

LAPORAN TUGAS KECIL 1 IF2211 STRATEGI ALGORITMA
Semester II tahun 2020/2021

Penyelesaian Cryptarithmic dengan Algoritma Brute Force

Faris Hasim Syauqi
13519050

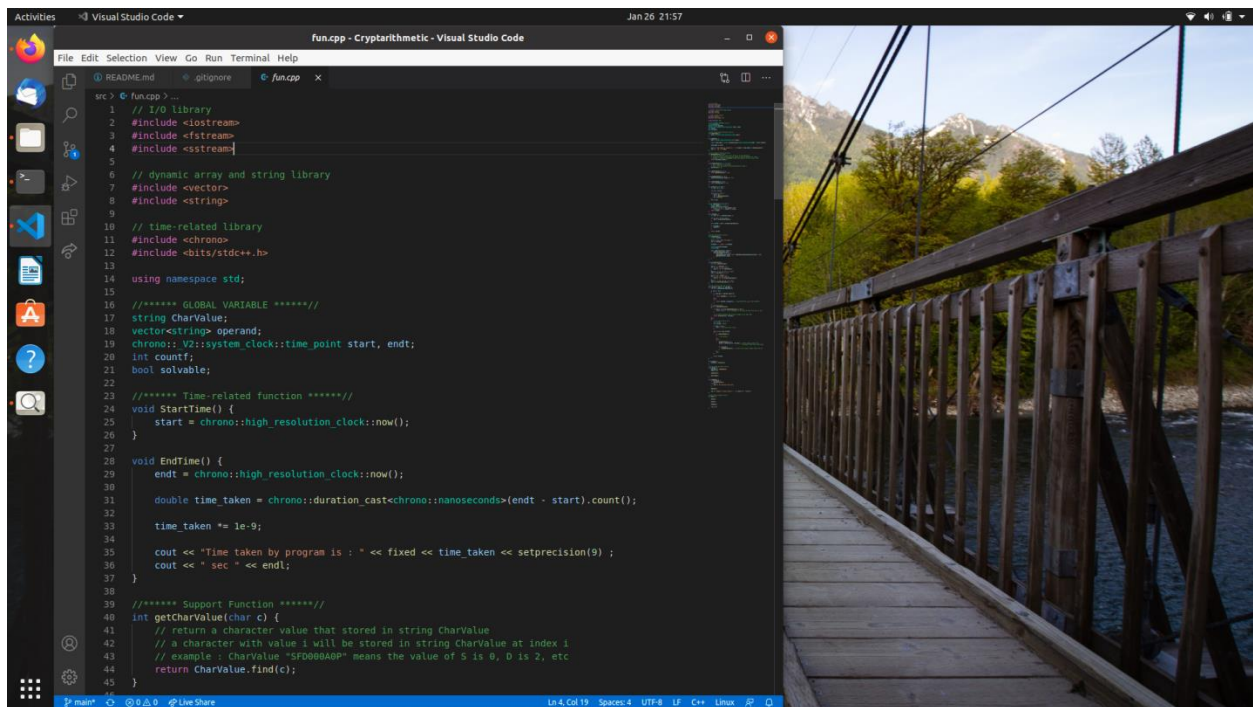
A. Algoritma

Berikut adalah langkah-langkah yang dilakukan dalam penyelesaian persoalan cryptarithmic dengan menggunakan Algoritma Brute Force:

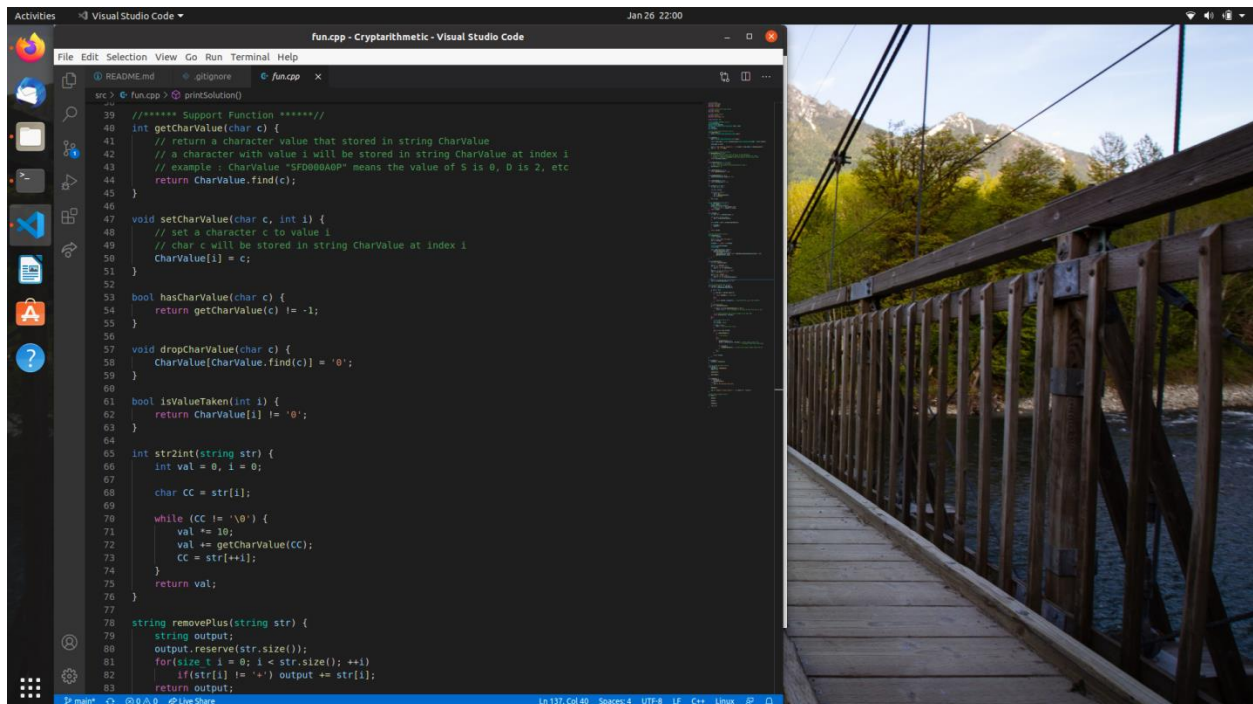
1. Persoalan cryptarithmic akan diselesaikan secara rekursif dimulai dengan beri nilai 1 untuk karakter pertama pada operand pertama. Kemudian geser ke karakter selanjutnya.
2. Basis : terjadi ketika semua huruf pada semua operand (termasuk dalam hal ini kata yang hasil penjumlahan) sudah di-assign dengan suatu nilai, yaitu ketika karakter terakhir pada kata terakhir sudah di-assign dengan suatu nilai, maka akan dilakukan pemeriksaan penjumlahan, setiap huruf akan diganti dengan angka yang di-assign.
Jika Pemeriksaan penjumlahan bernilai benar, maka solusi ditemukan, jika tidak maka mundur satu langkah ke karakter sebelumnya.
3. Rekurens : terjadi selama belum mencapai karakter terakhir pada kata terakhir. Pada kasus ini ada tiga kemungkinan
 - 3.a) Jika semua karakter pada kata/operand saat ini sudah di-assign dengan nilai, maka lanjut ke karakter pertama pada kata/operand selanjutnya.
 - 3.b) Jika karakter saat ini sudah memiliki nilai (karena huruf yang sama sudah muncul sebelumnya), periksa apakah karakter saat ini merupakan awal kata dan apakah nilai karakternya adalah 0. Jika iya, karena karakter pertama tidak boleh bernilai nol maka mundur satu langkah ke karakter sebelumnya. Selain kasus itu maka maju ke karakter selanjutnya.
 - 3.c) Jika karakter saat ini belum memiliki nilai, maka beri nilai terkecil yang tersedia untuk huruf tersebut, tersedia dalam artian rentang [0,9] yang belum diambil oleh huruf lain. Kemudian lanjut ke karakter selanjutnya. Jika berhasil maka solusi ditemukan, jika tidak maka karakter ini di-assign dengan nilai yang lebih besar terkecil yang tersedia, kemudian lanjut ke karakter selanjutnya. Jika tidak ada satupun nilai yang berujung dengan solusi, maka mundur satu langkah ke karakter sebelumnya.

