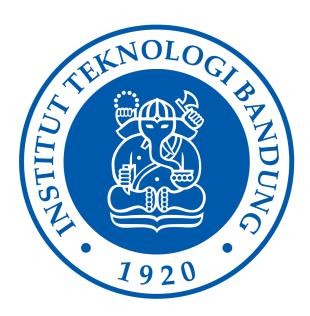
LAPORAN TUGAS KECIL

ALGORITMA BRUTE FORCE UNTUK MENYELESAIKAN TEKA-TEKI CRYPTARITHMETICS

IF2211 STRATEGI ALGORITMA



Disusun Oleh: James Chandra 13519078

Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung 2021

1. ALGORITMA BRUTE FORCE

Cryptarithmetics, juga dikenal sebagai *verbal arithmetics*, tergolong sebagai sebuah persoalan *Constraint Satisfaction* (CSP) yang melibatkan suatu persamaan dengan angka-angka tak diketahui yang direpresentasikan dengan huruf-huruf. Salah satu contoh persoalan *cryptarithmetics* adalah sebagai berikut.

СР	23
IS	74
FUN+	968+
TRUE	1065

Ilustrasi 1. Contoh Persoalan dan Penyelesaian Cryptarithmethic

Pada tugas kecil IF2211 mata kuliah Strategi Algoritma ini, akan digunakan sebuah algoritma *brute force* untuk mencari penyelesaian permasalahan *cryptarithmetic*, dengan langkah-langkah sebagai berikut.

- 1. Inisialisasi variabel dan array untuk penyimpan data operan, hasil, dan karakter unik.
- 2. Baca file kemudian masukan baris-baris operan sebagai elemen array data operan, kemudian masukan baris hasil dalam variabel hasil, dan catat pula karakter unik yang ditemui dalam suatu array.
- 3. Inisialisasi variabel indeks iterasi menjadi 10ⁿ-1 dengan n adalah jumlah elemen huruf unik, kemudian inisialisasi pula variabel penghitung jumlah solusi yang ditemukan menjadi 0.
- 4. Masuk ke loop *while* (dengan kondisi variabel iterasi masih lebih besar/sama dengan 10ⁿ⁻¹, dengan keterangan n adalah jumlah elemen huruf unik), setiap iterasi variabel indeks iterasi dikurangi 1.
- 5. Apabila panjang dari sekuens angka tidak memiliki digit duplikat dan merupakan sama panjangnya dengan banyak huruf unik, maka
- 6. Inisiasikan suatu dictionary kosong, kemudian definisikan *key-value pair* dengan memetakan masing-masing huruf unik dengan masing-masing angka yang dihasilkan.
- 7. Kemudian substitusikan huruf-huruf pada baris operan dan hasil dengan *key-value pairs* yang ada pada dictionary.
- 8. Lakukan *typecasting* terhadap string angka, lalu jumlahkan dan periksa apakah sama dengan angka pada baris hasil.
- 9. Apabila sama maka cetak solusi, beserta waktu menemukan solusi, dan tambahkan 1 ke variabel jumlah solusi yang ditemukan.
- 10. Cetak jumlah solusi yang ditemukan setelah keluar dari loop while.

2. SOURCE CODE PROGRAM PYTHON

```
# main.py
import time
# init empty row and res
rows = []
```

```
res = ''
arrChar = []
def uniqueChar(letter):
 for i in range(len(arrChar)):
   if (arrChar[i] == letter):
lineCount = len(open('test.txt').readlines())
f = open('test.txt')
print("PERSOALAN cryptarithmetics adalah sebagai berikut:")
for l in f:
 rows.append(l.strip("\n+- "))
 print(l.strip('\n'))
 for j in range(len(l)):
    if (uniqueChar(l[j]) \text{ and } ord(l[j]) >= 65 \text{ and } ord(l[j]) <= 90):
     arrChar.append(l[j])
res = rows.pop(-1)
rows.pop(-1)
start time = time.time()
num = (10**len(arrChar))-1
found = 0
while (num >= 10**(len(arrChar)-1)):
 if ( (len(str(num)) == len(set(str(num)))) and (len(set(str(num))) ==
len(arrChar)) ):
```

