

Penerapan Algoritma Dijkstra Dalam Pencarian Toilet Terdekat di GKU Barat ITB

M. Hanief Fatkhan Nashrullah - 13522100

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13522100@std.stei.itb.ac.id

Abstract—GKU Barat ITB merupakan sebuah gedung yang berada di ITB yang umumnya digunakan untuk pelaksanaan perkuliahan mata kuliah umum. Gedung kuliah ini memiliki banyak ruangan dan juga lebih dari satu tangga untuk menuju satu ruangan. Namun, dengan banyaknya tangga dan posisi yang sulit dibedakan dengan posisi lain pada gedung yang sama, akses ruangan menjadi membingungkan, terutama pada pencarian toilet. Kondisi ini tentu dapat menyulitkan pengunjung gedung ini. Maka dari itu, diperlukan suatu metode yang mangkus untuk menentukan toilet terdekat dan jalurnya. Salah satu metode tersebut adalah dengan menggunakan Algoritma Dijkstra.

Keywords—Algoritma Dijkstra, Pencarian, Toilet, GKUB.

I. PENDAHULUAN

Gedung Kuliah Umum Barat ITB (GKUB ITB), sebuah gedung berbentuk oktagon yang berada di dekat Jl. III ITB. Gedung ini memiliki tiga lantai dengan 23 ruang kuliah. Gedung ini umumnya digunakan untuk pelaksanaan mata kuliah umum dan mata kuliah TPB. Rancangan GKUB diprakarsai oleh Ir. Soekarno dan mulai dibangun pada tahun 1984, kemudian mulai digunakan pada tahun 1986.

GKUB memiliki 4 ruang kuliah di lantai 1, 11 ruang kuliah di lantai 2, 8 ruang kuliah di lantai 3, dan beberapa ruang lain seperti toilet, kantin, dan mushola. Ruangan yang berada di lantai 1 dapat diakses dengan cukup mudah karena kondisi atau lingkungan sekitar yang jelas dan mudah dibedakan. Berbeda dengan lantai 1, akses menuju ruangan spesifik yang berada di lantai 2 dan lantai 3 cukup rumit karena karakteristik dari jalur menuju ruangan tersebut secara visual tampak sama. Kondisi ini dapat menyebabkan kesulitan dalam pencarian ruangan, terutama untuk mencari toilet. Tidak hanya karena secara visual area GKUB tampak sama, tetapi papan penanda menuju toilet yang umumnya hanya ada di dekat toilet itu sendiri menambah tingkat kesulitan dalam pencarian toilet di GKUB.

Makalah ini bertujuan untuk menerapkan Algoritma Dijkstra untuk mencari rute terdekat untuk mencari toilet di area GKUB. Dengan memanfaatkan Algoritma Dijkstra, makalah ini diharapkan dapat mengurangi waktu pencarian dan juga mengurangi berputar ataupun naik-turun tangga di area GKUB untuk mencari toilet.



Gambar 1 Area Tengah GKUB ITB
sumber : dokumentasi penulis

II. TEORI DASAR

A. Definisi Graf

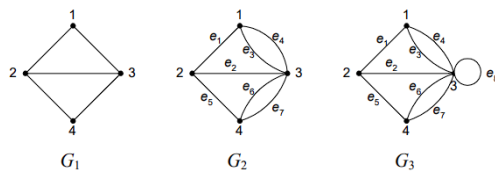
Graf merupakan himpunan tidak kosong dari simpul-simpul (vertices) dan himpunan sisi (edges) yang menghubungkan satu pasang simpul. Graf digunakan untuk merepresentasikan objek-objek dan hubungan antara objek-objek tersebut. Graf dapat dimanfaatkan untuk representasi jaringan sosial, sistem transportasi, routing jaringan komputer, dan sebagainya.

Berikut adalah definisi graf dalam notasi matematika

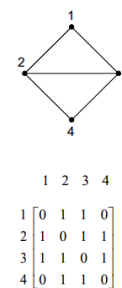
$G = (V, E)$, dengan V dan E :

V = himpunan tidak-kosong dari simpul-simpul
 $= \{v_1, v_2, \dots, v_n\}$

E = himpunan sisi yang menghubungkan sepasang simpul
 $= \{e_1, e_2, \dots, e_n\}$



Gambar 2 Contoh graf,
(a) graf sederhana, (b) graf ganda, dan (c) graf semu
sumber : [Graf \(bag.1\) - IF2120 - Rinaldi Munir](#)



Gambar 4 Representasi Graf dalam Matriks Ketetanggaan
sumber : [Graf \(bag.2\) - IF2120 - Rinaldi Munir](#)

B. Ketetanggaan Simpul dan Sisi Berurutan

Simpul yang bertetangga merupakan dua buah simpul yang terhubung satu sama lain secara langsung.

