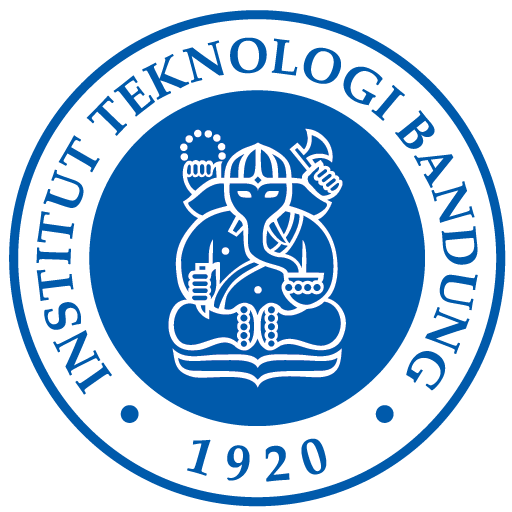
**CRYPHTARITMETIC SOLVER**

**LAPORAN TUGAS KECIL 1**

**MATA KULIAH IF2211 STRATEGI ALGORITMA**



Disusun oleh: Aria Bachrul Ulum Berlian (13519115)

**Program Studi Teknik Informatika**

**Sekolah Teknik Elektro dan Informatika**

**Institut Teknologi Bandung**

**Semester 2 Tahun 2020/2021**

**Algoritma Cryptarithmetic**

Program cryptarithmetic ini menggunakan algoritma brute force. Program ini awalnya membaca file txt. Lalu, program ini akan membersihkan data tersebut dan memasukannya ke dalam larik. Program akan mulai mencatat waktu. Dari array of character tadi, program membuat larik baru yang berisi huruf-huruf yang unik. Lalu, dibuat larik berisi angka dari 0 hingga 9. Dibuatlah sebuah kamus yang *key-*nya adalah elemen dari larik huruf unik dam *value*-nya adalah elemen dari larik berisi angka. Selanjutnya, larik data dari file akan diubah menurut kamus. Bila menurut aturan penjumlahan data ini sudah benar maka akan ditampilkan ke layar. Jika tidak, maka larik angka akan diacak lalu pembuatan kamus diulang lagi. Pengecekan dilakukan lagi dan seterusnya.

Pengacakan angka dilakukan dengan cara permutasi. Hal ini dilakukan agar tidak ada susunan angka yang berulang. Semua susunan akan dicoba satu per satu hingga solusi yang memenuhi persamaan aritmatikanya. Banyaknya percobaan maksimal, yakni nPr, untuk n adalah jumlah kemungkinan angka, dan r adalah jumlah huruf yang berbeda.

**SOURCE CODE**

from time import time

def permutasi(arr):

    if len(arr) == 0:

        return []

    if len(arr) == 1:

        return [arr]

    l = []

    for i in range(len(arr)):

        N = arr[i]

        sisa = arr[:i] + arr[i+1:]

        for p in permutasi(sisa):

            l.append([N] + p)

    return l

def nilai(kata, dict):

    total = 0

    s = 1

    balikin = kata[::-1]

    for i in range(len(balikin)):

        total += dict[balikin[i]] \* s

        s \*= 10

    return total

def print\_solusi(solusi):

    n = len(solusi)

    for i in range(n-2):

        print(solusi[i])

    print("{} +".format(solusi[n-2]))

    print("---------")

    print(solusi[n-1])

f = open("D:\Kuliahan\SMT 4\Strategi Algoritma\Tucil\\1\soal.txt", "r")

isi = f.read()

data = isi.split("\n")

idxMax = len(data) - 1

data.pop(idxMax - 1)

for i in range(idxMax):

    data[i] = data[i].replace("+", " ").strip() #Bersihin data

    data[i] = list(data[i])

f.close()

start = time()  # Mulai hitung waktu

angka = list(range(10))   #Kemungkinan angka: 0-9

# Semua huruf yang ada dalam string

char\_list = []

for i in range(idxMax):

    n = len(data[i])

    for j in range(n):

        char\_list.append(data[i][j])

char\_list = list(set(char\_list))

coba = 0  # Menghitung jumlah percobaan

for tes in permutasi(angka):

    char\_dict = dict(zip(char\_list, tes[::-1]))  # {char:angka}

    nol = 0

    for i in range(len(data)):

        if(char\_dict[data[i][0]] == 0) :

            nol += 1

    if nol != 0:

        continue

    else:

        hasil = 0

        jawaban = []

        for i in range(len(data)-1):

            hasil += nilai(data[i],char\_dict)

            jawaban.append(nilai(data[i],char\_dict))

        if (hasil == nilai(data[-1], char\_dict)):

            jawaban.append(nilai(data[-1], char\_dict))

            end = time()

            print(isi)

            print()

            print\_solusi(jawaban)

            print()

