

Tugas Kecil 3 IF2211 Strategi Algoritma

Implementasi Algoritma A* untuk Menentukan Lintasan Terpendek



Oleh:

La Ode Rajuh Emoko	13519170
Akeyla Pradia Naufal	13519178

**PROGRAM STUDI MAGISTER INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2021**

1. Deskripsi Aplikasi

Aplikasi ini dapat menerima input file berisi koordinat dan data jarak dalam bentuk matriks ketetanggaan dari beberapa simpul dalam sebuah peta, kemudian menerima input simpul awal dan simpul akhirnya, kemudian menampilkan jarak terpendek serta jalur yang dilewatinya. Format file adalah baris pertama berisi banyaknya titik yang akan dianalisis, misalnya N. N baris selanjutnya berisi data koordinat x dan y dari tiap titik serta nama dari titiknya dipisahkan dengan spasi. Kemudian N baris selanjutnya adalah matriks ketetanggaan yang berisi jarak sebenarnya(bobot) antara setiap simpul dengan setiap simpul lainnya.

Saat aplikasi dijalankan, program akan menampilkan daftar simpul beserta koordinatnya pada peta. Kemudian aplikasi akan meminta indeks dari simpul asal dan simpul tujuan. Setelah simpul awal dan simpul akhir telah diterima, program akan menampilkan jarak terpendek serta jalur yang dilewatinya untuk bergerak dari simpul asal ke simpul tujuan

2. Algoritma A*

Algoritma A* yang digunakan menggunakan acuan $f(n) = g(n) + h(n)$ dengan $g(n)$ adalah jarak sebenarnya dari simpul awal ke simpul n, dan $h(n)$ adalah jarak garis lurus dari simpul n ke simpul akhir.

Algoritma yang digunakan terdiri dari beberapa langkah. Yang pertama, menghitung dan menyimpan jarak garis lurus dari semua simpul ke simpul tujuan. Jarak dihitung menggunakan euclidean distance menggunakan koordinat dari setiap simpulnya.

Langkah kedua adalah membuat priority queue kosong yang akan diisi dengan informasi simpul yang aktif (jalur untuk sampai ke simpul tersebut), dengan bobot merupakan jarak asli dari matriks ketetanggaan dari simpul awal ke simpul tersebut ditambah dengan jarak lurus dari simpul tersebut ke simpul tujuan. Priority queue ini pertama kali diisi oleh simpul awal dengan bobot jarak garis lurus dari simpul awal ke simpul tujuan.

Langkah ketiga adalah mengambil simpul dengan bobot terkecil (dequeue/matikan simpulnya) dari priority queue. Jika simpul dengan bobot terkecil ini adalah simpul tujuan, maka pencarian selesai. Jika bukan simpul tujuan, maka akan lanjut ke langkah selanjutnya.

Langkah keempat adalah ekspansi simpul yang sebelumnya dimatikan(parent). Ekspansi dilakukan dengan mengambil semua simpul(child) yang terhubung dengan simpul yang dimatikan tersebut (jarak tidak sama dengan nol), kemudian ditambahkan child kedalam jalur yang disimpan pada simpul sebelumnya sehingga terbentuk jalur dari simpul awal ke simpul anak yang baru. Untuk jaraknya, dihitung dengan cara jarak pada simpul parent, dikurangi jarak lurus dari tujuan ke parent, ditambah dengan jarak

sebenarnya dari parent ke child, ditambah dengan jarak garis lurus dari child ke simpul tujuan. Jarak dan jalur baru yang terbentuk akan dimasukkan ke priority queue.

Langkah kelima adalah jika pada akhirnya priority queue kosong dan tidak bisa diekspansi lagi, maka artinya tidak ada jalur dari simpul awal ke simpul tujuan. Jika priority queue belum kosong atau belum ketemu jalur terpendeknya, ulangi lagi setiap langkah di atas sampai keluar hasilnya.

