_3 = 1$

$$\begin{array}{r} N \\ + R \\ \hline E \end{array} \quad \longrightarrow \quad \begin{array}{r} \textcircled{1} \\ 6 \\ + 8 \\ \hline 15 \end{array} \quad \text{we get } R = 8$$

On adding last two terms 1 carry must be produced

$$\begin{array}{r} D \\ + E \\ \hline Y \end{array} \quad \longrightarrow \quad \begin{array}{r} 7 \\ + 5 \\ \hline 12 \end{array}$$

keeping all constraints in mind the final state is :-

$$\begin{array}{r} c_3(0) \quad c_2(1) \quad c_1(1) \\ S(9) \quad E(5) \quad N(6) \quad D(7) \\ + M(1) \quad O(0) \quad R(8) \quad E(5) \\ \hline M(1) \quad O(0) \quad N(6) \quad E(5) \quad Y(2) \end{array}$$

S	9
E	5
N	6
D	7
M	1
O	0
R	8
Y	
Y	2