B. Source Code Program

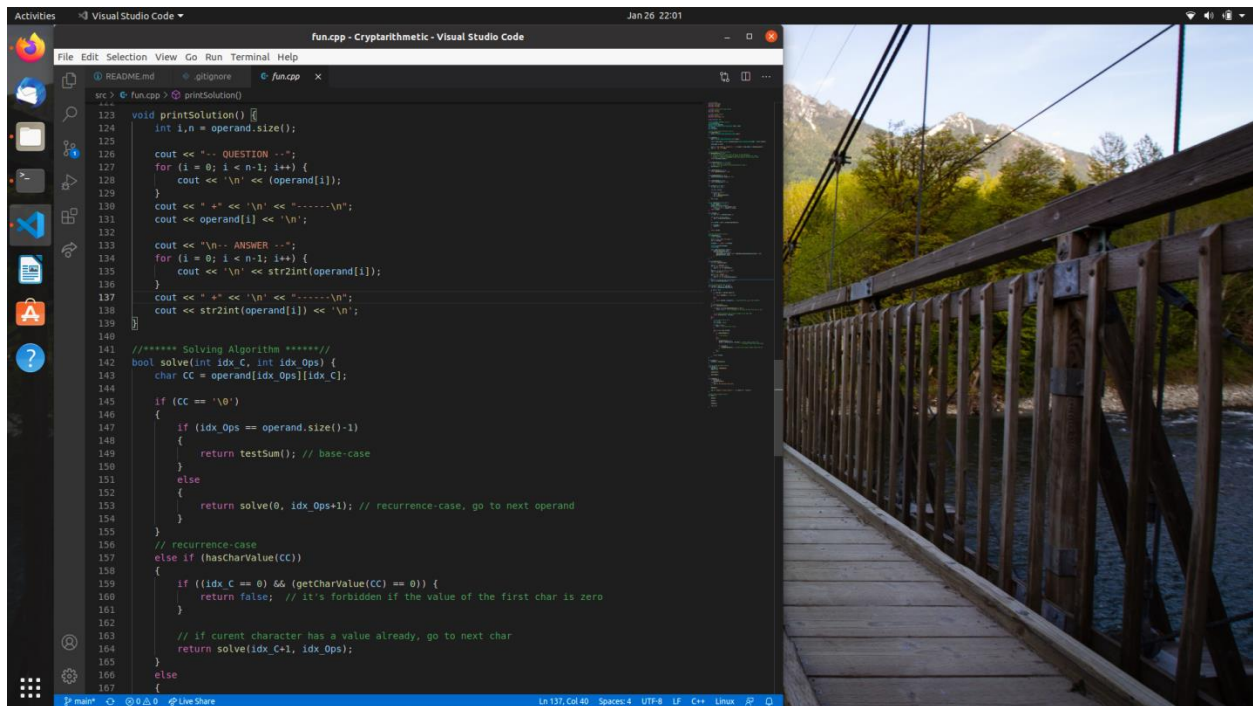
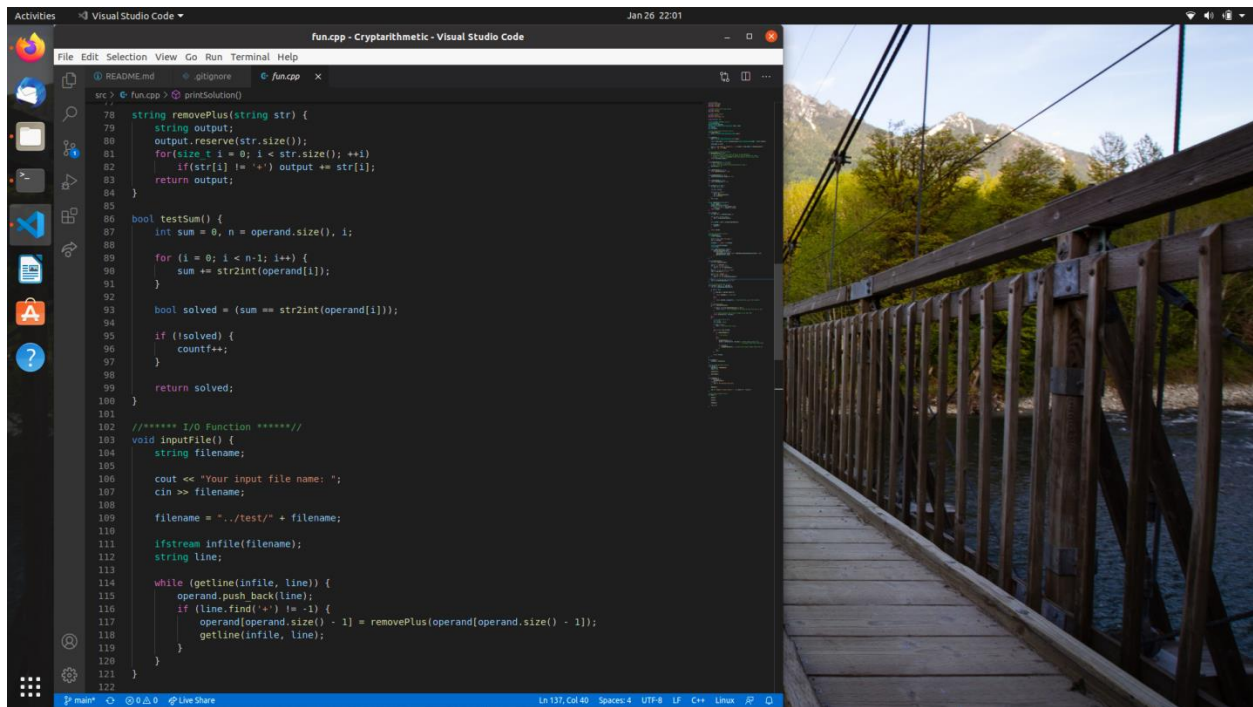
Berikut adalah source-code program sebagai implementasi dari algoritma di atas.