Tinjau graf G_1 pada Gambar 1 :

- Simpul 2 bertetangga dengan simpul 1, 3, dan 4.
- Simpul 1 tidak bertetangga dengan simpul 4.

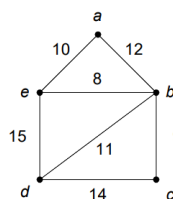
Untuk sembarang sisi $e = (v_i, v_j)$ dapat dikatakan bahwa sisi e berurutan dengan simpul v_i atau e berurutan dengan simpul v_j .

Tinjau graf G_1 pada Gambar 1 :

- Sisi (1,2) berurutan dengan simpul 1 dan simpul 2
- Sisi (1,3) berurutan dengan simpul 1 dan simpul 3
- Kedua Sisi tersebut tidak berurutan dengan simpul 4

C. Graf Berbobot (Weighted Graph)

Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga (bobot). Bobot dari sisi dapat merepresentasikan suatu harga, jarak, dan beberapa unit lain relatif dari satu simpul ke simpul lainnya.



Gambar 3 Contoh graf berbobot
sumber : [Graf \(bag.1\) - IF2120 - Rinaldi Munir](#)

D. Matriks Ketetanggaan (Adjacency Matrix)

Matriks ketetanggaan merupakan salah satu cara dalam merepresentasikan graf. Matriks ketetanggaan memiliki bentuk bujur sangkar atau jumlah baris dan kolomnya sama. Elemen pada matriks baris ke- i dan kolom ke- j menyatakan ketetanggaan dari simpul i dan simpul j . Jika elemennya bernilai nol, maka simpul tidak bertetangga. Jika elemennya bernilai satu, maka kedua simpul bertetangga. Representasi matriks ketetanggaan dapat diperluas lagi dengan mengisi elemen matriks dengan bobot sehingga matriks ketetanggaan dapat merepresentasikan graf berbobot.

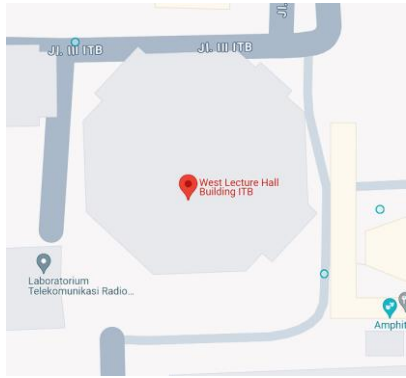
E. Algoritma Dijkstra

Algoritma Dijkstra merupakan suatu algoritma untuk menentukan lintasan terpendek dari satu simpul ke simpul lainnya. Algoritma ini dipublikasikan oleh Edsger W. Dijkstra pada tahun 1959. Algoritma ini ditemukan oleh Dijkstra pada tahun 1956 ketika ia sedang di Amsterdam dan mencari rute terpendek dari Rotterdam menuju Groningen.

Algoritma lintasan terpendek disusun berdasarkan langkah per langkah. Pada langkah pertama, dibangun lintasan terpendek pertama, pada langkah kedua dibangun lintasan terpendek kedua, sampai seluruh langkah telah habis dan didapatkan lintasan terpendek. Berikut adalah langkah-langkah dalam menentukan lintasan terpendek dalam konteks pencarian ruangan :

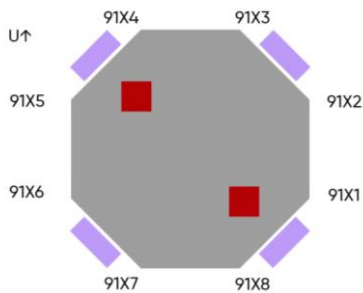
1. Kunjungi posisi ruangan awal mulai, ruangan tersebut menjadi "posisi ruangan saat ini".
2. Periksa jarak dari posisi ruangan saat ini menuju ruangan yang bertetangga.
3. Jika jarak tempuh menuju ruangan yang bertetangga lebih pendek dibandingkan jarak terpendek menuju ruangan bertetangga saat ini (atau jarak menuju ruangan bertetangga belum tercatat) :
 - a. Perbarui jarak terpendek menuju ruangan tersebut
 - b. Perbarui ruangan asal menuju ruangan tersebut dengan ruangan asal dengan jarak terpendek
4. Kemudian kunjungi ruangan terdekat dari posisi ruangan saat ini, kemudian ubah ruangan terdekat tersebut menjadi posisi ruangan saat ini.
5. Ulangi langkah 2 hingga 4 sampai seluruh ruangan telah dikunjungi.