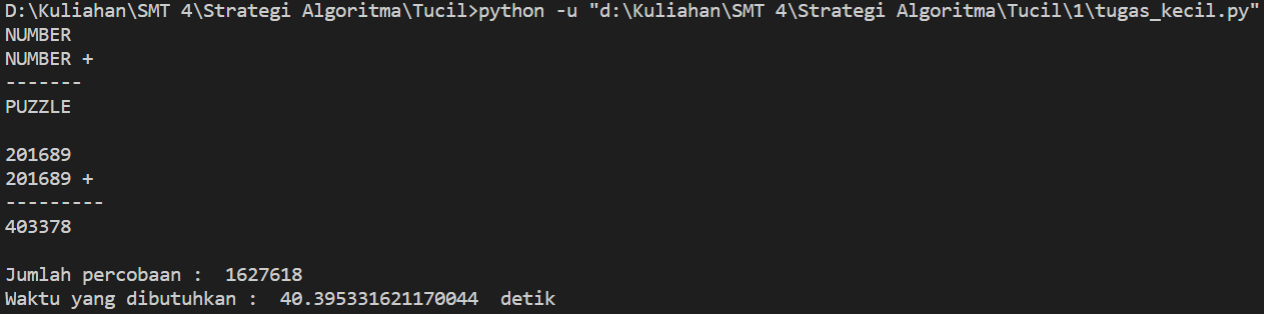
            print("Jumlah percobaan :", coba)

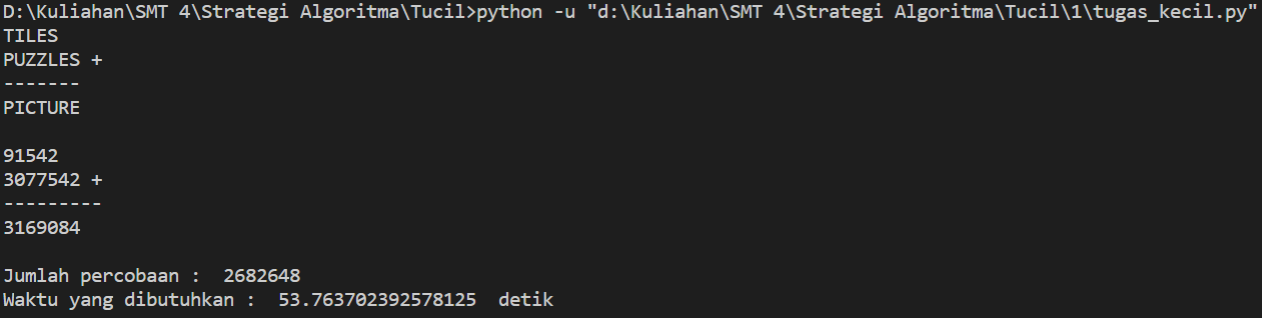
            print("Waktu yang dibutuhkan :", end-start,"detik")