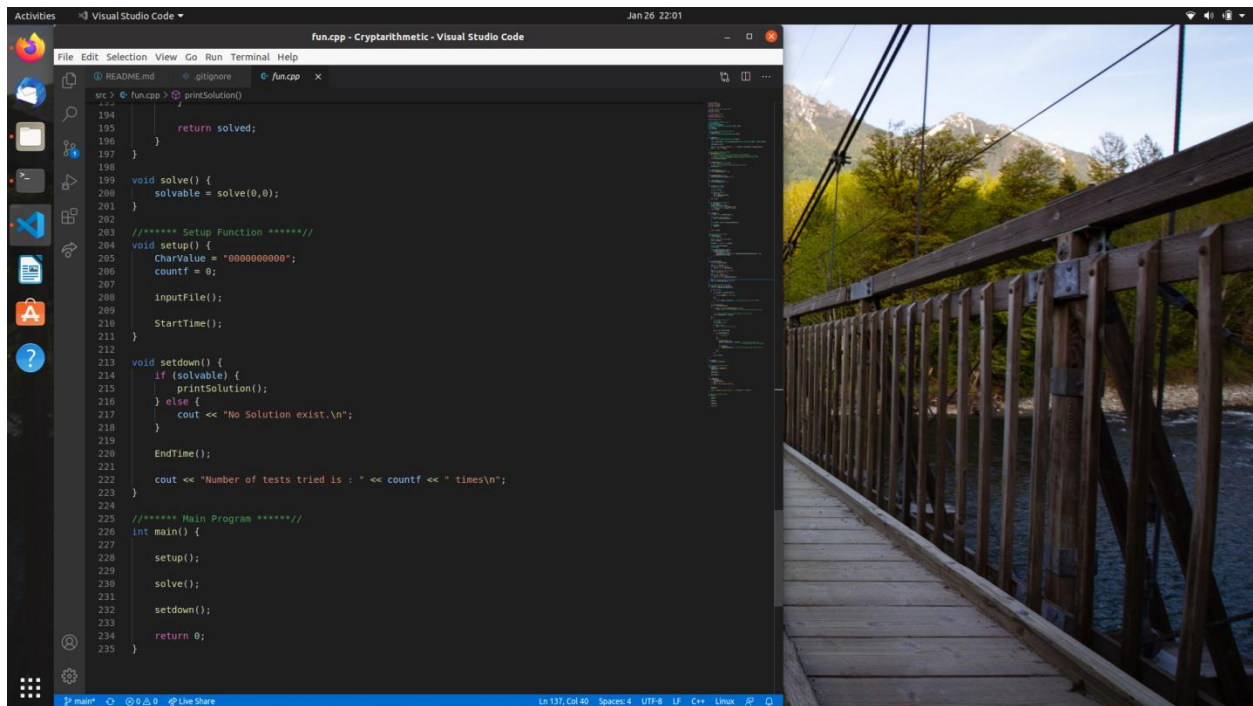
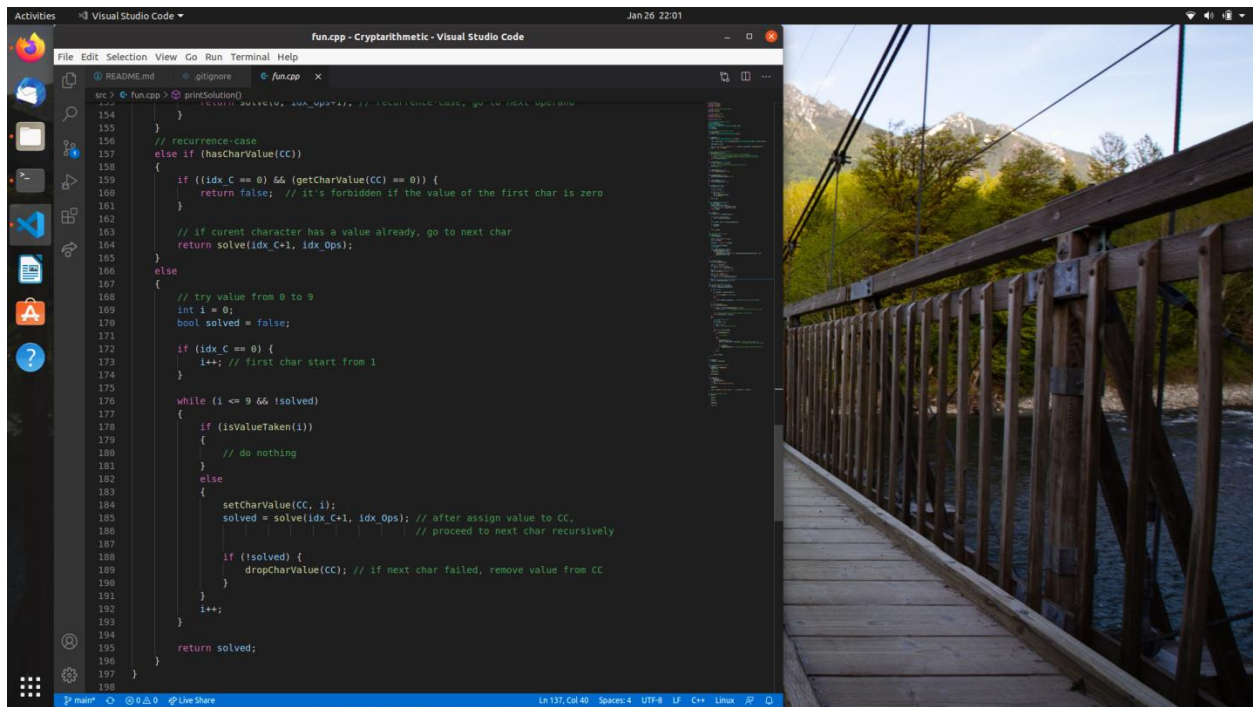


```
1 // I/O library
2 #include <iostream>
3 #include <fstream>
4 #include <sstream>
5
6 // dynamic array and string library
7 #include <vector>
8 #include <string>
9
10 // time-related library
11 #include <chrono>
12 #include <bits/stdc++.h>
13
14 using namespace std;
15
16 //***** GLOBAL VARIABLE *****/
17 string CharValue;
18 vector<string> operand;
19 chrono::V2::system_clock::time_point start, endt;
20 int countf;
21 bool solvable;
22
23 //***** Time-related function *****/
24 void StartTime() {
25     start = chrono::high_resolution_clock::now();
26 }
27
28 void EndTime() {
29     endt = chrono::high_resolution_clock::now();
30
31     double time_taken = chrono::duration_cast<chrono::nanoseconds>(endt - start).count();
32
33     time_taken *= 1e-9;
34
35     cout << "Time taken by program is : " << fixed << time_taken << setprecision(9) ;
36     cout << " sec " << endl;
37 }
38
39 //***** Support Function *****/
40 int getCharValue(char c) {
41     // return a character value that stored in string CharValue
42     // a character with value 1 will be stored in string CharValue at index 1
43     // example : CharValue "SF0000ARP" means the value of S is 0, 0 is 2, etc
44     return CharValue.find(c);
45 }
```



```
39 //***** Support Function *****/
40 int getCharValue(char c) {
41     // return a character value that stored in string CharValue
42     // a character with value 1 will be stored in string CharValue at index 1
43     // example : CharValue "SF0000ARP" means the value of S is 0, 0 is 2, etc
44     return CharValue.find(c);
45 }
46
47 void setCharValue(char c, int i) {
48     // set a character c to value i
49     // char c will be stored in string CharValue at index i
50     CharValue[i] = c;
51 }
52
53 bool hasCharValue(char c) {
54     return getCharValue(c) != -1;
55 }
56
57 void dropCharValue(char c) {
58     CharValue[CharValue.find(c)] = '0';
59 }
60
61 bool isValueTaken(int i) {
62     return CharValue[i] != '0';
63 }
64
65 int str2int(string str) {
66     int val = 0, i = 0;
67
68     char CC = str[i];
69
70     while (CC != '\0') {
71         val *= 10;
72         val += getCharValue(CC);
73         CC = str[++i];
74     }
75     return val;
76 }
77
78 string removePlus(string str) {
79     string output;
80     output.reserve(str.size());
81     for(size_t i = 0; i < str.size(); ++i)
82         if(str[i] != '+') output += str[i];
83     return output;
84 }
```

