Tugas Kecil 1 IF2211 Strategi Algoritma
Penyelesaian *Cryptarithmetic* dengan Algoritma *Brute Force*
Semester II Tahun 2020/2021

Nama (NIM/Kelas):
Giovani Anggasta (NIM 13519155 / K03)

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

BAB 1
Algoritma *Brute Force*

1. Menerima input berupa string dengan membaca file eksternal
2. Memasukkan string yang dibaca ke dalam satu list yaitu list ‘words’
3. Membentuk list yang berisi huruf-huruf unik yang terdapat pada list ‘words’
4. Membentuk list ‘soal’ yang berisi string pada list ‘words’ yang menjadi soal dan membentuk list ‘jawaban’ yang berisi string pada list ‘words’ yang menjadi jawaban
5. Membentuk list ‘first_letter’ yang berisi huruf pertama dari masing-masing string yang ada pada list ‘words’
6. Melakukan permutasi terhadap list angka dimana list angka telah diberikan yaitu dari 0-9.
7. Hasil permutasi dimasukkan kedalam dictionary.
8. Dihitung total penjumlahan dari list ‘soal’ dan list ‘jawaban’ dengan fungsi adding_number dimana fungsi tersebut berfungsi menjumlahkan perkalian hasil permutasi masing-masing huruf dengan faktor 10.
9. Apabila total adding_number list ‘soal’ dan list ‘jawaban’ memiliki nilai yang sama selanjutnya akan di cek apakah terdapat huruf pertama dari masing-masing string pada list ‘words’ yang hasil permutasinya adalah nol. Jika ya maka akan diulang langkah ke-6 sampai langkah ke-9. Jika tidak maka akan dikeluarkan output.

BAB 2
Source Program

```python
# tucil 1 :)

import re
import time

def from_file (nama_file):
    open_file = open(nama_file, 'r')
    read_file = open_file.read()
    alphabet = re.sub('[\W_]+', ' ', read_file)
    words = alphabet.split()
    return words

def permutasi (angka):
    if (len(angka) == 0):
        return []
    elif (len(angka) == 1):
        return [angka]
    else:
        list_angka = []
        for i in range (len(angka)):
            b = angka[i]
            cur_list = angka[:i] + angka[i+1:]
            for p in permutasi(cur_list):
                list_angka.append([b] + p)
        return list_angka

def adding_number (word, angka):
    total = 0
    faktor = 1
    for letter in reversed(word):
        total += faktor*int(angka[letter])
        faktor *= 10
    return total

def isNol (solusi,firstLetter_list):
    condition = True
    for i in range (len(firstLetter_list)):
        for key in solusi:
            if (key == firstLetter_list[i]):
                if (solusi[key] == '0'):
                    condition = False
                    break
    return condition


```

```
def solving (angka,words):
    # huruf pada tiap kata
    letters = []
    for kata in words:
        for i in range (len(kata)):
            letters.append(kata[i])

    letter_set = set()
    for i in range(len(letters)):
        letter_set.update(letters[i])
    letter_list = (list(letter_set))
    letter_list.sort()

    # pisahin soal sama jawaban
    count = 0
    for kata in words:
        count += 1

    i = count-1
    j = 0
    soal = []
    jawab = []
    while (i >= 0):
        if (i == count-1):
            jawab.insert(j,words[i])
        elif (i < count-1):
            soal.insert(j,words[i])
            j += 1
        i -= 1

    # huruf pertama tiap kata
    first_letter = []
    for kata in words:
        for i in range (len(kata)):
            if (i == 0):
                first_letter.append(kata[i])

    firstLetter_set = set()
    for i in range(len(first_letter)):
        firstLetter_set.update(first_letter[i])
    firstLetter_list = (list(firstLetter_set))
    firstLetter_list.sort()

