

Tugas Kecil 1 IF2211 Strategi Algoritma

Semester II tahun 2023/2024

Penyelesaian Cyberpunk 2077 Breach Protocol dengan Algoritma Brute Force

Algoritma Brute Force

Program akan mengecek setiap jalur pada matriks dengan menggunakan metode DFS, yaitu melalui kolom atau baris terkecil terlebih dahulu dengan tetap mengikuti aturan pola horizontal dan vertikal bergantian. Pada setiap pengecekan jalur, program akan langsung mengecek nilai poin yang didapatkan pola tersebut dan membandingkannya dengan nilai poin terbesar sebelumnya dan sekaligus membandingkan panjang pola jika memiliki poin yang sama.

Source Code

```
1 import random
2 import time
3
4 def countPoint(arrPos):
5     counter = 0
6     arraySeqBaru = [x[0] for x in arrSeq]
7     currSeq = [0 for x in arraySeqBaru]
8
9     for i in range(len(arrPos)):
10        for j in range(len(currSeq)):
11            if currSeq[j] == "Done":
12                continue
13            if (matriks[arrPos[i][0]][arrPos[i][1]] == arraySeqBaru[j][currSeq[j]]):
14                currSeq[j] += 1
15                if currSeq[j] == len(arraySeqBaru[j]):
16                    currSeq[j] = "Done"
17                    counter += arrSeq[j][1]
18            else:
19                currSeq[j] = 0
20    return counter
21
22 print("1. Read from File \n2. Generate Random by Input \n")
23 pilihan = input("Choose 'Initialize Game' Method: ")
24
25 if (pilihan == '1'):
26     namaFile = input("Masukkan nama file: ")
27     file = open("test/" + namaFile, "r")
28
29     bufSize = int(file.readline())
30     matSize = file.readline().split()
31
32     matriks = []
33     for i in range(int(matSize[1])):
34         row = file.readline().split()
35         matriks.append(row)
36
37     nSeq = int(file.readline())
38     arrSeq = []
39     for i in range(nSeq):
40         seq = file.readline().split()
41         row = int(file.readline())
42         arrSeq.append([seq, row])
43     file.close()
44
45 elif (pilihan == '2'):
46     nToken = input("Jumlah Token Unik: ")
47     listToken = input("Token Unik: ").split()
48
49     bufSize = int(input("Ukuran Buffer: "))
50     matSize = input("Ukuran Matriks: ").split()
51
52     matriks = []
53     for i in range(int(matSize[1])):
54         row = []
55         for j in range(int(matSize[0])):
56             elmt = random.choice(listToken)
57             row.append(elmt)
58             matriks.append(row)
59
60     nSeq = int(input("Jumlah Sekuens: "))
61     lenMaxSeq = int(input("Panjang Maksimal Sekuens: "))
62
63     arrSeq = []
64     for i in range(nSeq):
65         lenSeq = random.randint(2, lenMaxSeq)
66         seq = []
67         while (seq == [] or [seq] in arrSeq): # Prevent same sequence to appear
68             for j in range(lenSeq):
69                 elmt = random.choice(listToken)
70                 seq.append(elmt)
71             arrSeq.append([seq])
72
73     for i in range(nSeq):
74         rew = random.randint(-50, 50)
75         arrSeq[i].append(rew)
76
77     print("\nSekuens: ")
78     for i in arrSeq:
79         print(i)
80     print("\nMatriks: ")
81     for i in matriks:
82         print(i)
83
84 else:
85     print("Invalid Input")
86     exit()
87
88 arrSeq.sort(key=lambda x: x[1], reverse=True)
89
90 awal = time.time()
91
92 maxPoint = 0
93 maxSeq = []
94
95 def bruteforce(arrP):
96     global maxPoint
97     global maxSeq
98     arrPos = [a for a in arrP]
99     for i in range(int(matSize[1])):
100        currCol = arrPos[-1][1]
101        if (len(arrPos) == bufSize):
102            continue
103        elif ([i, currCol] in arrPos):
104            continue
105        else:
106            arrPos.append([i, currCol])
107            poin = countPoint(arrPos)
108
109            if (poin == maxPoint and len(maxSeq) > len(arrPos)):
110                maxSeq = [a for a in arrPos]
111            elif poin > maxPoint:
112                maxPoint = poin
113                maxSeq = [a for a in arrPos]
114
115        for j in range(int(matSize[0])):
116            if (len(arrPos) == bufSize):
117                continue
118            elif ([i, j] in arrPos):
119                continue
120            else:
121                arrPos.append([i, j])
122                poin = countPoint(arrPos)
123
124                if (poin == maxPoint and len(maxSeq) > len(arrPos)):
125                    maxSeq = [a for a in arrPos]
126                elif poin > maxPoint:
127                    maxPoint = poin
128                    maxSeq = [a for a in arrPos]
129
130        bruteforce(arrPos)
131
132    arrPos.pop()
133
134    arrPos.pop()
135
136 for i in range(int(matSize[0])):
137     bruteforce([[0, i]])
138
139 akhir = time.time()
140
141 print("\nOutput: ")
142 strMaxSeq = ""
143 for [x,y] in maxSeq:
144     strMaxSeq += (matriks[x][y] + " ")
145 strLocSeq = ""
146 for [x,y] in maxSeq:
147     strLocSeq += (str(y+1)+", "+str(x+1)+"\n")
148
149 print(maxPoint)
150 print(strMaxSeq)
151 print(strLocSeq)
152
153 print(str(int((akhir - awal)*1000)) + " ms")
154
155 saveState = input("\nApakah ingin menyimpan solusi? (y/n) ")
156 if (saveState == 'y'):
157     filename = input("Masukkan nama file: ")
158     file = open("test/" + filename, "w")
159     file.write(str(maxPoint))
160     file.write('\n')
161     file.write(strMaxSeq)
162     file.write('\n')
163     file.write(strLocSeq)
164     file.close()
```