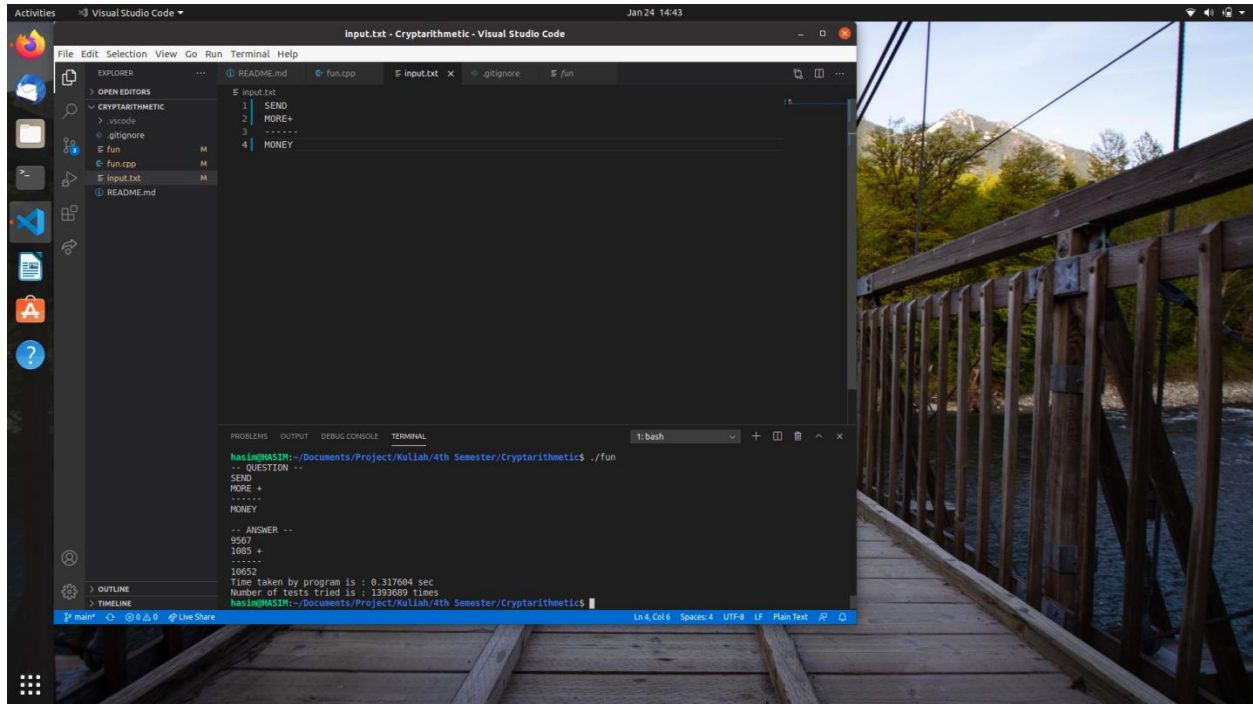




C. Screenshot Percobaan

Berikut adalah beberapa foto hasil percobaan dari program.

1. SEND + MORE = MONEY

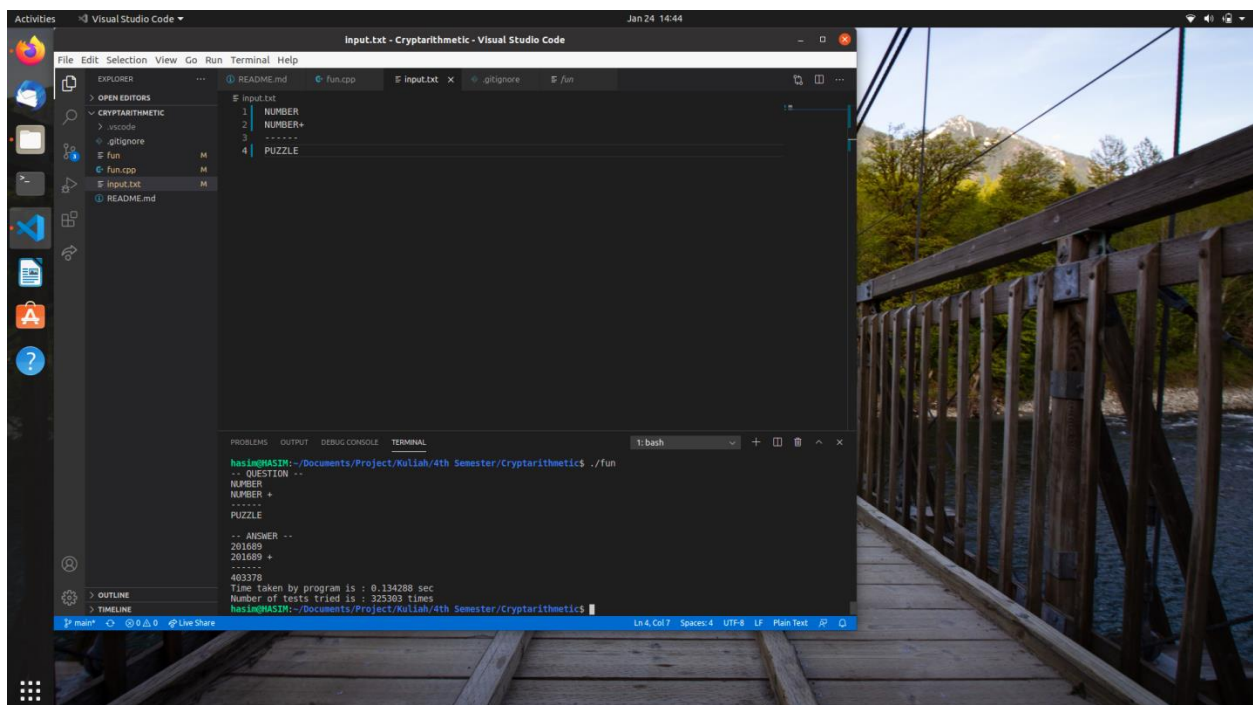


The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'CRYPTARITHMETIC' with files 'input.txt', 'fun.cpp', and 'README.md'. The terminal displays the output of the program for the puzzle 'SEND + MORE = MONEY'.

```
haslinghasim@~/Documents/Project/Kuliah/4th Semester/Cryptarithmetics: ./fun
-- QUESTION --
SEND
MORE +
-----
MONEY

-- ANSWER --
9567
1085 +
-----
10652
Time taken by program is : 0.317664 sec
Number of tests tried is : 1393689 times
haslinghasim@~/Documents/Project/Kuliah/4th Semester/Cryptarithmetics:
```

