

Ex.No.3

Date:4/2/2021

Implementation of Constraint Satisfaction Problem

Aim: To Study and Implement Constraint Satisfaction Problem.

Methodology:

- There should be a unique digit to be replaced with a unique alphabet.
- The result should satisfy the predefined arithmetic rules, i.e., $2+2=4$, nothing else.
- Digits should be from 0-9 only.
- There should be only one carry forward, while performing the addition operation on a problem.
- The problem can be solved from both sides, i.e., left hand side (L.H.S), or right hand side (R.H.S)

Code:

```
from re import sub

def solve(q):
    try:
        n = (i for i in q if i.isalpha()).__next__()
    except StopIteration:
        return q if eval(sub(r'([0-9])0+([1-9]+)', r'\1\2', q)) else False
    else:
        for i in (str(i) for i in range(10) if str(i) not in q):
            res = solve(q.replace(n, str(i)))
            if res:
                return res
        return False
```

```

if __name__ == "__main__":
    query = str(input("Enter the String:"))
    r = solve(query)
    if r:
        print(r)
        for j in range(len(query)):
            print(query[j], "-->", r[j])
    else:
        print("Solution Not Found")

```

Output:

```

Enter the String:REASON==IT*IS+THERE
023456==71*74+18202
R --> 0
E --> 2
A --> 3
S --> 4
O --> 5
N --> 6
= --> =
= --> =
I --> 7
T --> 1
* --> *
I --> 7
S --> 4
+ --> +
T --> 1
H --> 8
E --> 2
R --> 0
E --> 2

```

Result: We have successfully studied and implemented Constraint Satisfaction Problem