Contoh Hasil Eksekusi

```
test > ≡ contoh1.txt
1 7
2 6 6
3 7A 55 E9 E9 1C 55
4 55 7A 1C 7A E9 55
5 55 1C 1C 55 E9 BD
6 BD 1C 7A 1C 55 BD
7 BD 55 BD 7A 1C 1C
8 1C 55 55 7A 55 7A
9 3
10 BD E9 1C
11 15
12 BD 7A BD
13 20
14 BD 1C BD 55
15 30
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS D:\Tucil1_13522041> python -u "d:\Tucil1_13522041\game.py" -f contoh1.txt

1. Read from File
2. Generate Random by Input

Choose 'Initialize Game' Method: 1
Masukkan nama file: contoh1.txt

Output:
50
7A BD 7A BD 1C BD 55
1, 1
1, 4
3, 4
3, 5
6, 5
6, 3
1, 3

484 ms

Apakah ingin menyimpan solusi? (y/n) y
Masukkan nama file: solusi1.txt

```
test > ≡ contoh2.txt
1 5
2 5 4
3 55 55 55 55 BD
4 55 7A BD 55 BD
5 1C 1C 7A 7A E9
6 E9 55 E9 55 BD
7 3
8 E9 1C
9 -1
10 7A 7A
11 46
12 1C E9 BD
13 -46
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS D:\Tucil1_13522041> python -u "d:\Tucil1_13522041\game.py" -f contoh2.txt

