

Tugas Kecil 1 IF2211 Strategi Algoritma
Penyelesaian *Cryptarithmic* 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.


```

46 def solving (angka,words):
47     # huruf pada tiap kata
48     letters = []
49     for kata in words:
50         for i in range (len(kata)):
51             letters.append(kata[i])
52
53     letter_set = set()
54     for i in range(len(letters)):
55         letter_set.update(letters[i])
56     letter_list = (list(letter_set))
57     letter_list.sort()
58
59     # pisahin soal sama jawaban
60     count = 0
61     for kata in words:
62         count += 1
63
64     i = count-1
65     j = 0
66     soal = []
67     jawab = []
68     while (i >= 0):
69         if (i == count-1):
70             jawab.insert(j,words[i])
71         elif (i < count-1):
72             soal.insert(j,words[i])
73             j += 1
74             i -= 1
75
76     # huruf pertama tiap kata
77     first_letter = []
78     for kata in words:
79         for i in range (len(kata)):
80             if (i == 0):
81                 first_letter.append(kata[i])
82
83     firstLetter_set = set()
84     for i in range(len(first_letter)):
85         firstLetter_set.update(first_letter[i])
86     firstLetter_list = (list(firstLetter_set))
87     firstLetter_list.sort()
88

```

```

89     percobaan = 0
90     for p in permutasi(angka):
91         solusi = dict(zip(letter_list,p))
92         total_soal = sum(adding_number(word,solusi) for word in soal)
93         total_jawab = sum(adding_number(word,solusi) for word in jawab)
94         nol = isNol(solusi,firstLetter_list)
95         if (total_soal == total_jawab):
96             if (nol == True):
97                 sol = solusi
98             percobaan += 1
99
100     printJawaban(words,sol)
101     print("Jumlah percobaan : ", percobaan)
102
103 def printSoal (words):
104     print('S O A L :')
105     # pisahin soal sama jawaban
106     count = 0
107     for kata in words:
108         count += 1
109
110     i = count-1
111     j = 0
112     soal = []
113     jawab = []
114     while (i >= 0):
115         if (i == count-1):
116             jawab.insert(j,words[i])
117         elif (i < count-1):
118             soal.insert(j,words[i])
119             j += 1
120         i -= 1
121
122     for kata in soal:
123         print(kata)
124     for kata in jawab:
125         separator = len(kata)*'-' + ' ' + ' '
126         print(separator)
127         print(kata)
128     print('\n')
129

```

```

89     percobaan = 0
90     for p in permutasi(angka):
91         solusi = dict(zip(letter_list,p))
92         total_soal = sum(adding_number(word,solusi) for word in soal)
93         total_jawab = sum(adding_number(word,solusi) for word in jawab)
94         nol = isNol(solusi,firstLetter_list)
95         if (total_soal == total_jawab):
96             if (nol == True):
97                 sol = solusi
98             percobaan += 1
99
100     printJawaban(words,sol)
101     print("Jumlah percobaan : ", percobaan)
102
103 def printSoal (words):
104     print('S O A L :')
105     # pisahin soal sama jawaban
106     count = 0
107     for kata in words:
108         count += 1
109
110     i = count-1
111     j = 0
112     soal = []
113     jawab = []
114     while (i >= 0):
115         if (i == count-1):
116             jawab.insert(j,words[i])
117         elif (i < count-1):
118             soal.insert(j,words[i])
119             j += 1
120         i -= 1
121
122     for kata in soal:
123         print(kata)
124     for kata in jawab:
125         separator = len(kata)*'-' + ' ' + ' '
126         print(separator)
127         print(kata)
128     print('\n')
129

```

```

130 def printJawaban (words,solusi):
131     print("J A W A B A N :")
132     # pisahin soal sama jawaban
133     count = 0
134     for kata in words:
135         count += 1
136
137     i = count-1
138     j = 0
139     soal = []
140     jawab = []
141     while (i >= 0):
142         if (i == count-1):
143             jawab.insert(j,words[i])
144         elif (i < count-1):
145             soal.insert(j,words[i])
146             j += 1
147             i -= 1
148
149     for kata in soal:
150         huruf = list(kata)
151         for i in range (len(huruf)):
152             for key in solusi:
153                 if (huruf[i] == key):
154                     print(solusi[key],end='')
155             print('')
156
157     for kata in jawab:
158         huruf = list(kata)
159         separator = len(huruf)*'-' + ' +'
160         print(separator)
161         for i in range (len(huruf)):
162             for key in solusi:
163                 if (huruf[i] == key):
164                     print(solusi[key],end='')
165             print('')
166     print('\n')
167