3. Kode Program

```
from queue import PriorityQueue

#fungsi fungsi
def jarak(a, b): # a dan b adalah coordinate (absis, ordinat, nama simpul)
    return ((a[0]-b[0])**2 + (a[1] - b[1])**2)**0.5

#####

# Baca file
print("Silahkan masukkan nama file: ")
f = input()
fFile = open(f, "r")
N = int(fFile.readline())
dictPlace = {}
for _ in range(N):
    x, y, place = fFile.readline().split(" ")
    dictPlace[_] = [float(x), float(y), place[:-1]]

adjMat = []
for line in fFile.readlines():
    adjMat.append([float(x) for x in line.split(" ")])
#print(adjMat)

#menampilkan indeks, koordinat dari tiap simpul
print("indeks, data simpul (x, y, nama simpul)")
for i in dictPlace:
    print(i, ": ", dictPlace[i])
```

```
#####

coordinate = dictPlace #gantinama doang
matriksKetetanggaan = adjMat #gantinama doang
# masukkan indeks kota asal dan tujuan(diasumsikan input sesuai, tidak ada
yang diluar indeks yang diberikan)
asal = int(input("masukkan indeks tempat asal: "))
tujuan = int(input("masukkan indeks tempat tujuan: "))

#hitung jarak garis lurus dari setiap kota ke kota tujuan dan simpan di
sebuah array
jarakLurusSetiapTempatKeTujuan = [jarak(coordinate[i], coordinate[tujuan])
for i in range (len(coordinate))]

#####
#Algoritma A*

# Step 1, bikin simpul terbuka dan tertutup dengan awal simpul buka adalah
start node
simpulBuka = PriorityQueue() #priorityQueue yang prioritasnya adalah jarak
terkecil terlebih dahulu dengan nilai berupa jalur yang dilewati
simpulBuka.put((jarakLurusSetiapTempatKeTujuan[asal], [asal]))

stop = False
jarak = 0
while (not stop):
    if (simpulBuka.empty()): # Step 2, Kalau simpul bukanya kosong, maka
tidak ada jalan dari asal ke tujuan.
        stop = True
        print("tidak ada jalur yang menghubungkan asal dan tujuan")

    #step 3, ambil node yang paling kecil, masukin ke simpul Tutup
    else:
        t = simpulBuka.get()
        #print("Proses: ", t) #jika ingin melihat prosesnya
```

```

        #step 4, jika ujung jalur merupakan simpul tujuan, maka selesai.
ambil jarak dan jalurnya
        if(t[1][len(t[1])-1]== tujuan):
            stop = True
            jarak = t[0]
            jalur = t[1]

        #step 5, jika ujung jalur bukan tujuan,
        else: #masukkan setiap anak dari simpul yang nilai f nya terkecil
ke open list

            ujungjalur = t[1][len(t[1]) - 1]
            #copy jalur
            for i in range(0, len(coordinate)):
                a = []
                for j in range(0, len(t[1])):
                    a.append (t[1][j])

                    if (matriksKetetanggaan[i][ujungjalur] != 0): #merupakan
anak jika mempunyai jarak dengan ujung jalur
                        a.append(i)
                        nilai = t[0] -
jarakLurusSetiapTempatKeTujuan[ujungjalur] +
matriksKetetanggaan[i][ujungjalur] + jarakLurusSetiapTempatKeTujuan[i]
                        simpulBuka.put((nilai, a)) # step 6, ekspansi simpul
terbuka

print("\nJarak terpendek adalah: ", jarak)
print('')
print("jalurnya adalah:")

for i in jalur:
    print (coordinate[i][2])

```

4. Peta/Graf Input

Test Case 1: Daerah ITB - Dago (nama file: itbdago.txt)

9

-6.887345550894515 107.612748902012 NoahBarn

-6.887393217822871 107.61352163865868 SimpangDagoDayangSumbi

-6.893760289427523 107.61294369131426 SimpangDagoGanesa
-6.893216164816723 107.610455739738 DepanKubus
-6.892846698254164 107.61041885936436 ATMBNI
-6.88788177098051 107.60829142766119 TamanSari
-6.8938797570520105 107.60845924965025 TamanSariGanesa
-6.894784108646859 107.61016541549583 AngkringanNarji
-6.89487943430489 107.60883971701267 BebekStallone
0 86 0 0 0 500 0 0 0
86 0 700 0 0 0 0 0 0
0 700 0 290 0 0 0 0 0
0 0 290 0 42 0 240 210 0
0 0 0 42 0 0 0 0 0
500 0 0 0 0 0 700 0 0
0 0 0 240 0 700 0 0 120
0 0 0 210 0 0 0 0 150
0 0 0 0 0 0 120 150 0

Test Case 2: Daerah Alun-alun Kota Bandung (nama file: alun-alun.txt)