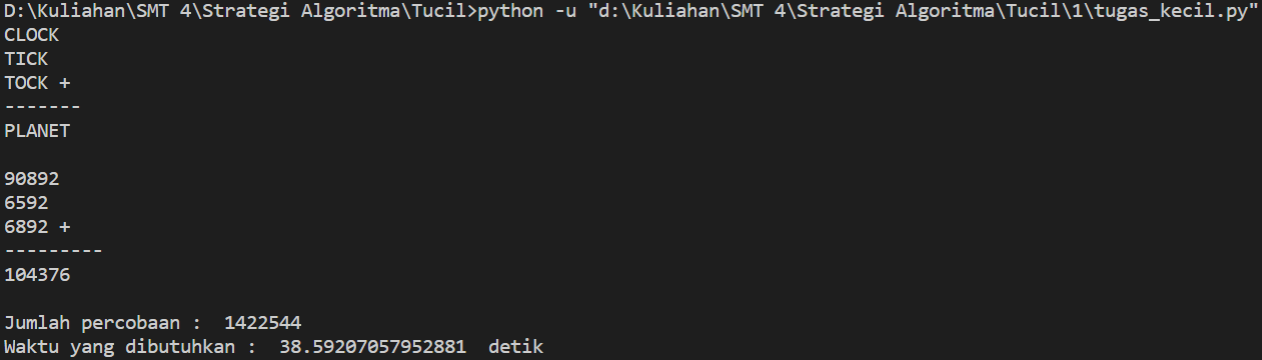
            break

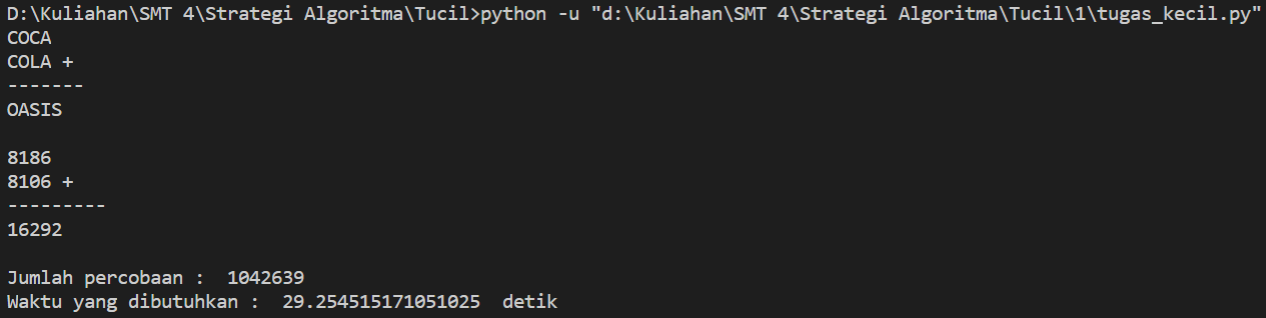
    coba += 1

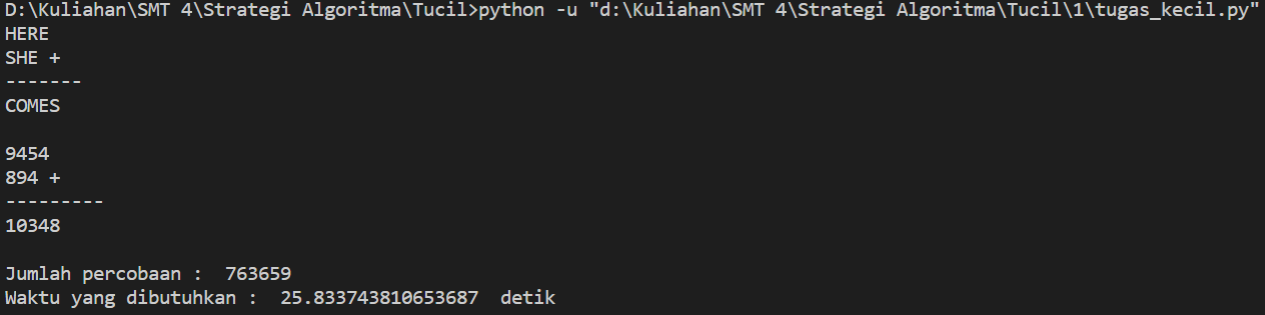