```

```

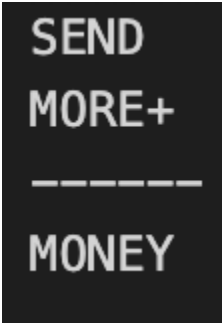
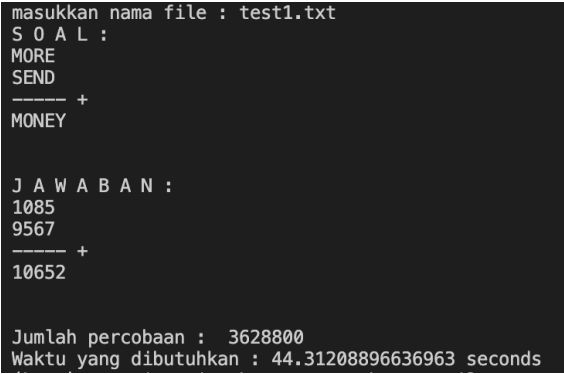
168 # main
169 nama_file = input("masukkan nama file : ")
170 direct = '../test/'
171 words = from_file[direct+nama_file]
172 start_time = time.time()
173 angka = list('0123456789')
174 printSoal(words)
175 solving(angka,words)
176 print("Waktu yang dibutuhkan : %s seconds" % (time.time() - start_time))

```

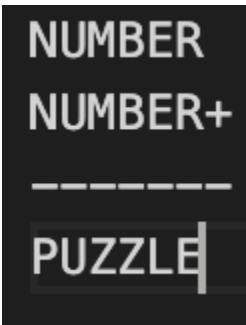
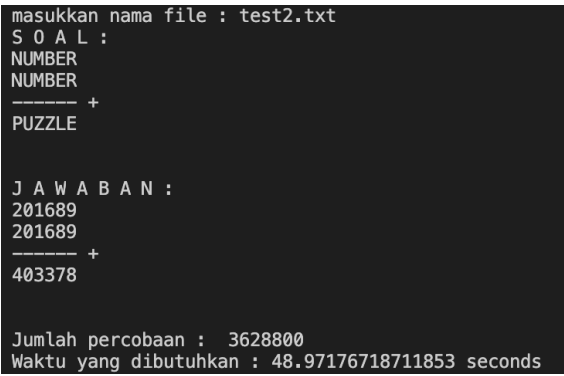
BAB 3

Screenshot Hasil Input Output


1. File : test1.txt

Input	Output
	

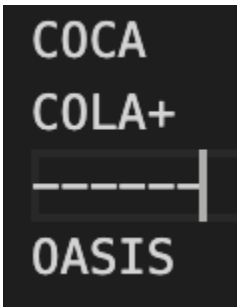
2. File : test2.txt

Input	Output
	

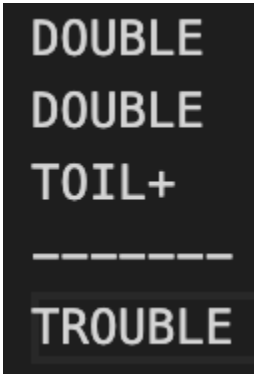
3. File : test3.txt

Input	Output
	<pre> 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 </pre>

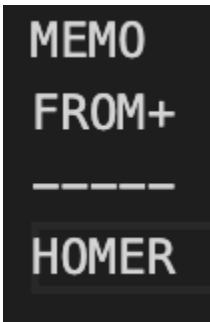
4. File : test4.txt

Input	Output
	<pre> 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 </pre>


5. File : test5.txt

Input	Output
	<pre> 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 </pre>

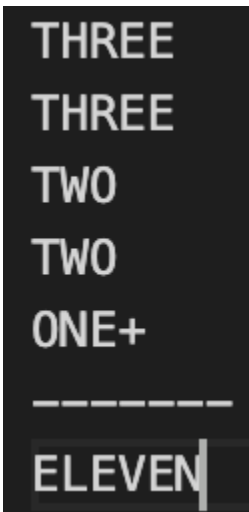
6. File : test6.txt

Input	Output
	<pre> 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 </pre>

7. File : test7.txt

Input	Output
	<pre> masukkan nama file : test7.txt S O A L : ROADS CROSS ----- + DANGER J A W A B A N : 62513 96233 ----- + 158746 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 48.30113101005554 seconds </pre>

8. File : test8.txt

Input	Output
	<pre> masukkan nama file : test8.txt S O A L : ONE TWO TWO THREE THREE ----- + ELEVEN J A W A B A N : 391 803 803 84611 84611 ----- + 171219 Jumlah percobaan : 3628800 Waktu yang dibutuhkan : 71.33085989952087 seconds </pre>

BAB 4

Ketercapaian

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

Drive Link

<https://drive.google.com/drive/folders/1lS2ey1lFIC9lcgg2rD10qnJcwbqMcqYB?usp=sharing>