2. NUMBER + NUMBER = PUZZLE

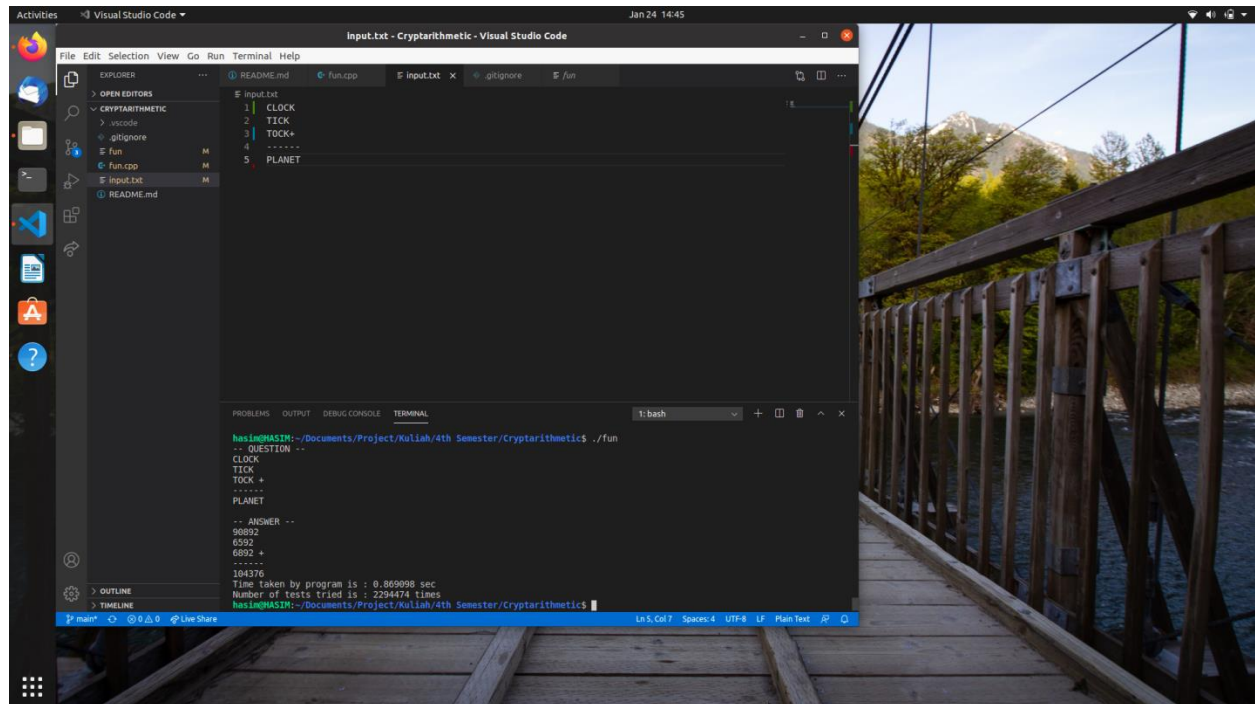


The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'CRYPTARITHMETIC' with files 'input.txt', 'fun.cpp', and 'README.md'. The terminal displays the output of the program for the puzzle 'NUMBER + NUMBER = PUZZLE'.

```
haslinghasim@~/Documents/Project/Kuliah/4th Semester/Cryptarithmetics: ./fun
-- QUESTION --
NUMBER
NUMBER +
-----
PUZZLE

-- ANSWER --
201689
201689 +
-----
403378
Time taken by program is : 0.134288 sec
Number of tests tried is : 325303 times
haslinghasim@~/Documents/Project/Kuliah/4th Semester/Cryptarithmetics:
```

3. CLOCK + TICK + TOCK = PLANET



```
Input.txt - Cryptarithmic - Visual Studio Code
Jan 24 14:45

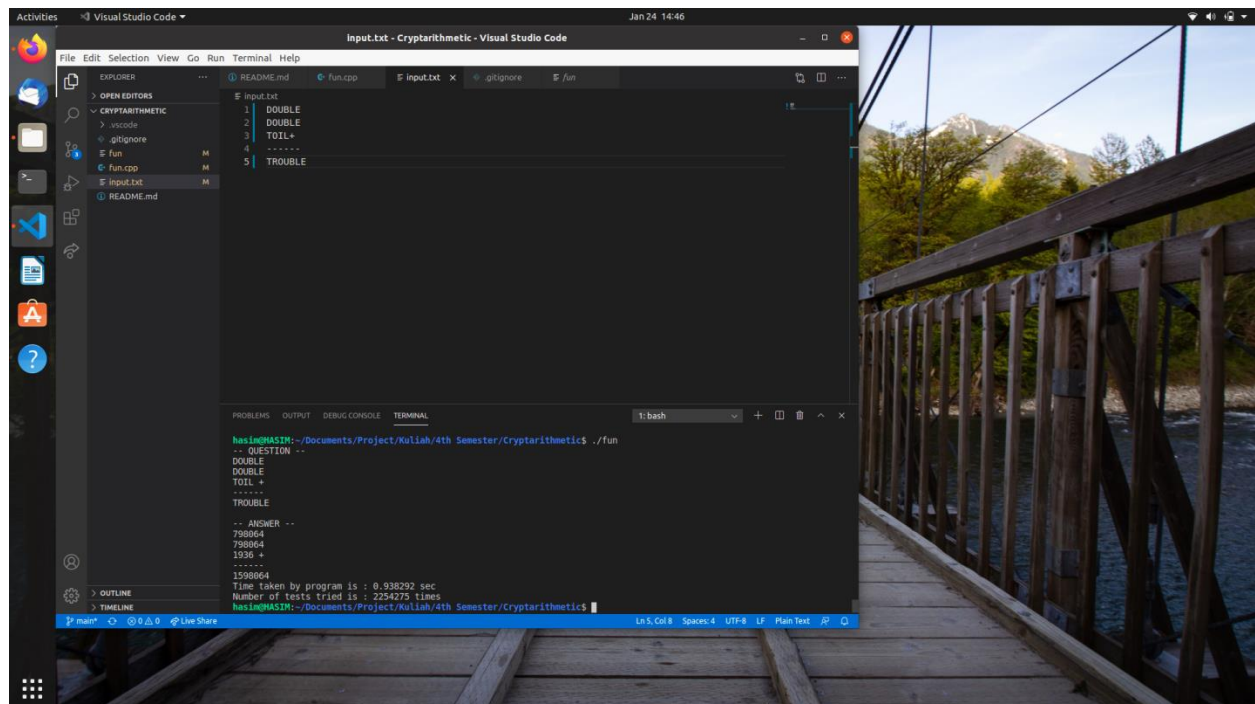
EXPLORER
- CRYPTARITHMETIC
  - README.md
  - fun.cpp
  - input.txt
  - README.md

1 | CLOCK
2 | TICK
3 | TOCK+
4 | -----
5 | PLANET

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: bash
hasin@HASIM:~/Documents/Project/KuLiab/4th Semester/Cryptarithmetics$ ./fun
-- QUESTION --
CLOCK
TICK
TOCK +
-----
PLANET

-- ANSWER --
90802
6592
6692 +
-----
104376
Time taken by program is : 0.869098 sec
Number of tests tried is : 2294474 times
hasin@HASIM:~/Documents/Project/KuLiab/4th Semester/Cryptarithmetics$
```

