# **Assignment 4 CryptArithmetic Problem**

Name: Upendra kadre Div: C

Roll No. 69 GR No. 1710505

Implementation of crypt arithmetic problem.

SEND

+

MORE

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

MONEY

Ans.

#include <bits/stdc++.h>

using namespace std;

vector<int> taken(10);

struct node

{

char c;

int v;

};

int check(node\* array, const int count, string s1,

string s2, string s3)

{

int n1 = 0, n2 = 0, n3 = 0, m = 1, j, i;

for (i = s1.length() - 1; i >= 0; i--)

{

char ch = s1[i];

for (j = 0; j < count; j++)

if (array[j].c == ch)

break;

n1 += m \* array[j].v;

m \*= 10;

}

m = 1;

for (i = s2.length() - 1; i >= 0; i--)

{

char ch = s2[i];

for (j = 0; j < count; j++)

if (array[j].c == ch)

break;

n2 += m \* array[j].v;

m \*= 10;

}

m = 1;

for (i = s3.length() - 1; i >= 0; i--)

{

char ch = s3[i];

for (j = 0; j < count; j++)

if (array[j].c == ch)

break;

n3 += m \* array[j].v;

m \*= 10;

}

if (n3 == (n1 + n2))

return 1;

return 0;

}

bool combinations(const int count, node\* array, int n,

string s1, string s2, string s3)

{

if (n == count - 1)

{

for (int i = 0; i < 10; i++)

{

if (taken[i] == 0)

{

array[n].v = i;

if (check(array, count, s1, s2, s3) == 1)

{

cout << "\nSolution found: ";

for (int j = 0; j < count; j++)

cout << " " << array[j].c << " = "

<< array[j].v;

return true;

}

}

}

return false;

}

for (int i = 0; i < 10; i++)

{

if (taken[i] == 0)

{

array[n].v = i;

taken[i] = 1;

if (combinations(count, array, n + 1, s1, s2, s3))

return true;

taken[i] = 0;

}

}

return false;

}

bool CryptoArth(string s1, string s2, string s3)

{

int count = 0;

int l1 = s1.length();

int l2 = s2.length();

int l3 = s3.length();

vector<int> freq(26);

for (int i = 0; i < l1; i++)

++freq[s1[i] - 'A'];

for (int i = 0; i < l2; i++)

++freq[s2[i] - 'A'];

for (int i = 0; i < l3; i++)

++freq[s3[i] - 'A'];

for (int i = 0; i < 26; i++)

if (freq[i] > 0)

count++;

if (count > 10)

{

cout << "Invalid strings";

return 0;

}

node array[count];

for (int i = 0, j = 0; i < 26; i++)

{

if (freq[i] > 0)

{

array[j].c = char(i + 'A');

j++;

}

}

return combinations(count, array, 0, s1, s2, s3);

}

int main()

{

string s1 = "SEND";

string s2 = "MORE";

string s3 = "MONEY";

if (CryptoArth(s1, s2, s3) == false)

cout << "No solution";

return 0;

}

**Output:**

Solution found:

D = 1 E = 5 M = 0 N = 3 O = 8 R = 2 S = 7 Y = 6