

Laporan Tugas Kecil 1 IF2211 Strategi Algoritma

Semester II tahun 2020/2021

Penyelesaian *Cryptarithmic* dengan Algoritma *Brute Force*



Disusun Oleh:

Habibina Arif Muzayyan (13519125)

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG**

2021

BAB I

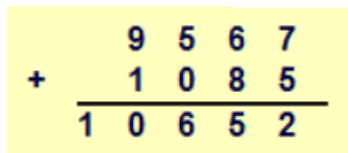
Deskripsi Masalah

Cryptarithmic (atau *cryptarithm*) adalah sebuah puzzle penjumlahan di dalam matematika dimana angka diganti dengan huruf. Setiap angka dipresentasikan dengan huruf yang berbeda. Deskripsi permainan ini adalah: diberikan sebuah penjumlahan huruf, carilah angka yang merepresentasikan huruf-huruf tersebut.

Contoh:

$$\begin{array}{r} \text{S E N D} \\ + \text{M O R E} \\ \hline \text{M O N E Y} \end{array}$$

Solusinya adalah:


$$\begin{array}{r} 9 \ 5 \ 6 \ 7 \\ + 1 \ 0 \ 8 \ 5 \\ \hline 1 \ 0 \ 6 \ 5 \ 2 \end{array}$$

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

Persoalan *Cryptarithmic* dapat diselesaikan dengan algoritma *brute force*, berikut ini adalah langkah-langkah penyelesaian *Cryptarithmic* dengan algoritma *brute force*:

1. Menerima masukannya berupa operan-operan dan hasil bertipe string
2. Substitusi huruf-huruf dengan semua angka-angka yang mungkin
3. Uji setiap kemungkinan angka apakah penjumlahan operan sama dengan hasil

BAB II

Source Program

Penyelesaian persoalan *cryptarithmatic* berikut ini diselesaikan dengan algoritma *brute force* dalam program berbahasa python

```
import time
import os

# fungsi untuk konversi kata dengan angka yang sesuai
def konversi(kata, subs):
    s=''
    for huruf in kata:
        idx=subs[0].index(huruf)
        idx=subs[0].index(huruf)
        s=s+str(subs[1][idx])
    return int(s)

# fungsi untuk permutasi semua kemungkinan angka
def permut(arr,n):
    if n<=0:
        return [[]]
    dummy=[]
    for i in range(len(arr)):
        current=arr[i]
        remain=arr[:i]+arr[i+1:]
        for p in permut(remain,n-1):
            dummy.append([current]+p)
    return dummy

# menerima input dari folder test
cur_dir = os.path.dirname(os.path.realpath(__file__))
new_dir = cur_dir[:len(cur_dir)-3]+'test\\'
namafile = input("Masukkan nama file: ")
f = open(new_dir+namafile)

start = time.time()
data = f.read()
baris = data.split()

hasil=[]
for huruf in baris[len(baris)-1]:
    hasil.append(huruf)
```

```

operan=[]
for i in range(len(baris)-3):
    dummy=[]
    for huruf in baris[i]:
        dummy.append(huruf)
    operan.append(dummy)

dummy=[]
for i in range(len(baris[len(baris)-3])-1):
    dummy.append(baris[len(baris)-3][i])
operan.append(dummy)

huruf=operan+list(hasil)
union=sorted(list(set().union(*huruf)))

# mencari penyelesaian dengan menguji satu-satu permutasinya
angka=[0,1,2,3,4,5,6,7,8,9]
counter=0
solusi=False
for p in permut(angka,len(union)):
    counter+=1
    subs=[union]+[p]
    if (sum(konversi(opr,subs) for opr in operan) == konversi(hasil,subs)) and
(konversi(op,subs) >= pow(10,len(op)-
1) for op in operan) and (konversi(hasil,subs) >= pow(10,len(hasil)-1)):
        solusi=True
        break;

end=time.time()

# menuliskan output
print(data)
print()
if solusi:
    for i in range(len(operan)-1):
        print(konversi(operan[i],subs))
    print(konversi(operan[len(operan)-1],subs),end='+\n')
    print('-----')
    print(konversi(hasil,subs))
else:
    print("Tidak ada solusi")
print()
print("Waktu eksekusi programnya adalah "+str(end-start))
print("Total tes yang dilakukan adalah "+str(counter))

```

```
dummy=input() # menerima sembarang input agar dapat menampilkan output pada file .exe
```

BAB III

Eksperimen

Berikut ini adalah tangkapan layar dari eksekusi program penyelesaian persoalan *cryptarithm*:

```
Masukkan nama file: test1.txt
NUMBER
NUMBER+
-----
PUZZLE

201689
201689+
-----
403378

Waktu eksekusi programnya adalah 70.70876622200012
Total tes yang dilakukan adalah 2490667
```

```
Masukkan nama file: test2.txt
TILES
PUZZLES+
-----
PICTURE

91542
3077542+
-----
3169084

Waktu eksekusi programnya adalah 75.89939475059509
Total tes yang dilakukan adalah 2346083
```

```
Masukkan nama file: test3.txt
CLOCK
TICK
TOCK+
-----
PLANET

90892
6592
6892+
-----
104376

Waktu eksekusi programnya adalah 67.48393440246582
Total tes yang dilakukan adalah 1807451
```

Masukkan nama file: test4.txt

COCA

COLA+

OASIS

8186

8106+

16292

Waktu eksekusi programnya adalah 1.872140884399414

Total tes yang dilakukan adalah 103951

Masukkan nama file: test5.txt

HERE

SHE+

COMES

9454

894+

10348

Waktu eksekusi programnya adalah 3.6760849952697754

Total tes yang dilakukan adalah 86768

```
Masukkan nama file: test6.txt
DOUBLE
DOUBLE
TOIL+
-----
TROUBLE

798064
798064
1936+
-----
1598064

Waktu eksekusi programnya adalah 28.213728427886963
Total tes yang dilakukan adalah 258950
```

```
Masukkan nama file: test7.txt
NO
GUN
NO+
-----
HUNT

87
908
87+
-----
1082

Waktu eksekusi programnya adalah 2.2461659908294678
Total tes yang dilakukan adalah 139416
```

```
Masukkan nama file: test8.txt
THREE
THREE
TWO
TWO
ONE+
-----
ELEVEN

84611
84611
803
803
391+
-----
171219

Waktu eksekusi programnya adalah 35.73101997375488
Total tes yang dilakukan adalah 513693
```



```
Masukkan nama file: test9.txt
CROSS
ROADS+
-----
DANGER

96233
62513+
-----
158746

Waktu eksekusi programnya adalah 60.67718768119812
Total tes yang dilakukan adalah 2144748
```

```
Masukkan nama file: test10.txt
MEMO
FROM+
-----
HOMER

8485
7358+
-----
15843

Waktu eksekusi programnya adalah 1.431105613708496
Total tes yang dilakukan adalah 70938
```

LAMPIRAN

[Tucil1_13519125 - Google Drive](#)

[habibinaarif/Tucil1_13519125 \(github.com\)](#)

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