**EXPERIMENT 12 - Crypt Arithmetic Problem**

**Aim:**

To implement basic search strategies – Crypt arithmetic.

**Theoretical Background:**

The Crypt-Arithmetic problem in [Artificial Intelligence](https://www.includehelp.com/ml-ai/introduction-to-artificial-intelligence.aspx" \t "https://www.includehelp.com/ml-ai/_blank) is a type of encryption problem in which the written message in an alphabetical form which is easily readable and understandable is converted into a numeric form which is neither easily readable nor understandable. The crypt-arithmetic problem deals with the converting of the message from the readable plain text to the non-readable ciphertext. The constraints which this problem follows during the conversion is as follows:

* A number 0-9 is assigned to a particular alphabet.
* Each different alphabet has a unique number.
* All the same, alphabets have the same numbers.
* The numbers should satisfy all the operations that any normal number does.

Let us take an example of the message: SEND + MORE = MONEY.

Here, to convert it into numeric form, we first split each word separately and represent it as follows:

S E N D

M O R E

-------------

M O N E Y

These alphabets then are replaced by numbers such that all the constraints are satisfied. So initially we have all blank spaces.

The solution to the given Crypt-Arithmetic problem is:

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

Which can be shown in layout form as:

9 5 6 7

1 0 8 5

-------------

1 0 6 5 2

**Algorithm:**

1. The code starts by declaring a variable called num.

2. It is initialized to 0 and then it starts looping through the word, assigning a value of 10 for each letter in the word.

3. The assignment statement assigns an integer value to every character in the string.

4. The code continues by checking if any of the first letters are zero or if any of the last letters are zero.

5. If either one is true, then that means there was no valid assignment and so this function returns false which will

cause our program to stop running at that point.

6. The code is written to find the value of a word in an assignment.

7. The code begins by declaring two variables, num and assigned.

8. The first variable, num, will store the number of times that letter appears in the word.

9. The second variable, assigned, stores the letters in the word with their corresponding values.

10. Next, a for loop is created which iterates through each character in the word.

11. In this loop we check if any letter has been set to zero or not assigned at all.

12. If either condition is true then it returns false and stops iterating through characters within that iteration of the loop.

13. Otherwise it increments num by 10 and assigns its value to assigned [char].

14. This process repeats until all characters have been checked for

15. The code is a function that takes in three arguments: word1, word2, and result.

16. The function is called solve and it returns the solutions to the equation of word1 + word2 = result.

17. The first line of code assigns values to letters so that they are sorted alphabetically by their value.

18. This is done with set () because each letter has two possible values (A-Z or a-z).

19. The next line creates an empty list named assigned which will hold all the possible assignments for words 1 and 2.

20. Next, we iterate through 10 numbers from 0-9 using range ().

21. We then check if any number not in assigned is found by checking if it's not in the list of letters (assigned) .

22. If there isn't one yet, we create a new letter at index cur letter with pop (), assign num to it with assignment

operator (=), then call \_solve (word1, word2, result) on our solution list created earlier

23. Then we remove cur letter from assigned so that only A-Z remain as valid characters for assigning values to words 1 and 2 respectively.

24. The code is the solution to a word puzzle.

25. It will take in two words and assign them to a result.

26. The assignment can be done by finding values in the words, or by assigning letters to numbers.

27. The first line of code assigns the value of 1 to letter "a" and 2 to letter "b".

28. This is done with find value ().

29. The next line assigns 3 to letter "c", 4 to letter "d", 5 to letter "e", 6 to letter

30. The code is trying to find the number of solutions.

31. The code starts by checking if len (result) > max (len (word1), len (word2)) + 1 or len (letters) > 10, which means that it will stop searching when either one of those conditions is met.11

32. Then, the code creates a list called solutions and iterates through each solution in order to print out its contents.

33. The first line says "CRYPTARITHMETIC PUZZLE SOLVER".

34. This is just an example of what you might see on your screen after running this program.

35. The code will print 0 Solutions!

36. If the input is not a word or number.

37. The code above will then iterate through the list of solutions and print them out.

**Conclusion:**

Thus the program to implement crypt arithmetic search strategy is implemented and executed successfully.