4. DOUBLE + DOUBLE + TOIL = TROUBLE



```
Input.txt - Cryptarithmic - Visual Studio Code
Jan 24 14:46

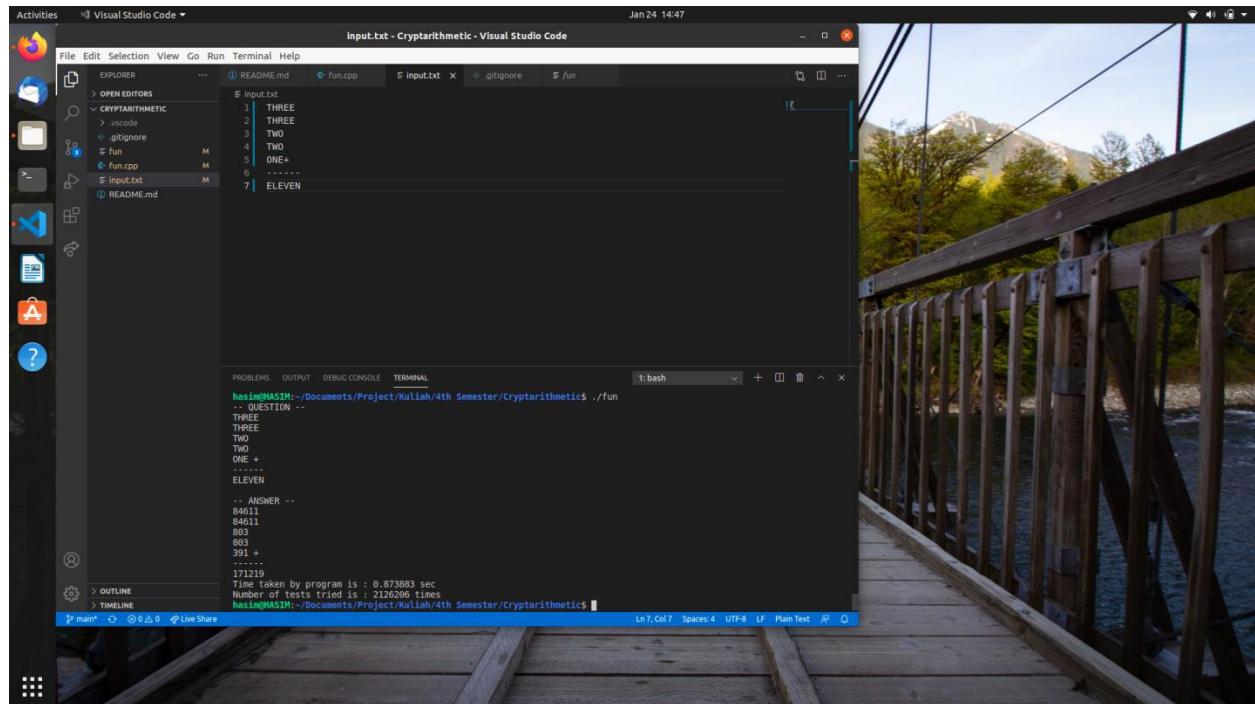
EXPLORER
- CRYPTARITHMETIC
  - README.md
  - fun.cpp
  - input.txt
  - README.md

1 | DOUBLE
2 | DOUBLE
3 | TOIL+
4 | -----
5 | TROUBLE

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: bash
hasin@HASIM:~/Documents/Project/KuLiab/4th Semester/Cryptarithmetics$ ./fun
-- QUESTION --
DOUBLE
DOUBLE
TOIL +
-----
TROUBLE

-- ANSWER --
798064
798064
1936 +
-----
1598064
Time taken by program is : 0.938292 sec
Number of tests tried is : 2254275 times
hasin@HASIM:~/Documents/Project/KuLiab/4th Semester/Cryptarithmetics$
```

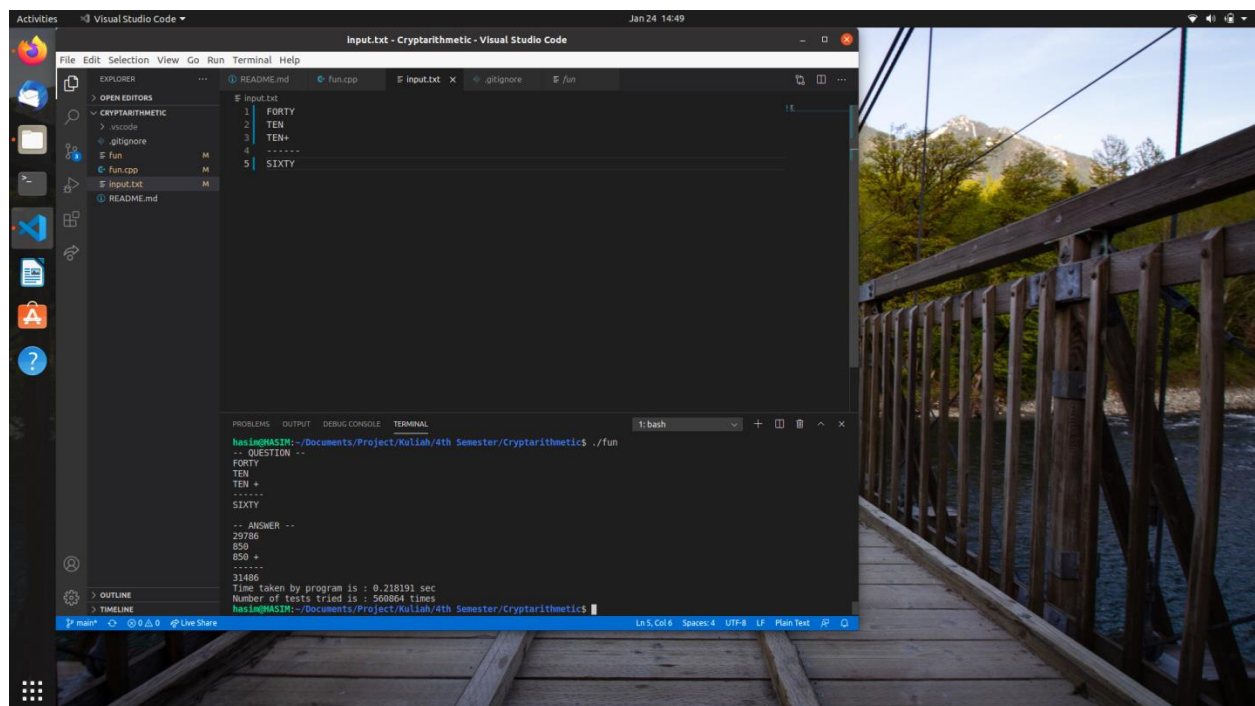

5. THREE + THREE + TWO + TWO + ONE = ELEVEN



```
Input.txt - Cryptarithmic - Visual Studio Code
1 THREE
2 THREE
3 TWO
4 TWO
5 ONE+
6 -----
7 ELEVEN

hasInMASIM:~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics ./fun
-- QUESTION --
THREE
THREE
TWO
TWO
ONE +
-----
ELEVEN
-- ANSWER --
84611
84611
803
803
391 +
-----
171219
Time taken by program is : 0.873883 sec
Number of tests tried is : 2126206 times
hasInMASIM:~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics
```

