

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.

CP	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 menyimpan 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 $10^n - 1$ 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 10^{n-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 = ''

# init array char DARI fungsi yang memasukan karakter unik ke array (STATIK
dan TIDAK BERUBAH)
arrChar = []

# function untuk melakukan pengecekan unik
def uniqueChar(letter):
    for i in range(len(arrChar)):
        if (arrChar[i]==letter):
            return False
    return True

# counting file line length
lineCount = len(open('test.txt').readlines())

# opening, printing and reading file
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]) and ord(l[j])>=65 and ord(l[j])<=90):
            arrChar.append(l[j])
res = rows.pop(-1)
rows.pop(-1)

# timer start
start_time = time.time()

# init variabel num untuk angka dari jumlah digit yang sama dengan huruf
unik, untuk keperluan inkremen angka
num = (10**len(arrChar))-1
# init counter solutions found
found = 0

# masuk loop untuk iterasi permutasi inkremental angka-angka yang mungkin
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

# define key-value pairs
for i in range(len(arrChar)):
    keyValue.update({arrChar[i] : str(num)[i]})

# substitute rows with key-value
fail = False
sum = 0; j = 0
while(j<len(rows) and not(fail)):
    storeStr = ""
    for k in range(len(rows[j])):
        storeStr += keyValue[rows[j][k]]
        if (k==0 and keyValue[rows[j][k]]=='0'):
            fail = True
    sum += int(storeStr)
    j += 1

# substitute res with key-value
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)

# check if math is true
if (sum==sumRes and not(fail)):
    print("\nSalah satu SOLUSI yang cocok (%s PERCOBAAN) adalah:" %
((10**len(arrChar))-num))
    print("--- Waktu menemukan solusi: %s detik ---" % (time.time() -
start_time))
    start_time = time.time()

file = open("test.txt", "r")
for line in file:
    for character in line:
        if (ord(character)>=65 and ord(character)<=90):
            print(keyValue[character],end="")
        else:

```

```

        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

1	COCA
2	COLA+
3	-----
4	OASIS

```

PERSOALAN cryptarithmetics adalah sebagai berikut:
COCA
COLA+
-----
OASIS

Salah satu SOLUSI yang cocok (183971 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 0.6079816818237305 detik ---
8186
8106+
-----
16292

Jumlah semua JAWABAN yang mungkin ada : 1 solusi

```

Gambar 1,2. Input dan Output Soal Coca

- NUMBER

1	NUMBER
2	NUMBER+
3	-----
4	PUZZLE

```

PERSOALAN cryptarithmetics adalah sebagai berikut:
NUMBER
NUMBER+
-----
PUZZLE

Salah satu SOLUSI yang cocok (798310563 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 1212.2022800445557 detik ---
201689
201689+
-----
403378

Jumlah semua JAWABAN yang mungkin ada : 1 solusi

```

Gambar 3,4. Input dan Output Soal Number

- HERE

1	✓	HERE
2		SHE+
3		-----
4		COMES

```

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

Jumlah semua JAWABAN yang mungkin ada : 1 solusi

```

Gambar 5,6. Input dan Output Soal Here

- NO

1		NO
2	✓	GUN
3		NO+
4		-----
5		HUNT

```

PERSOALAN cryptarithmetics adalah sebagai berikut:
NO
GUN
NO+
-----
HUNT

Salah satu SOLUSI yang cocok (120988 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 0.38399815559387207 detik ---
87
908
 87+
-----
1082

Jumlah semua JAWABAN yang mungkin ada : 1 solusi

```

Gambar 7,8. Input dan Output Soal No

- CROSS

1		CROSS
2		ROADS+
3		-----
4		DANGER

```

PERSOALAN cryptarithmetics adalah sebagai berikut:
CROSS
ROADS+
-----
DANGER

Salah satu SOLUSI yang cocok (37648126 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 57.50201439857483 detik ---
96233
62513+
-----
158746

Jumlah semua JAWABAN yang mungkin ada : 1 solusi

```

Gambar 9,10. Input dan Output Soal Cross

- MEMO

1	MEMO
2	FROM+
3	-----
4	HOMER

```
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

1	TILES
2	PUZZLES+
3	-----
4	PICTURE

```
PERSOALAN cryptarithmetics adalah sebagai berikut:
TILES
PUZZLES+
-----
PICTURE

Salah satu SOLUSI yang cocok (845769232 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 1302.0276079177856 detik ---
91542
3077542+
-----
3169084

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

Gambar 13,14. Input dan Output Soal Number

- SEND

1
2
3
4

SEND
MORE+

MONEY

Salah satu SOLUSI yang cocok (4328918 PERCOBAAN) adalah:
--- Waktu menemukan solusi: 7.543996810913086 detik ---
9567
1085+

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/1y7u0LIiha5tV8R6sr9vOnMFOqVzRVUf2>

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 cryptarithmic hanya benar untuk persoalan cryptarihtmetic dengan dua buah operand.		✓
5. Solusi cryptarithmic benar untuk persoalan cryptarihtmetic untuk lebih dari dua buah operand.	✓	