8

-6.921253791397621 107.60768308791431 MonumenAsiaAfrika
-6.922567188312322 107.60752572266021 AlunAlunTimur-DalemKaum
-6.9223810924048985 107.60645954255628 DepanPlazaParahyangan
-6.923395154567201 107.6063127928438 DewiSartikaKepatihan
-6.923079913787076 107.60394075833749 OtistaKepatihan
-6.920816040287142 107.60409065098258 AsiaAfrikaOtista
-6.922084631440316 107.60402289006068 DalemKaumCibadak
-6.921683745469032 107.60136441774301 CibadakNightCulinaryFestival
0 150 0 0 0 400 0 0
150 0 120 0 0 0 0 0
0 120 0 120 0 0 280 0
0 0 120 0 270 0 0 0
0 0 0 270 0 250 110 0
400 0 0 0 250 0 140 0
0 0 280 0 110 140 0 300
0 0 0 0 0 0 300 0

Test Case 3: Daerah Buah Batu (nama file: buahbatu.txt)

8

-6.966213363258414 107.63525227051032 Palem1Residence
-6.965100780603841 107.63572753192938 AdhyaksaRayaBatununggal
-6.965393244325823 107.63795716166906 AdhyaksaRayaBojongsoang
-6.966659628093693 107.63770536552552 DominosPizza

-6.968390897444043 107.63728005181532 SukapuraBojongsoang
-6.967826127641298 107.63458208946096 SukapuraAdhyaksaRaya
-6.966251984378363 107.63480375879166 TikunganAdhyaksaRaya
-6.968669140658813 107.63732743065503 MartabakJayaraga
0 140 0 0 0 0 51 0
140 0 250 0 0 0 0 0
0 250 0 150 0 0 0 0
0 0 150 0 200 0 0 0
0 0 0 200 0 300 0 23
0 0 0 0 300 0 180 0
51 0 0 0 0 180 0 0
0 0 0 0 23 0 0 0

Test Case 4: Peta jalan kawasan rumah Rajuh, Bau-bau. (nama file: jalanbaubau.txt)

8

-5.399788231252937 122.6685447340743 SDN1Liabuku
-5.460534609987479 122.61791641392628 SimpangRSSiloam
-5.465474153809062 122.61842008216698 SimpangJambuMenteGatotSubroto
-5.467533402967176 122.60645535412243 JembatanBeli
-5.4603787386434925 122.61653862563644 SimpangHasanuddinDipenogoro
-5.480560440708423 122.60219249488718 MuseumKebudayaanWolio
-5.3734681218780995 122.68721279451182 KolamRenangBanyuBiru
-5.471433337299639 122.59813447992826 SupermarketDuaSekawan
0 10040 0 0 0 0 3900 0
10040 0 650 0 160 0 0 0
0 650 0 1500 0 0 0 0
0 0 1500 0 1600 2000 0 1200
0 160 0 1600 0 0 0 0
0 0 0 2000 0 0 0 1700
3900 0 0 0 0 0 0 0
0 0 0 1200 0 1700 0 0

Test Case 5: Peta jalan kawasan rumah Akeyla, Kota Medan (nama file: jalanmedan.txt)

8

3.5753047476884205 98.68408520894492 IstanaMaimun
3.5760463838063323 98.68763140498996 SimpangRaya
3.5686611479145123 98.6911133137037 SimpangMakamPahlawan
3.5724637915810806 98.6856217465715 AceHardware
3.572484305481788 98.68931624414436 SimpangHalatJuanda
3.568636698402871 98.69573078861713 UpnormalGedungArca
3.565142791232093 98.69231080441259 RamayanaTeladan

3.5649484740282342 98.69558421533549 StadionTeladan
0 500 0 400 0 0 0 0
500 0 900 0 450 0 1300 0
0 900 0 0 450 500 400 0
400 0 0 0 400 0 0 0
0 450 450 400 0 900 0 0
0 0 500 0 900 0 0 500
0 1300 400 0 0 0 0 400
0 0 0 0 0 500 400 0