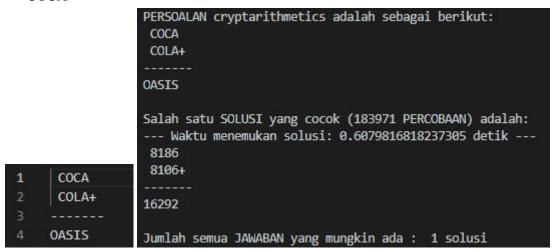
```
keyValue = {} # init empty dictionary
   for i in range(len(arrChar)):
     keyValue.update({arrChar[i] : str(num)[i]})
   fail = False
   sum = 0; j = 0
   while(j<len(rows) and not(fail)):</pre>
     storeStr = ""
     for k in range(len(rows[j])):
       storeStr += keyValue[rows[j][k]]
       if (k==0 \text{ and } keyValue[rows[j][k]]=='0'):
          fail = True
     sum += int(storeStr)
   if (not(fail)):
       storeStr = ""
       for i in range(len(res)):
          storeStr += keyValue[res[i]]
          if (i==0 and keyValue[res[i]]=='0'):
            fail = True
        sumRes = int(storeStr)
   if (sum==sumRes and not(fail)):
     print("\nSalah satu SOLUSI yang cocok (%s PERCOBAAN) adalah:" %
     print("--- Waktu menemukan solusi: %s detik ---" % (time.time() -
start time))
     file = open("test.txt", "r")
     for line in file:
        for character in line:
            print(keyValue[character],end="")
```

```
print(character,end="")
  found += 1
  print()
  num -= 1

# print jumlah solusi
print("\nJumlah semua JAWABAN yang mungkin ada : "+str(found)+" solusi")
```

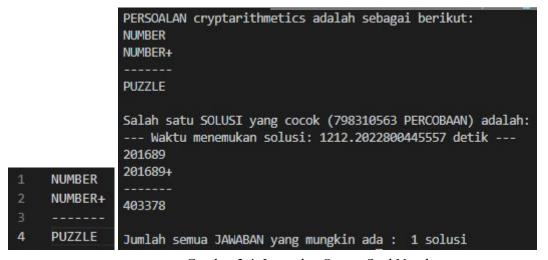
3. SKRINSHUT INPUT DAN OUTPUT 8 SOAL

- COCA



Gambar 1,2. Input dan Output Soal Coca

- NUMBER



Gambar 3,4. Input dan Output Soal Number

- HERE

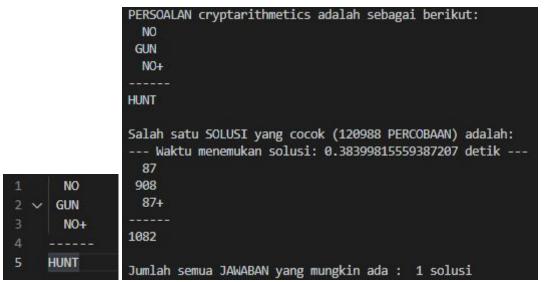
```
PERSOALAN cryptarithmetics adalah sebagai berikut:
HERE
SHE+
-----
COMES

Salah satu SOLUSI yang cocok (541897 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 1.159998893737793 detik ---
9454
894+
-----
10348

Tumlah semua JAWABAN yang mungkin ada : 1 solusi
```

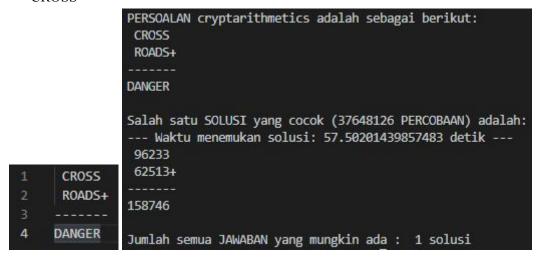
Gambar 5,6. Input dan Output Soal Here

- NO



Gambar 7,8. Input dan Output Soal No

- CROSS



Gambar 9,10. Input dan Output Soal Cross

- MEMO

```
PERSOALAN cryptarithmetics adalah sebagai berikut:

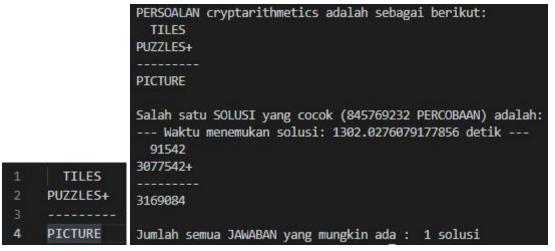
MEMO
FROM+
----
HOMER

Salah satu SOLUSI yang cocok (154269 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 0.5279934406280518 detik ---
8485
7358+
----
15843

Jumlah semua JAWABAN yang mungkin ada : 1 solusi
```

Gambar 11,12. Input dan Output Soal Number

- TILES



Gambar 13,14. Input dan Output Soal Number

- SEND

```
PERSOALAN cryptarithmetics adalah sebagai berikut:

SEND

MORE+
-----
MONEY

Salah satu SOLUSI yang cocok (4328918 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 7.543996810913086 detik ---
9567

SEND
MORE+
MORE+
3 -----
10652

Jumlah semua JAWABAN yang mungkin ada : 1 solusi
```

4. ALAMAT DRIVE YANG BERISI KODE PROGRAM

Program beserta dokumentasi berada pada tautan berikut.

Kelas K2: https://drive.google.com/drive/u/0/folders/1y7u0LIIha5tV8R6sr9vOnMFQqVzRVUf2

5. CEK LIST 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 cryptarithmetic hanya benar untuk persoalan cryptarihtmetic dengan dua buah operand.		✓
5. Solusi cryptarithmetic benar untuk persoalan cryptarihtmetic untuk lebih dari dua buah operand.	√	