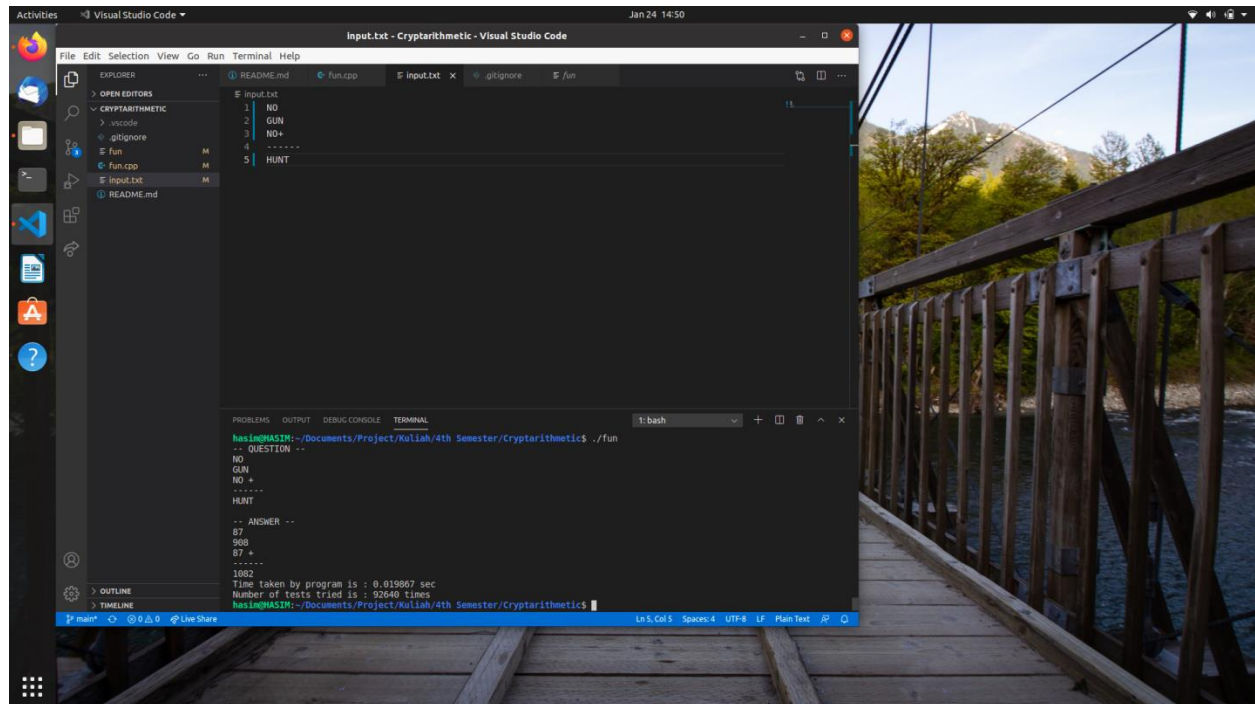
6. FORTY + TEN + TEN = SIXTY



```
Input.txt - Cryptarithmic - Visual Studio Code
1 FORTY
2 TEN
3 TEN+
4 -----
5 SIXTY

hasInMASIM:~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics ./fun
-- QUESTION --
FORTY
TEN
TEN +
-----
SIXTY
-- ANSWER --
29786
850
850 +
-----
31486
Time taken by program is : 0.218191 sec
Number of tests tried is : 560864 times
hasInMASIM:~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics
```

7. NO + GUN + NO = HUNT

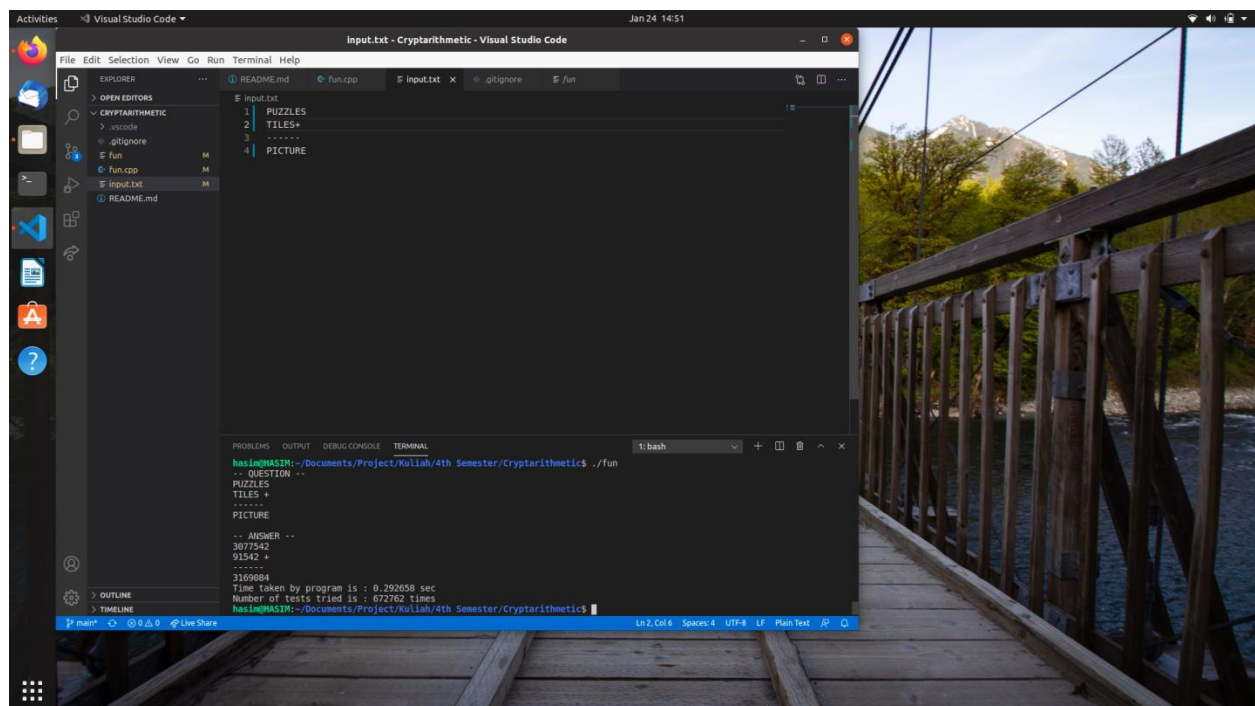


The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'CRYPTARITHMETIC' with files 'input.txt', 'fun.cpp', and 'README.md'. The terminal displays the output of a program run in the directory 'hasInMASIM~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics'.

```
hasInMASIM~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics $ ./fun
-- QUESTION --
NO
GUN
NO +
-----
HUNT

-- ANSWER --
87
908
87 +
-----
1082
Time taken by program is : 0.019867 sec
Number of tests tried is : 92640 times
hasInMASIM~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics
```

