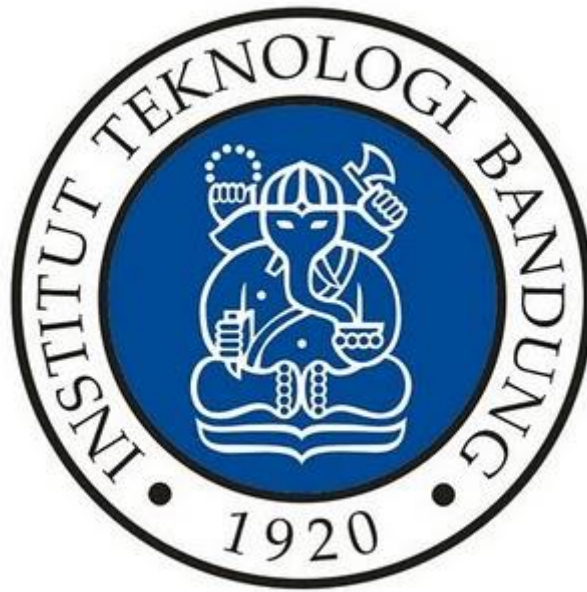


LAPORAN TUGAS KECIL 1

IF2211 – STRATEGI ALGORITMA

Penyelesaian Cryptarithmic dengan Algoritma Brute Force



Oleh:

Alifah Rahmatika Basyasya 13519053

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

2021

Algoritma Brute Force

Deskripsi langkah penyelesaian menggunakan algoritma Brute Force:

1. Program akan membaca sebuah file txt. Pengguna memberikan masukan namafile.txt. File berisi minimal 2 operan diakhiri dengan tanda tambah (+) kemudian diikuti oleh hasil penjumlahan yang dipisahkan oleh garis.
2. Program akan menyimpan operan dan hasil pada sebuah matriks. Inisial dari tiap operan dan hasil (jika belum ada) ditambahkan ke dalam sebuah array. Program juga menyimpan huruf unik pada operan dan hasil pada array 2 dimensi. Baris kedua array ini akan diisi oleh solusi permutasi jika ada. Jika jumlah huruf unik lebih dari 10, program akan menampilkan pesan kesalahan.
3. Selanjutnya, program membentuk permutasi (10, jumlah huruf unik), huruf yang berada dalam array inisial tidak akan disubstitusi dengan 0. Tiap terbentuk satu kombinasi, program akan langsung mengecek kesesuaian dengan penjumlahan di file txt menggunakan konversi huruf ke angka yang sudah didapatkan. Jika total kolom sebuah operan tidak sama dengan hasil, maka program akan melanjutkan pencarian.
4. Jika terdapat kombinasi yang sesuai, hasilnya akan disimpan ke dalam array 2 dimensi yang akan digunakan untuk menampilkan hasil pencarian kepada pengguna. Jumlah pencarian serta waktu yang dibutuhkan juga dicatat dan ditampilkan kepada pengguna.