1. Read from File
2. Generate Random by Input

Choose 'Initialize Game' Method: 1
Masukkan nama file: contoh2.txt

Output:
46
55 7A 7A
3, 1
3, 3
4, 3

0 ms

Apakah ingin menyimpan solusi? (y/n) y
Masukkan nama file: solusi2.txt

```
1. Read from File
2. Generate Random by Input

Choose 'Initialize Game' Method: 2
Jumlah Token Unik: 5
Token Unik: 7A 55 E9 1C BD
Ukuran Buffer: 5
Ukuran Matriks: 5 5
Jumlah Sekuens: 5
Panjang Maksimal Sekuens: 4

Sekuens:
[['55', '7A', '55', '55'], -10]
[['BD', '55', '55', 'E9'], 37]
[['E9', '1C'], -40]
[['55', 'E9'], -33]
[['55', 'BD'], -4]

Matriks:
[['1C', 'BD', '7A', '55', 'E9']
[['1C', 'BD', 'BD', '7A', 'BD']
[['55', '1C', '1C', '7A', '55']
[['1C', '55', '1C', 'E9', 'BD']
[['E9', 'BD', '7A', 'BD', '55']]

Output:
0

0 ms

Apakah ingin menyimpan solusi? (y/n) y
Masukkan nama file: solusi5.txt
```

```
test > ≡ contoh3.txt
1 8
2 6 5
3 E9 7A 1C 7A E9 BD
4 55 1C BD 55 E9 BD
5 BD 1C 1C 1C BD 7A
6 7A 7A 55 7A BD 1C
7 BD E9 BD 1C BD E9
8 4
9 7A 1C E9
10 5
11 55 55 55
12 10
13 7A 7A 7A E9
14 14
15 7A 1C 1C
16 7
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1. Read from File
2. Generate Random by Input

Choose 'Initialize Game' Method: 1
Masukkan nama file: contoh3.txt

Output:
21
7A 7A 7A E9 7A 1C 1C
2, 1
2, 4
1, 4
1, 1
4, 1
4, 3
2, 3

1092 ms

Apakah ingin menyimpan solusi? (y/n) y
Masukkan nama file: solusi3.txt

```
1. Read from File
2. Generate Random by Input

Choose 'Initialize Game' Method: 2
Jumlah Token Unik: 5
Token Unik: 7A 55 E9 1C BD
Ukuran Buffer: 8
Ukuran Matriks: 6 6
Jumlah Sekuens: 4
Panjang Maksimal Sekuens: 5

Sekuens:
[['55', 'E9', 'E9', 'E9', 'E9'], 2]
[['1C', 'E9', 'E9'], 26]
[['E9', '55'], -6]
[['1C', 'BD'], -28]

Matriks:
[['7A', 'BD', '7A', 'E9', '55', '7A']
[['E9', '55', '55', 'E9', 'E9', 'E9']
[['1C', '55', '1C', '7A', '7A', 'BD']
[['7A', '1C', 'BD', 'BD', 'BD', 'BD']
[['1C', '55', '1C', '7A', 'BD', 'BD']
[['BD', '1C', 'E9', '55', '1C', 'E9']]

Output:
26
BD 1C E9 E9
2, 1
2, 6
6, 6
6, 2

2702 ms

Apakah ingin menyimpan solusi? (y/n) y
Masukkan nama file: solusi4.txt
```

```
1. Read from File
2. Generate Random by Input

Choose 'Initialize Game' Method: 2
Jumlah Token Unik: 7
Token Unik: 7A 55 E9 1C BD AK 67
Ukuran Buffer: 6
Ukuran Matriks: 5 6
Jumlah Sekuens: 4
Panjang Maksimal Sekuens: 4

Sekuens:
[['AK', 'AK', '67'], 26]
[['BD', '7A', '67'], -40]
[['55', '67', 'BD', '67'], -41]
[['1C', '67', 'AK'], -12]

Matriks:
[['7A', 'BD', '55', '55', '7A']
[['1C', 'E9', '67', 'AK', 'AK']
[['BD', '7A', 'E9', 'BD', '1C']
[['67', '55', 'AK', 'BD', '1C']
[['7A', '1C', 'E9', '7A', 'BD']
[['1C', '7A', '55', 'BD', '67']]

Output:
26
55 AK AK 67
4, 1
4, 2
5, 2
5, 6

70 ms

Apakah ingin menyimpan solusi? (y/n) y
Masukkan nama file: solusi6.txt
```

Repository

https://github.com/Bana-man/Tucil1_13522041

Lampiran

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa kesalahan	✓	
2. Program berhasil dijalankan	✓	
3. Program dapat membaca masukan berkas .txt	✓	
4. Program dapat menghasilkan masukan secara acak	✓	
5. Solusi yang diberikan program optimal	✓	
6. Program dapat menyimpan solusi dalam berkas .txt	✓	
7. Program memiliki GUI		✓

AHMAD HASAN ALBANA (13522041)