```

```
    percobaan = 0
    for p in permutasi(angka):
        solusi = dict(zip(letter_list,p))
        total_soal = sum(adding_number(word,solusi) for word in soal)
        total_jawab = sum(adding_number(word,solusi) for word in jawab)
        nol = isNol(solusi,firstLetter_list)
        if (total_soal == total_jawab):
            if (nol == True):
                sol = solusi
        percobaan += 1

    printJawaban(words,sol)
    print("Jumlah percobaan : ", percobaan)

def printSoal (words):
    print('S O A L :')
    # pisahin soal sama jawaban
    count = 0
    for kata in words:
        count += 1

    i = count-1
    j = 0
    soal = []
    jawab = []
    while (i >= 0):
        if (i == count-1):
            jawab.insert(j,words[i])
        elif (i < count-1):
            soal.insert(j,words[i])
            j += 1
        i -= 1

    for kata in soal:
        print(kata)
    for kata in jawab:
        separator = len(kata)*'-' + ' +'
        print(separator)
        print(kata)
    print('\n')

```

```
    percobaan = 0
    for p in permutasi(angka):
        solusi = dict(zip(letter_list,p))
        total_soal = sum(adding_number(word,solusi) for word in soal)
        total_jawab = sum(adding_number(word,solusi) for word in jawab)
        nol = isNol(solusi,firstLetter_list)
        if (total_soal == total_jawab):
            if (nol == True):
                sol = solusi
        percobaan += 1

    printJawaban(words,sol)
    print("Jumlah percobaan : ", percobaan)

def printSoal (words):
    print('S O A L :')
    # pisahin soal sama jawaban
    count = 0
    for kata in words:
        count += 1

    i = count-1
    j = 0
    soal = []
    jawab = []
    while (i >= 0):
        if (i == count-1):
            jawab.insert(j,words[i])
        elif (i < count-1):
            soal.insert(j,words[i])
            j += 1
        i -= 1

    for kata in soal:
        print(kata)
    for kata in jawab:
        separator = len(kata)*'-' + ' +'
        print(separator)
        print(kata)
    print('\n')

```

```python
def printJawaban (words,solusi):
    print("J A W A B A N :")
    # pisahin soal sama jawaban
    count = 0
    for kata in words:
        count += 1

    i = count-1
    j = 0
    soal = []
    jawab = []
    while (i >= 0):
        if (i == count-1):
            jawab.insert(j,words[i])
        elif (i < count-1):
            soal.insert(j,words[i])
            j += 1
        i -= 1

    for kata in soal:
        huruf = list(kata)
        for i in range (len(huruf)):
            for key in solusi:
                if (huruf[i] == key):
                    print(solusi[key],end='')
        print('')

    for kata in jawab:
        huruf = list(kata)
        separator = len(huruf)*'-' + ' +'
        print(separator)
        for i in range (len(huruf)):
            for key in solusi:
                if (huruf[i] == key):
                    print(solusi[key],end='')
        print('')
    print('\n')

```

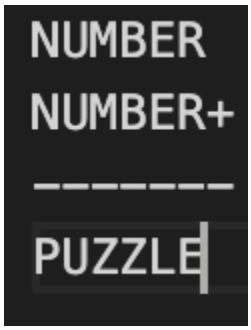
```python
# main
nama_file = input("masukkan nama file : ")
direct = '../test/'
words = from_file(direct+nama_file)
start_time = time.time()
angka = list('0123456789')
printSoal(words)
solving(angka,words)
print("Waktu yang dibutuhkan : %s seconds" % (time.time() - start_time))
```

BAB 3
Screenshot Hasil Input Output

1. File : test1.txt

Input	Output
	masukkan nama file : test1.txt S O A L : MORE SEND ----- + MONEY J A W A B A N : 1085 9567 ----- + 10652 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 44.31208896636963 seconds

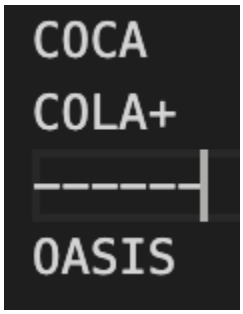
2. File : test2.txt

Input	Output
	masukkan nama file : test2.txt S O A L : NUMBER NUMBER ------ + PUZZLE J A W A B A N : 201689 201689 ------ + 403378 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 48.97176718711853 seconds