F. GKUB ITB



Gambar 5 Peta GKUB ITB
sumber : Google Maps

GKUB ITB merupakan sebuah gedung berlantai 3 yang berada di bagian barat ITB. Gedung ini memiliki 23 ruang kelas, 6 toilet, mushola, dan kantin. Gedung ini memiliki penomoran yang unik. Sistem penomoran ruangan GKUB ITB dapat dilihat pada denah berikut ini :



Gambar 6 Denah sederhana GKUB ITB
sumber : dokumentasi penulis

Angka terakhir dari ruangan dimulai dari ruangan yang menghadap timur dengan angka 1. Kemudian berputar melawan arah jarum jam dengan angka terakhir bertambah 1 sampai angka terakhir berangka 8. Huruf X menandakan nomor lantai. Untuk lantai 1 X bernilai 0, lantai 2 X bernilai 1 dan 2, dan lantai 3, X bernilai 3.

Hanya terdapat 6 toilet pada GKUB ITB, yaitu di sisi barat laut dan timur laut pada setiap lantainya kecuali pada lantai 1. Tidak ada toilet pada lantai 1. Toilet berada di tengah lantai 1 dan lantai 2. Area toilet ditandai dengan warna merah pada denah tersebut.

Untuk berpindah dari ruangan ke ruangan lain, hampir selalu melalui bagian tangga yang diwarnai dengan warna ungu pada denah. Tangga ini cukup rumit, beberapa ruangan di lantai X dapat diakses dengan menaiki sejumlah N set tangga, tetapi di ruangan lain dengan lantai yang sama diperlukan untuk menaiki N+1 set tangga.

III. PENERAPAN ALGORITMA DIJKSTRA DALAM
PENCARIAN TOILET DI GKUB ITB

A. Deskripsi Implementasi dan Struktur Data

Langkah pertama dalam pencarian toilet terdekat dilakukan dengan membuat representasi GKUB ITB dalam bentuk graf.

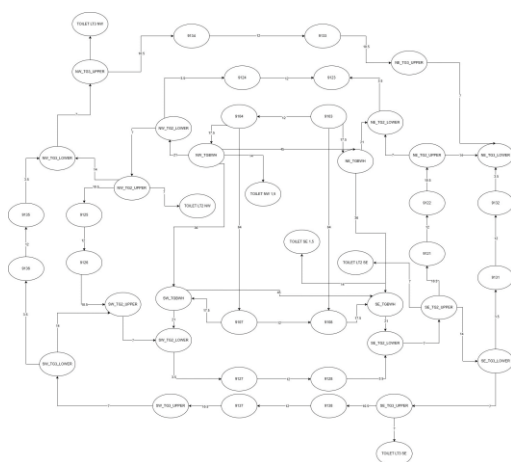
Simpul yang ada dalam graf terbagi menjadi tiga bagian, yaitu simpul ruang kelas, simpul toilet, dan simpul tangga. Simpul ruang kelas merupakan simpul yang akan digunakan sebagai titik awal dari pencarian. Lalu, ada simpul tangga yang merupakan representasi dari tangga yang ada di GKUB ITB. Simpul tangga ini diperlukan karena rumitnya akses setiap lantai dan juga sebagai penanda beberapa tangga tambahan yang posisinya berada di antara lantai 2 dan 3 atau di antara lantai 1 dan 2. Terakhir, simpul toilet digunakan sebagai tujuan akhir pencarian. Sesuai dengan kondisi pada GKUB ITB, simpul ini hanya ada pada area tenggara dan area barat laut dari GKUB ITB. Berikut adalah tabel dari graf yang digunakan dalam implementasi :

Nomor	Asal	Tujuan	Jarak (meter)
1	9103	9104	12
2	9103	9108	64
3	9103	Tangga_Lantai_Dasar_NE	18
4	9104	9103	12
5	9104	9107	64
6	9104	Tangga_Lantai_Dasar_NW	18
7	9107	9104	64
8	9107	9108	12
9	9107	Tangga_Lantai_Dasar_SW	18
10	9108	9103	64
11	9108	9107	12
12	9108	Tangga_Lantai_Dasar_SE	18
13	9121	9122	12
14	9121	Tangga_Upper_LT2_SE	11
15	9122	9121	12
16	9122	Tangga_Upper_LT2_NE	11
17	9123	9124	12
18	9123	Tangga_Lower_LT2_NE	3.5
19	9124	9123	12
20	9124	Tangga_Lower_LT2_NW	3.5
21	9125	9126	12
22	9125	Tangga_Upper_LT2_NW	11
23	9126	9125	12
24	9126	Tangga_Upper_LT2_SW	11
25	9127	9128	12
26	9127	Tangga_Lower_LT2_SW	3.5
27	9128	9127	12
28	9128	Tangga_Lower_LT2_SE	3.5
29	9131	9132	12