**PENGUJIAN**

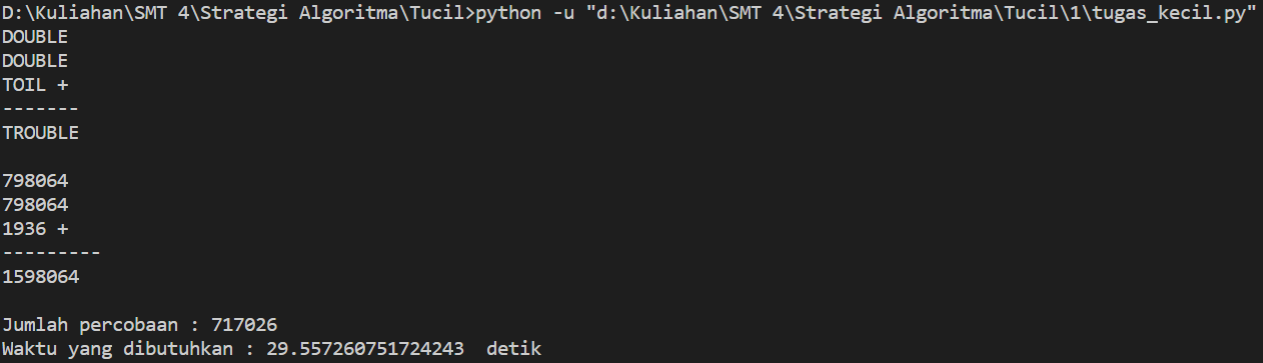


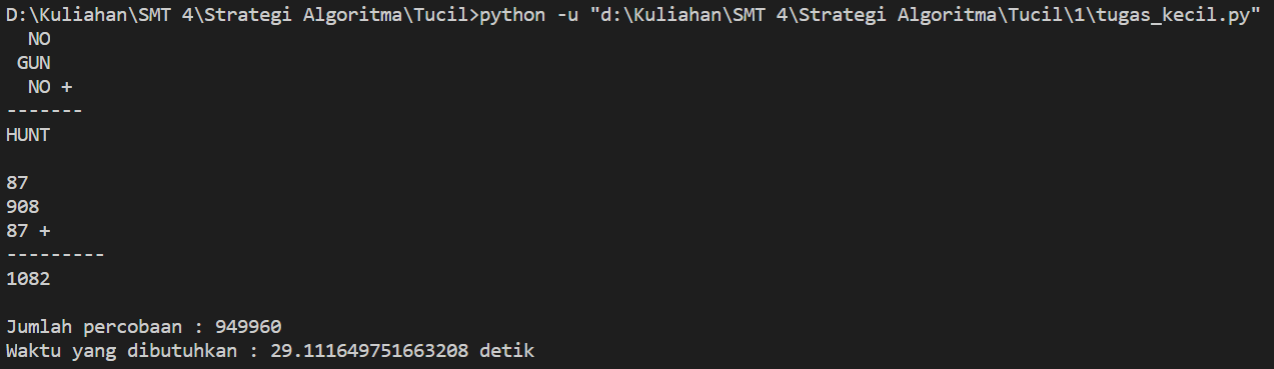


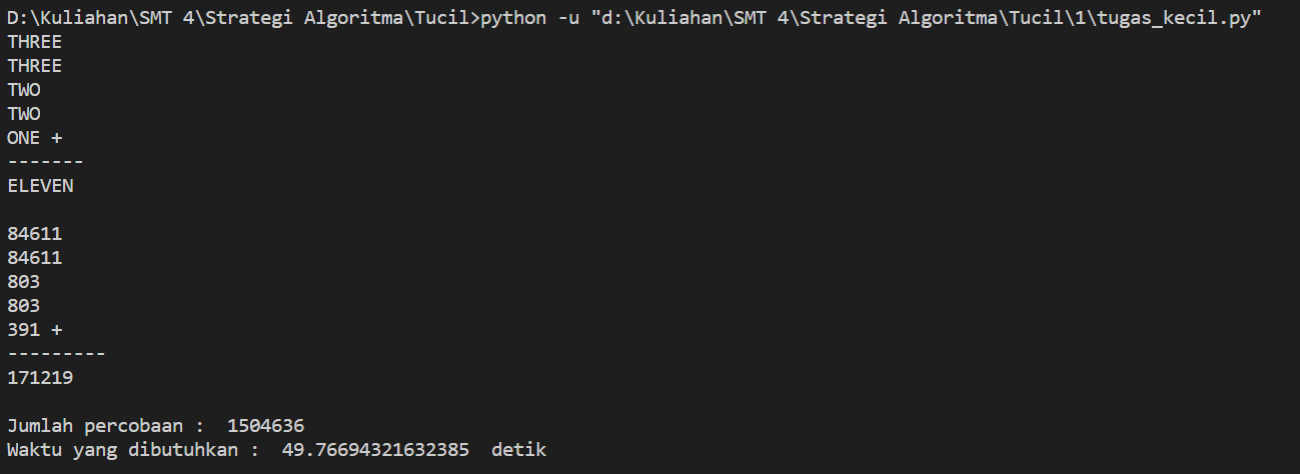


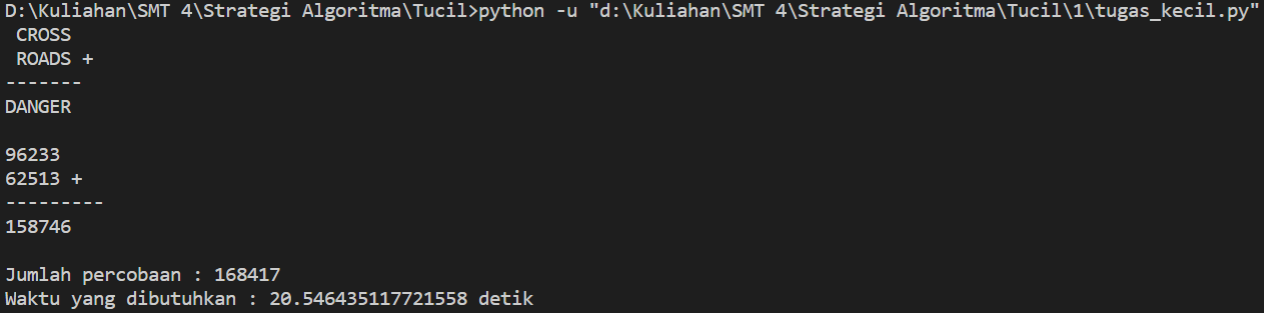