8. PUZZLES + TILES = PICTURE

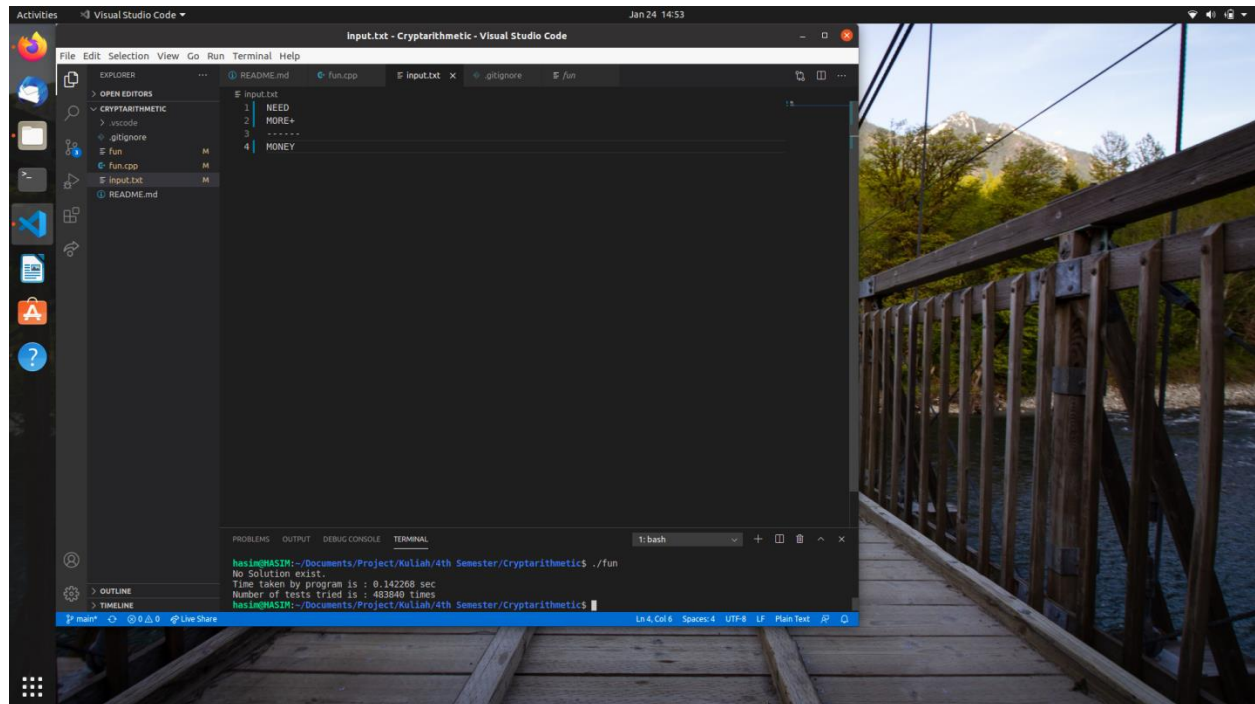


The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'CRYPTARITHMETIC' with files 'input.txt', 'fun.cpp', and 'README.md'. The terminal displays the output of a program run in the directory 'hasInMASIM~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics'.

```
hasInMASIM~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics $ ./fun
-- QUESTION --
PUZZLES
TILES +
-----
PICTURE

-- ANSWER --
3077542
93542 +
-----
3169084
Time taken by program is : 0.292658 sec
Number of tests tried is : 672762 times
hasInMASIM~/Documents/Project/KuLiah/4th Semester/Cryptarithmetics
```


9. NEED + MORE = MONEY



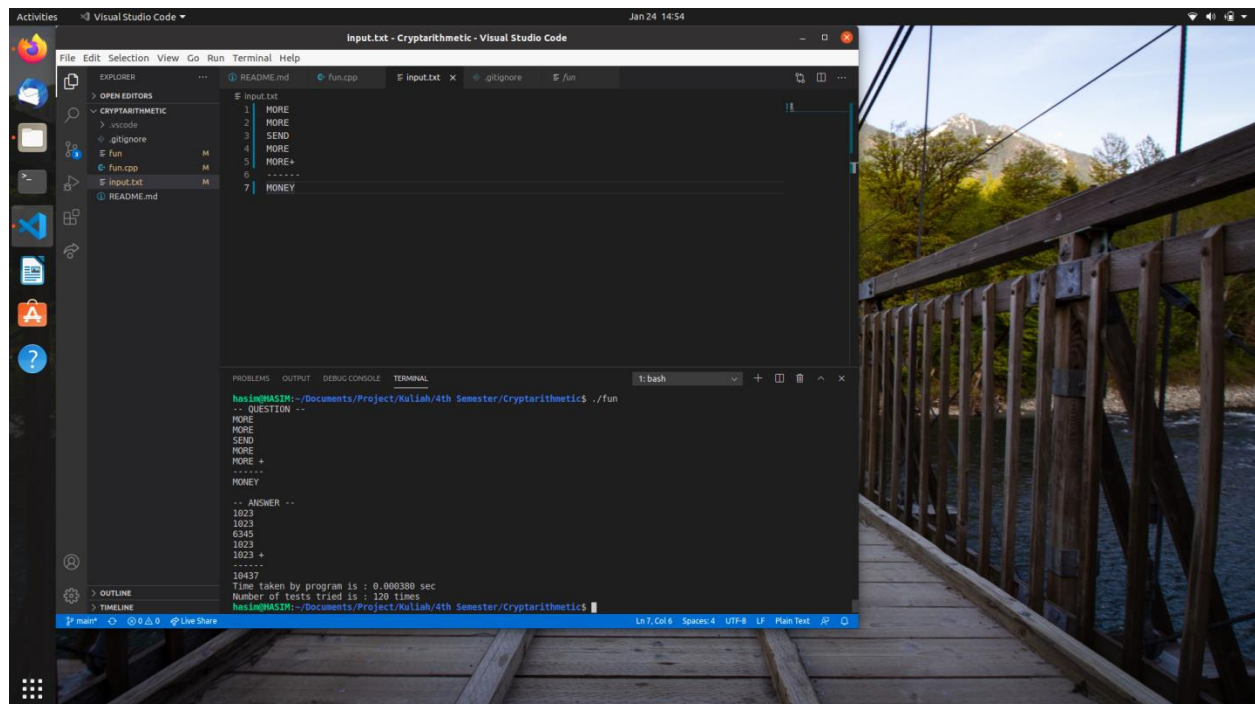
The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The main editor displays a file named `input.txt` containing the following text:

```
1 NEED
2 MORE+
3 -----
4 MONEY
```

The terminal window shows the output of the `./fun` command:

```
hasInMASIM:~/Documents/Project/Kuliah/4th Semester/Cryptarithmetics$ ./fun
No Solution exist.
Time taken by program is : 0.142268 sec
Number of tests tried is : 483840 times
```

10. MORE + MORE + SEND + MORE + MORE = MONEY



The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The main editor displays a file named `input.txt` containing the following text:

```
1 MORE
2 MORE
3 SEND
4 MORE
5 MORE+
6 -----
7 MONEY
```

The terminal window shows the output of the `./fun` command:

```
hasInMASIM:~/Documents/Project/Kuliah/4th Semester/Cryptarithmetics$ ./fun
-- QUESTION --
MORE
MORE
SEND
MORE
MORE +
-----
MONEY

-- ANSWER --
1023
1023
6345
1023
1023 +
-----
10437
Time taken by program is : 0.000380 sec
Number of tests tried is : 120 times
```

D. Alamat Kode Program

Program dapat diunduh dari alamat berikut :

Drive : <https://drive.google.com/drive/u/0/folders/1wb5hmqUxUIWa5SahqD7oxCDjFZB-GyF3>

(Pilih Tucil1_13519050.zip) atau dapat diunduh dari github pada tautan berikut :

GitHub : <https://github.com/farishasim/Cryptarithmic>

E. Tabel Penilaian

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa kesalahan (no syntax error)	√	
2. Program berhasil running	√	
3. Program dapat membaca file masukan dan menuliskan luaran	√	
4. Solusi cryptarithmic hanya benar untuk persoalan cryptarihtmetic dengan dua buah operand.		√
5. Solusi cryptarithmic benar untuk persoalan cryptarihtmetic untuk lebih dari dua buah operand.	√	