

Tugas Kecil 1 IF2211 Strategi Algoritma

Semester II tahun 2020/2021

Nizamixavier Rafif Lutvie (13519085)

A. Algoritma Brute Force

Algoritma *brute force* adalah teknik pemecahan masalah yang sangat umum yang dilakukan dengan mencoba semua kemungkinan yang ada dan memeriksa apakah kemungkinan tersebut memenuhi pernyataan masalah. Algoritma *brute force* sangat cocok untuk masalah yang memiliki kemungkinan solusi yang sedikit, tetapi kurang tepat untuk mengatasi masalah dengan dengan kemungkinan solusi yang banyak. Algoritma *brute force* juga menjamin ditemukannya kemungkinan solusi yang tepat dari sebuah masalah.

Pada pemecahan masalah *cryptarithm* yang diberikan, digunakan algoritma *brute force*. Pertama-tama, perlu dimasukkan nama file yang akan digunakan sebagai tes *cryptarithm*, misalnya “test.txt”. Selanjutnya, akan dibaca setiap baris dari file tersebut. Setiap baris akan dimasukkan dalam sebuah array. Akan dibuat beberapa array dan string yang diperlukan, yaitu array yang mengandung setiap kata yang ada, array yang mengandung setiap huruf unik yang ada, string yang mengandung nilai setiap huruf unik yang ada, array yang mengandung huruf unik yang merupakan huruf pertama pada setiap kata, dan array yang mengandung nilai setiap huruf unik yang merupakan huruf pertama pada setiap kata.

Untuk mencoba semua kemungkinan solusi, nilai awal setiap huruf akan di-*set* menjadi 0, sehingga string yang berisi nilai huruf akan menjadi “0000...” dan akan dicoba semua kemungkinan sampai mencapai “9999...”. Kemudian, akan dijalankan suatu loop. Loop berfungsi untuk menambah string dengan cara mengubah tipe string menjadi int terlebih dahulu kemudian ditambah dengan 1 kemudian dikembalikan menjadi tipe string. Agar tidak mengalami error saat ternyata tidak ada solusi yang memungkinkan, akan diberikan batas percobaan kemungkinan. Selama nilai string masih di bawah “9999...”, loop akan terus berjalan. Jika nilai setiap huruf tidak unik, atau nilai huruf pertama dalam suatu kata adalah 0, atau nilai penjumlahan kata tidak sama dengan hasil, penambahan nilai pada string nilai huruf sebesar 1 akan terus dilakukan sampai semua kondisi tersebut tidak memenuhi, artinya telah ditemukan jawaban yang benar atau percobaan telah mencapai batas (tidak ada solusi).

Jika tidak ditemukan solusi dari persoalan, akan diberi output yang menyatakan bahwa tidak ada solusi. Jika terdapat solusi dari persoalan, akan diberi output isi soal dan konversi soal tersebut yang setiap hurufnya telah digantikan oleh nilai huruf yang sebenarnya, serta tampilan durasi eksekusi program dan jumlah kemungkinan yang dicoba.