Source Code Program

```
1  # Nama      : Alifah Rahmatika Basyasya
2  # NIM       : 13519053
3  # Tanggal  : 27 Januari 2021
4  # Deskripsi : Cryptarithmic - Brute Force (Tucil1 Stima)
5
6  import time
7
8  def BacaFile(namafile):
9      #Membaca file dan menyimpannya dalam matriks. Menyimpan inisial, mengecek jumlah huruf
10     #KAMUS LOKAL
11     #i: integer
12     global Maks
13     global Matriks
14     global Inisial
15     global JmlHuruf
16     global ArrayHuruf
17     #ALGORITMA
18     file = open("../test/" + namafile, "r")
19     teks = file.readline().rstrip("\n")
20     while (teks != ""):
21         if (teks[0] != "-" and teks[0] != "+"):
22             idxhuruf = len(teks)-1
23             if (teks.strip(" ")[0] not in Inisial):
24                 Inisial.append(teks.strip(" ")[0])
25                 idx = 9
26                 huruf = [" " for i in range(10)]
27                 for i in range(idxxhuruf, -1, -1):
28                     huruf[idx] = teks[i]
29                     idx -= 1
30                     if (teks[i] not in ArrayHuruf[0]):
31                         if (JmlHuruf < 10):
32                             ArrayHuruf[0].append(teks[i])
33                             JmlHuruf += 1
34                         else:
35                             print("Error. Maksimal terdapat 10 huruf berbeda")
36                             Maks = True
37                 Matriks.append(huruf)
38                 teks = file.readline().strip("+\n")
39     file.close()
40
41 def KonversiKar(Kar, ArrayHuruf, Array):
42     #Mengubah karakter menjadi angka sesuai jawaban
43     #KAMUS LOKAL
44     #i: integer
45     #ALGORITMA
46     i = 0
47     if Kar == "" or Kar == " ":
48         return 0
49     else:
50         i = 0
51         while i < len(ArrayHuruf[0]) and Kar != ArrayHuruf[0][i]:
52             i += 1
53         return Array[i]
54
```

```

54
55 def Cek(Matriks, ArrayHuruf, Inisial, Array):
56     #Mengecek kebenaran jawaban
57     #KAMUS LOKAL
58     #Baris, Total, i, j, c: integer
59     #Benar: boolean
60     #ALGORITMA
61     Baris = len(Matriks)-1
62     Total = 0
63     i = 9
64     Benar = True
65     for c in Inisial:
66         if (KonversiKar(c, ArrayHuruf, Array) == 0):
67             Benar = False
68     while (i >= 0) and (Benar):
69         for j in range(Baris):
70             Total += KonversiKar(Matriks[j][i], ArrayHuruf, Array)
71             if (Total%10 != KonversiKar(Matriks[Baris][i], ArrayHuruf, Array)):
72                 Benar = False
73             else:
74                 Total = Total//10
75                 i -= 1
76     return Benar
77
78 def Permutasi(JmlHuruf, Jawab, Inisial, Matriks):
79     #Mencari kemungkinan kombinasi angka pada jumlah huruf kemudian mengecek kebenaran
80     #KAMUS LOKAL
81     #i, j, idx: integer
82     global Found
83     global Count
84     global ArrayHuruf
85     #ALGORITMA
86     if (not Found):
87         if (JmlHuruf==0):
88             Count += 1
89             if (Cek(Matriks, ArrayHuruf, Inisial, Jawab)):
90                 Found = True
91                 for j in range(len(ArrayHuruf[0])):
92                     ArrayHuruf[1][j] = Jawab[j]
93         else:
94             i = 0
95             while (not Found) and (i < 10):
96                 idx = len(Jawab)-JmlHuruf
97                 if (i not in Jawab) and ((i!=0) or (ArrayHuruf[0][idx] not in Inisial)):
98                     Jawab[idx] = i
99                     Permutasi(JmlHuruf-1, Jawab, Inisial, Matriks)
100                 if (not Found):
101                     i += 1
102                 Jawab[idx] = -1
103
104

```

```

105 def PrintSoal(Matriks, ArrayHuruf):
106     #KAMUS LOKAL
107     #i, j, k: integer
108     #ALGORITMA
109     Baris = len(Matriks)-1
110     for i in range(Baris):
111         print()
112         for j in range(10):
113             if (Matriks[i][j] != ""):
114                 print(Matriks[i][j], end=" ")
115             else:
116                 print(" ", end=" ")
117         print("\n"+10*" ")
118         for k in range(10):
119             if (Matriks[Baris][k] != ""):
120                 print(Matriks[Baris][k], end=" ")
121             else:
122                 print(" ", end=" ")
123         print("\n")
124
125 def PrintJawaban(Matriks, ArrayHuruf):
126     #KAMUS LOKAL
127     #i, j, k: integer
128     #ALGORITMA
129     print("Solusi:")
130     Baris = len(Matriks)-1
131     for i in range(Baris):
132         print()
133         for j in range(10):
134             if (Matriks[i][j] == " ") or (Matriks[i][j] == ""):
135                 print(" ", end=" ")
136             else:
137                 print(KonversiKar(Matriks[i][j], ArrayHuruf, ArrayHuruf[1]), end=" ")
138         print("\n"+10*" ")
139         for k in range(10):
140             if (Matriks[Baris][k] != ""):
141                 print(KonversiKar(Matriks[Baris][k], ArrayHuruf, ArrayHuruf[1]), end=" ")
142             else:
143                 print(" ", end=" ")
144         print("\n")
145