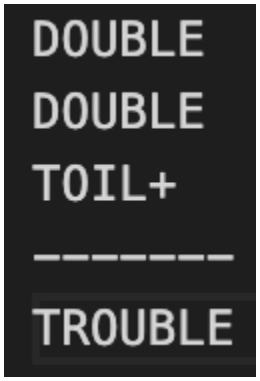
3. File : test3.txt

Input	Output
	masukkan nama file : test3.txt S O A L : TOCK TICK CLOCK ------ + PLANET J A W A B A N : 6892 6592 90892 ------ + 104376 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 54.91173481941223 seconds

4. File : test4.txt

Input	Output
	masukkan nama file : test4.txt S O A L : COLA COCA ----- + OASIS J A W A B A N : 8106 8186 ----- + 16292 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 42.276569843292236 seconds

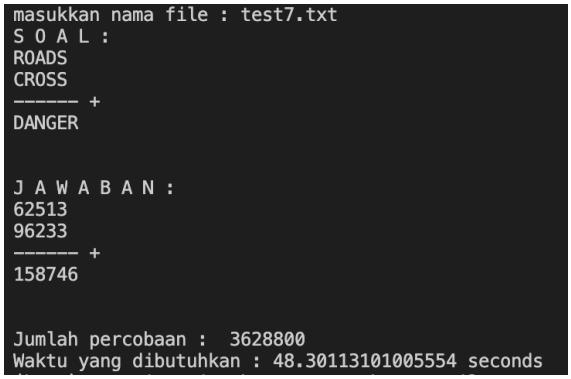
5. File : test5.txt

Input	Output
	masukkan nama file : test5.txt S O A L : TOIL DOUBLE DOUBLE ------- + TROUBLE J A W A B A N : 1936 798064 798064 ------- + 1598064 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 53.5408501625061 seconds

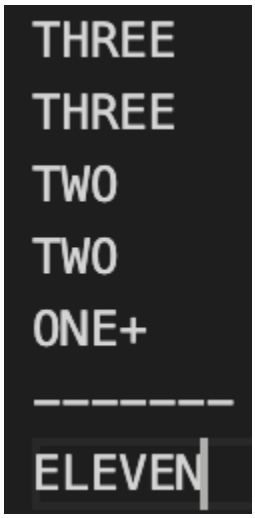
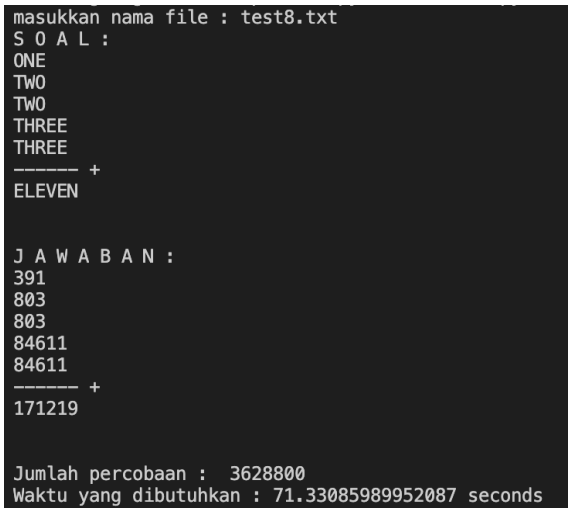
6. File : test6.txt

Input	Output
	masukkan nama file : test6.txt S O A L : FROM MEMO ----- + HOMER J A W A B A N : 7358 8485 ----- + 15843 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 44.04651427268982 seconds

7. File : test7.txt

Input	Output
	

8. File : test8.txt

Input	Output
	

Drive Link

https://drive.google.com/drive/folders/1lS2ey1lFlC9lcgg2rD10qnJcwbqMcqYB?usp=sharing