Test Case 6: Peta jalan kawasan Kota Semarang (nama file: jalansemarang.txt)
9

-7.060403696122771 110.36164489781254 AsramaPutraSMASemesta
-6.983885183610703 110.41043781229105 Lawang1000
-7.011720712671395 110.37925988175601 UntungSuropatiAbdurahmanSaleh
-6.984793737689247 110.38384721652506 BundaranDepanEksBandara
-6.99486151255203 110.39941888337987 SamPooKong
-6.997038778109881 110.38000880890579 Suratmo
-7.00295091525412 110.39240323829273 SimpangKimiaFarma
-6.994457914142763 110.4068153053267 RSUPKariadi
-6.988940018852768 110.42390169493245 MallCL
0 0 7100 0 0 0 0 0
0 0 0 3200 3100 0 0 1300 1900
7100 0 0 0 0 2100 1900 0 0
0 3200 0 0 3400 2000 0 0 0
0 3100 0 3400 0 0 1500 950 0
0 0 2100 2000 0 0 1600 0 0
0 0 1900 0 1500 1600 0 0 0
0 1300 0 0 950 0 0 0 2700
0 1900 0 0 0 0 0 2700 0

5. Hasil

Test case 1: Kawasan ITB Dago


```

Silahkan masukkan nama file:
../test/itbdago.txt
indeks, data simpul (x, y, nama simpul)
0 : [-6.887345550894515, 107.612748902012, 'NoahBarn']
1 : [-6.887393217822871, 107.61352163865868, 'SimpangDagoDayangSumbi']
2 : [-6.893760289427523, 107.61294369131426, 'SimpangDagoGanesa']
3 : [-6.893216164816723, 107.610455739738, 'DepanKubus']
4 : [-6.892846698254164, 107.61041885936436, 'ATMBNI']
5 : [-6.88788177098051, 107.60829142766119, 'TamanSari']
6 : [-6.8938797570520105, 107.60845924965025, 'TamanSariGanesa']
7 : [-6.894784108646859, 107.61016541549583, 'AngkringanNarji']
8 : [-6.89487943430489, 107.60883971701267, 'BebekStallone']
masukkan indeks tempat asal: 0
masukkan indeks tempat tujuan: 8

Jarak terpendek adalah: 1320.0 meter

jalurnya adalah:
NoahBarn
TamanSari
TamanSariGanesa
BebekStallone

```

Test case 2: Kawasan Alun Alun Bandung

```

Silahkan masukkan nama file:
../test/alunalun.txt
indeks, data simpul (x, y, nama simpul)
0 : [-6.921253791397621, 107.60768308791431, 'MonumenAsiaAfrika']
1 : [-6.922567188312322, 107.60752572266021, 'AlunAlunTimur-DalemKaum']
2 : [-6.9223810924048985, 107.60645954255628, 'DepanPlazaParahyangan']
3 : [-6.923395154567201, 107.6063127928438, 'DewiSartikaKepatihan']
4 : [-6.923079913787076, 107.60394075833749, 'OtistaKepatihan']
5 : [-6.920816040287142, 107.60409065098258, 'AsiaAfrikaOtista']
6 : [-6.922084631440316, 107.60402289006068, 'DalemKaumCibadak']
7 : [-6.921683745469032, 107.60136441774301, 'CibadakNightCulinaryFestival']
masukkan indeks tempat asal: 0
masukkan indeks tempat tujuan: 7

Jarak terpendek adalah: 840.0 meter

jalurnya adalah:
MonumenAsiaAfrika
AsiaAfrikaOtista
DalemKaumCibadak
CibadakNightCulinaryFestival

```

Test case 3: Kawasan Buah Batu

```

Silahkan masukkan nama file:
../test/buahbatu.txt
indeks, data simpul (x, y, nama simpul)
0 : [-6.966213363258414, 107.63525227051032, 'Palem1Residence']
1 : [-6.965100780603841, 107.63572753192938, 'AdhyaksaRayaBatununggal']
2 : [-6.965393244325823, 107.63795716166906, 'AdhyaksaRayaBojongsoang']
3 : [-6.96659628093693, 107.63770536552552, 'DominosPizza']
4 : [-6.968390897444043, 107.63728005181532, 'SukapuraBojongsoang']
5 : [-6.967826127641298, 107.63458208946096, 'SukapuraAdhyaksaRaya']
6 : [-6.966251984378363, 107.63480375879166, 'TikunganAdhyaksaRaya']
7 : [-6.968669140658813, 107.63732743065503, 'MartabakJayaraga']
masukkan indeks tempat asal: 07
masukkan indeks tempat tujuan: 0

Jarak terpendek adalah: 554.0 meter

jalurnya adalah:
MartabakJayaraga
SukapuraBojongsoang
SukapuraAdhyaksaRaya
TikunganAdhyaksaRaya
Palem1Residence