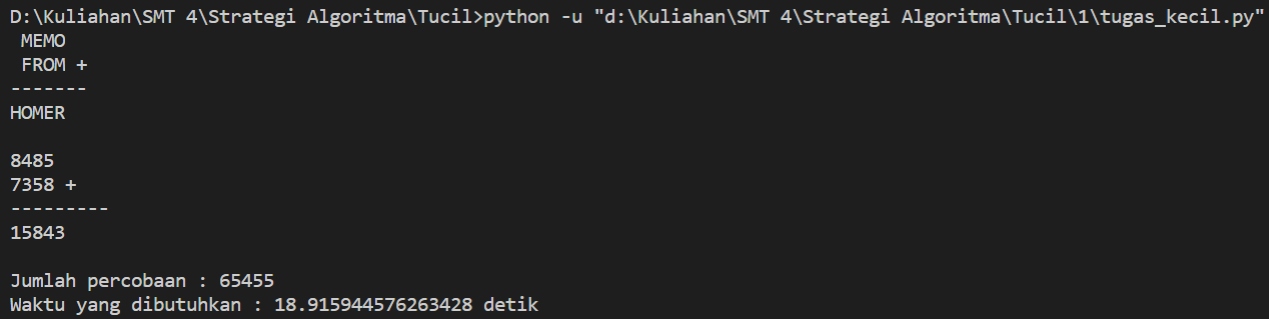


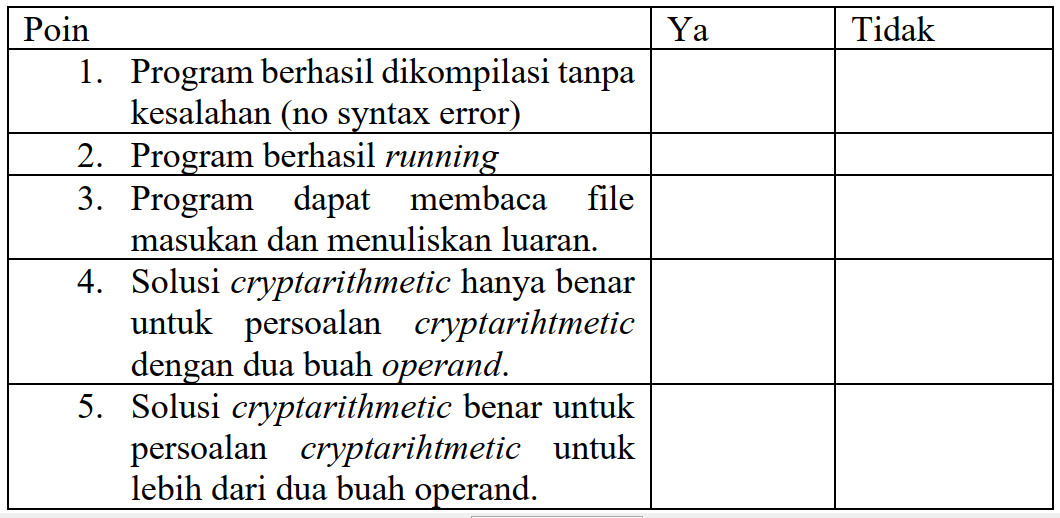














https://drive.google.com/file/d/1PCSulvrILi0n2D1tVuHZE4UxCopschD\_/view?usp=sharing