```

```

146 #MAIN PROGRAM
147 #KAMUS
148 #ALGORITMA
149 Maks = False
150 Matriks = []
151 Inisial = []
152 ArrayHuruf = [[],[0 for i in range(10)]]
153 JmlHuruf = 0
154 Count = 0
155 Found = False
156 #Input nama file
157 namafile = input("Masukkan nama file beserta .txt: ")
158
159 #Membaca file
160 BacaFile(namafile)
161
162 if (not Maks):
163     #Menyimpan waktu awal
164     mulai = time.time()
165
166     #Mencari solusi
167     Jawab = [-1 for i in range(JmlHuruf)]
168     Permutasi(JmlHuruf, Jawab, Inisial, Matriks)
169
170     #Menampilkan hasil
171     if (Found):
172         PrintSoal(Matriks, ArrayHuruf)
173         PrintJawaban(Matriks, ArrayHuruf)
174     else:
175         print("Tidak memiliki solusi\n")
176         print("Solusi ditemukan pada percobaan ke-" + str(Count))
177         print("Waktu yang dibutuhkan yaitu", time.time()-mulai)
178

```

Input/Output

Input	Output
<pre> TILES PUZZLES+ ----- PICTURE </pre>	<pre> TILES PUZZLES+ ----- PICTURE Solusi: 91542 3077542+ ----- 3169084 Solusi ditemukan pada percobaan ke-759326 Waktu yang dibutuhkan yaitu 4.801534414291382 </pre>
<pre> NO GUN NO+ ----- HUNT </pre>	<pre> NO GUN NO+ ----- HUNT Solusi: 87 908 87+ ----- 1082 Solusi ditemukan pada percobaan ke-83346 Waktu yang dibutuhkan yaitu 0.35506701469421387 </pre>
<pre> CLOCK TICK TOCK+ ----- PLANET </pre>	<pre> CLOCK TICK TOCK+ ----- PLANET Solusi: 90892 6592 6892+ ----- 104376 Solusi ditemukan pada percobaan ke-843574 Waktu yang dibutuhkan yaitu 7.2596893310546875 </pre>

<pre> THREE THREE TWO TWO ONE+ ----- ELEVEN </pre>	<pre> THREE THREE TWO TWO ONE+ ----- ELEVEN </pre> <p>Solusi:</p> <pre> 84611 84611 803 803 391+ ----- 171219 </pre> <p>Solusi ditemukan pada percobaan ke-176060 Waktu yang dibutuhkan yaitu 1.1289198398590088</p>
<pre> COCA COLA+ ----- OASIS </pre>	<pre> COCA COLA+ ----- OASIS </pre> <p>Solusi:</p> <pre> 8186 8106+ ----- 16292 </pre> <p>Solusi ditemukan pada percobaan ke-82745 Waktu yang dibutuhkan yaitu 0.2598733901977539</p>
<pre> CROSS ROADS+ ----- DANGER </pre>	<pre> CROSS ROADS+ ----- DANGER </pre> <p>Solusi:</p> <pre> 96233 62513+ ----- 158746 </pre> <p>Solusi ditemukan pada percobaan ke-926098 Waktu yang dibutuhkan yaitu 4.900569677352905</p>

<pre> HERE SHE+ ----- COMES </pre>	<pre> HERE SHE+ ----- COMES Solusi: 9454 894+ ----- 10348 Solusi ditemukan pada percobaan ke-204913 Waktu yang dibutuhkan yaitu 0.8411247730255127 </pre>
<pre> MEMO FROM+ ----- HOMER </pre>	<pre> MEMO FROM+ ----- HOMER Solusi: 8485 7358+ ----- 15843 Solusi ditemukan pada percobaan ke-63743 Waktu yang dibutuhkan yaitu 0.2588026523590088 </pre>

Checklist

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

Alamat Kode Program

Link: <https://github.com/AlifahRBasyasya/Tucil1-Stima>