```

Test case 4: Jalan di Baubau

```

Silahkan masukkan nama file:
../test/jalanbaubau.txt
indeks, data simpul (x, y, nama simpul)
0 : [-5.399788231252937, 122.6685447340743, 'SDN1Liabuku']
1 : [-5.460534609987479, 122.61791641392628, 'SimpangRSSiloam']
2 : [-5.465474153809062, 122.61842008216698, 'SimpangJambuMenteGatotSubroto']
3 : [-5.467533402967176, 122.60645535412243, 'JembatanBeli']
4 : [-5.4603787386434925, 122.61653862563644, 'SimpangHasanuddinDipenogoro']
5 : [-5.480560440708423, 122.60219249488718, 'MuseumKebudayaanWolio']
6 : [-5.3734681218780995, 122.68721279451182, 'KolamRenangBanyuBiru']
7 : [-5.471433337299639, 122.59813447992826, 'SupermarketDuaSekawan']
masukkan indeks tempat asal: 0
masukkan indeks tempat tujuan: 5

Jarak terpendek adalah: 13800.0 meter

jalurnya adalah:
SDN1Liabuku
SimpangRSSiloam
SimpangHasanuddinDipenogoro
JembatanBeli
MuseumKebudayaanWolio

```

Test case 5: Jalan di Medan

```

Silahkan masukkan nama file:
../test/jalanmedan.txt
indeks, data simpul (x, y, nama simpul)
0 : [3.5753047476884205, 98.68408520894492, 'IstanaMaimun']
1 : [3.5760463838063323, 98.68763140498996, 'SimpangRaya']
2 : [3.5686611479145123, 98.6911133137037, 'SimpangMakamPahlawan']
3 : [3.5724637915810806, 98.6856217465715, 'AceHardware']
4 : [3.572484305481788, 98.68931624414436, 'SimpangHalatJuanda']
5 : [3.568636698402871, 98.69573078861713, 'UpnormalGedungArca']
6 : [3.565142791232093, 98.69231080441259, 'RamayanaTeladan']
7 : [3.5649484740282342, 98.69558421533549, 'StadionTeladan']
masukkan indeks tempat asal: 0
masukkan indeks tempat tujuan: 7

Jarak terpendek adalah: 2050.0 meter

jalurnya adalah:
IstanaMaimun
AceHardware
SimpangHalatJuanda
SimpangMakamPahlawan
RamayanaTeladan
StadionTeladan

```

Test case 6: Jalan di Semarang

```

Silahkan masukkan nama file:
../test/jalansemarang.txt
indeks, data simpul (x, y, nama simpul)
0 : [-7.060403696122771, 110.36164489781254, 'AsramaPutraSMASemesta']
1 : [-6.983885183610703, 110.41043781229105, 'Lawang1000']
2 : [-7.011720712671395, 110.37925988175601, 'UntungSuropatiAbdurahmanSaleh']
3 : [-6.984793737689247, 110.38384721652506, 'BundaranDepanEksBandara']
4 : [-6.99486151255203, 110.39941888337987, 'SamPookong']
5 : [-6.997038778109881, 110.38000880890579, 'Suratmo']
6 : [-7.00295091525412, 110.39240323829273, 'SimpangKimiaFarma']
7 : [-6.994457914142763, 110.4068153053267, 'RSUPKariadi']
8 : [-6.988940018852768, 110.42390169493245, 'MallCL']
masukkan indeks tempat asal: 0
masukkan indeks tempat tujuan: 8

Jarak terpendek adalah: 14150.0 meter

jalurnya adalah:
AsramaPutraSMASemesta
UntungSuropatiAbdurahmanSaleh
SimpangKimiaFarma
SamPookong
RSUPKariadi
MallCL

```

Alamat Kode Program dan Check List

<https://github.com/anakpindahan/Tucil3OOP>

1	Program dapat menerima input graf	V
2	Program dapat menghitung lintasan terpendek	V
3	Program dapat menampilkan lintasan terpendek serta jaraknya	V
4	Bonus: Program dapat menerima input peta dengan Google Map API dan dapat menampilkan peta	X