

Laporan Tugas Kecil 1 IF2211 Strategi Algoritma

Semester II Tahun 2023/2024

Penyelesaian ‘Cyberpunk 2077 Breach Protocol’ dengan Algoritma Brute Force

Disusun oleh:

13522127 - Maulana Muhamad Susetyo

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

2023

1. Algoritma

1.1. Algoritma Input Random

Menggunakan numpy untuk menghasilkan nilai random pada matrix, panjang sekuens, dan nilai masing-masing sekuens.

1.2. Input/Output File

Membaca per baris input untuk program. Menampilkan hasil program dan opsi untuk menyimpan hasil pada file.

1.3.

2. Source Code

2.1. kode Input

```
def inputFile():
    filename=str(input("Masukkan Nama File (Tanpa Extension): "))
    current_directory = os.path.dirname(__file__)
    file_path = os.path.join(current_directory, '..', 'test', filename+'.txt')
    print(file_path)
    if os.path.exists(file_path):
        with open(file_path, 'r') as file:
            Buffer = int(file.readline().strip())
            m, n = map(int, file.readline().strip().split())
            Matrix_File = []
            for _ in range(m):
                row = list(map(str, file.readline().strip().split()))
                Matrix_File.append(row)
            Sum_CQ = int(file.readline().strip())
            Arr_CQ = []
            Arr_CQ_Val = []
            for _ in range(Sum_CQ):
                Arr_CQ.append(file.readline().strip())
                Arr_CQ_Val.append(int(file.readline().strip()))
    else:
        print("File Tidak Ditemukan. ")
        inputFile()
```

```

def inputOto():
    NToken = int(input("Masukkan Jumlah Token Unik: "))
    lenTemp = 0
    while (lenTemp != NToken):
        print("Masukkan Token: ")
        Arr_Token = list(map(str, input().split()))
        SetToken = set(Arr_Token)
        if len(SetToken) == len(Arr_Token):
            lenTemp = len(Arr_Token)
            if lenTemp != NToken:
                print("Jumlah token tidak sesuai. ")
        else:
            print("Token harus unik. ")
    Buffer = int(input("Masukkan Ukuran Buffer: "))
    lenTemp = 0
    while (lenTemp != 2):
        print("Masukkan Ukuran Matrix: ")
        DimensiMat = list(map(int, input().split()))
        lenTemp=len(DimensiMat)
    SumCQ = 0
    while SumCQ<1:
        SumCQ = int(input("Masukkan Jumlah Sekuens: "))
        if SumCQ<1:
            print("Jumlah Harus Lebih Besar Daripada Nol. ")
    MaxCQ = 0
    while MaxCQ<1:
        MaxCQ = int(input("Masukkan Ukuran Maksimal Sekuens: "))
        if MaxCQ<1:
            print("Ukuran Harus Lebih Besar Daripada Nol. ")
    Arr_CQ = np.random.randint(1,MaxCQ+1, size=(SumCQ))
    Arr_CQ.sort()
    Arr_CQ_Val = np.random.randint(1,100, size=(SumCQ))
    Arr_CQ_Val.sort()
    Arr_CQ = [""] for x in range(SumCQ)
    for i in range(SumCQ):
        Arr_CQ[i]=CQ_Rand(Arr_CQ[i],Arr_Token)
    Matrix_Hasil = Matrix_Rand(DimensiMat[0],DimensiMat[1],Arr_Token)
    DisplayMatrix(Matrix_Hasil,DimensiMat[0],DimensiMat[1])

```

2.2. Kode Generasi Sekuens dan Matrix Acak

```

def Matrix_Rand(m,n,Arr):
    temp = np.random.randint(len(Arr), size=(m,n))
    res = np.empty((m,n), dtype='<U4')
    for i in range (0,m):
        for j in range (0,n):
            res[i][j]=Arr[temp[i][j]]
    return res

def CQ_Rand(n,Arr):
    temp = np.random.randint(len(Arr), size=(n))
    res = Arr[temp[0]]
    for i in range (1,n):
        res+=' '
        res+=Arr[temp[i]]
    return res

```

2.3.

3. Tangkapan Layar

4. Pranala dan Lampiran

4.1. Pranala ke Repository Github

https://github.com/LastPrism7/Tucil1_13522127

4.2. 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		✓