B. Source Code

Berikut adalah source code program. (Bahasa: Python)

```
import time
import os

def kalku(kata,key,keynum):
    a = 0
    for i in range(len(kata)):
        a = a + int(keynum[key.index(kata[i])]) * (10**(len(kata) - 1 - i))
    return a

counter = 1
katahasil = ""
katastorage = []
hurufpertama = ""

name = str(input("Masukkan nama file: "))
cur_path = os.path.dirname(__file__)
new_path = os.path.relpath('..\test\' + name, cur_path)
f = open(new_path,"r")
kata = f.readline()
kata = kata.rstrip('\n')

while(kata[len(kata)-1] != "+"):
    katahasil = katahasil + kata
```

```
katastorage.append(kata)

hurufpertama = hurufpertama + kata[0]

kata = f.readline()

kata = kata.rstrip('\n')

kata2 = kata

kata = kata[:-1]

katahasil = katahasil + kata

katastorage.append(kata)

hurufpertama = hurufpertama + kata[0]


garis = f.readline()

garis = garis.rstrip('\n')

kata3 = f.readline()

kata3 = kata3.rstrip('\n')

f.close()


start_time = time.time()


katahasil = katahasil + kata3

hurufpertama = hurufpertama + kata3[0]


key = ['*']

for i in range(len(katahasil)):

    if(key.count(katahasil[i]) == 0):

        key.append(katahasil[i])

keynum = "1" + ("0" * (len(key) - 1))

keynum2 = keynum[1:]
```

```
angkapertama = []
for i in range(len(hurufpertama)):
    angkapertama.append(keynum[key.index(hurufpertama[i])])

total = 0
for i in range(len(katastorage)):
    total = total + kalku(katastorage[i],key,keynum)

while((keynum[0] == '1') and ((total != kalku(kata3,key,keynum)) or (len(keynum2) !=
len(set(keynum2)))) or (angkapertama.count('0') != 0)):
    keynum = str(int(keynum) + 1)
    keynum2 = keynum[1:]
    counter = counter + 1

    angkapertama = []
    for i in range(len(hurufpertama)):
        angkapertama.append(keynum[key.index(hurufpertama[i])])

    total = 0
    for i in range(len(katastorage)):
        total = total + kalku(katastorage[i],key,keynum)

for i in range(len(katastorage) - 1):
    print(katastorage[i])
print(kata2)
print(garis)
print(kata3)
print('\n')
```

```

if(keynum[0] == '1'):
    print('Solusi:\n')
    for i in range(len(katastorage) - 1):
        print(kalku(katastorage[i],key,keynum))
    print(str(kalku(kata,key,keynum)) + "+")
    print(garis)
    print(kalku(kata3,key,keynum))
    print('\n')
else:
    print('Tidak ada solusi yang memenuhi\n')

print('Jumlah tes yang dilakukan:',counter,'kali')
print("Waktu eksekusi: %.10f detik" % (time.time() - start_time))

```

C. Screenshot

Berikut adalah hasil *screenshot* dari eksekusi program.

```

C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 1.txt
HERE
SHE+
-----
COMES

Solusi:
9454
894+
-----
10348

Jumlah tes yang dilakukan: 9458104 kali
Waktu eksekusi: 144.6808934212 detik

-----
(program exited with code: 0)
Press any key to continue . . .

```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 2.txt
EAT
THAT+
-----
APPLE

Solusi:

819
9219+
-----
10038

Jumlah tes yang dilakukan: 819204 kali
Waktu eksekusi: 11.8740077019 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 3.txt
NO
GUN
NO+
-----
HUNT

Solusi:

87
908
87+
-----
1082

Jumlah tes yang dilakukan: 879013 kali
Waktu eksekusi: 12.8566861153 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 4.txt
MEMO
FROM+
-----
HOMER

Solusi:

8485
7358+
-----
15843

Jumlah tes yang dilakukan: 845732 kali
Waktu eksekusi: 13.1133506298 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 5.txt
ABCDEF
ABCDEF
ABCDEF+
-----
BCDEFA

Solusi:

142857
142857
142857+
-----
428571

Jumlah tes yang dilakukan: 142858 kali
Waktu eksekusi: 3.6160397530 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 6.txt
MOST
MOST+
-----
TOKYO

Solusi:

6271
6271+
-----
12542

Jumlah tes yang dilakukan: 627155 kali
Waktu eksekusi: 10.1029746532 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 7.txt
COCA
COLA+
-----
OASIS

Solusi:

8186
8106+
-----
16292

Jumlah tes yang dilakukan: 816030 kali
Waktu eksekusi: 12.7899997234 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 8.txt
LOGIC
LOGIC+
-----
PROLOG

Solusi:

90452
90452+
-----
180904

Jumlah tes yang dilakukan: 9045219 kali
Waktu eksekusi: 174.0126223564 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

```
C:\WINDOWS\SYSTEM32\cmd.exe
Masukkan nama file: 9.txt
USA
USSR+
-----
PEACE

Solusi:

932
9338+
-----
10270

Jumlah tes yang dilakukan: 9328108 kali
Waktu eksekusi: 144.8279817104 detik

-----
(program exited with code: 0)
Press any key to continue . . .
```

D. Alamat Program

Berikut adalah alamat menuju repository Github yang berisi kode program.

https://github.com/thivh/Tucil01_Stima

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa kesalahan (no syntax error)	✓	
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 <i>operand</i> .		✓
5. Solusi <i>cryptarithmic</i> benar untuk persoalan <i>cryptarithmic</i> untuk lebih dari dua buah operand.	✓	