30	9131	Tangga_Lower_LT3 _SE	3.5
31	9132	9131	12
32	9132	Tangga_Lower_LT3 _NE	3.5
33	9133	9134	12
34	9133	Tangga_Upper_LT3 _NE	11
35	9134	9133	12
36	9134	Tangga_Upper_LT3 _NW	11
37	9135	9136	12
38	9135	Tangga_Lower_LT3 _NW	3.5
39	9136	9135	12
40	9136	Tangga_Lower_LT3 _SW	3.5
41	9137	9138	12
42	9137	Tangga_Upper_LT3 _SW	11
43	9138	9137	12
44	9138	Tangga_Upper_LT3 _SE	11
45	Toilet_NW_LT1	Tangga_Lantai_Dasa r_NW	14
46	Toilet_NW_LT2	Tangga_Upper_LT2 _NW	7
47	Toilet_NW_LT3	Tangga_Upper_LT3 _NW	7
48	Toilet_SE_LT1	Tangga_Lantai_Dasa r_SE	14
49	Toilet_SE_LT2	Tangga_Upper_LT2 _SE	7
50	Toilet_SE_LT3	Tangga_Upper_LT3 _SE	7
51	Tangga_Upper_LT2 _SE	9121	11
52	Tangga_Upper_LT2 _SE	Toilet_SE_LT2	7
53	Tangga_Upper_LT2 _SE	Tangga_Lower_LT2 _SE	7
54	Tangga_Upper_LT2 _SE	Tangga_Lower_LT3 _SE	14
55	Tangga_Upper_LT2 _NE	9122	11
56	Tangga_Upper_LT2 _NE	Tangga_Lower_LT2 _NE	7
57	Tangga_Upper_LT2 _NE	Tangga_Lower_LT3 _NE	14
58	Tangga_Upper_LT2 _NW	9125	11
59	Tangga_Upper_LT2 _NW	Toilet_NW_LT2	7
60	Tangga_Upper_LT2 _NW	Tangga_Lower_LT2 _NW	7
61	Tangga_Upper_LT2 _NW	Tangga_Lower_LT3 _NW	14
62	Tangga_Upper_LT2 _SW	9126	11
63	Tangga_Upper_LT2 _SW	Tangga_Lower_LT2 _SW	7
64	Tangga_Upper_LT2 _SW	Tangga_Lower_LT3 _SW	14

65	Tangga_Upper_LT3 _SE	9138	11
66	Tangga_Upper_LT3 _SE	Toilet_SE_LT3	7
67	Tangga_Upper_LT3 _SE	Tangga_Lower_LT3 _SE	7
68	Tangga_Upper_LT3 _NE	9133	11
69	Tangga_Upper_LT3 _NE	Tangga_Lower_LT3 _NE	7
70	Tangga_Upper_LT3 _NW	9134	11
71	Tangga_Upper_LT3 _NW	Toilet_NW_LT3	7
72	Tangga_Upper_LT3 _NW	Tangga_Lower_LT3 _NW	7
73	Tangga_Upper_LT3 _SW	9137	11
74	Tangga_Upper_LT3 _SW	Tangga_Lower_LT3 _SW	7
75	Tangga_Lower_LT2 _SE	9128	3.5
76	Tangga_Lower_LT2 _SE	Tangga_Upper_LT2 _SE	7
77	Tangga_Lower_LT2 _SE	Tangga_Lantai_Dasa r_SE	21
78	Tangga_Lower_LT2 _NE	9123	3.5
79	Tangga_Lower_LT2 _NE	Tangga_Upper_LT2 _NE	7
80	Tangga_Lower_LT2 _NE	Tangga_Lantai_Dasa r_NE	21
81	Tangga_Lower_LT2 _NW	9124	3.5
82	Tangga_Lower_LT2 _NW	Tangga_Upper_LT2 _NW	7
83	Tangga_Lower_LT2 _NW	Tangga_Lantai_Dasa r_NW	21
84	Tangga_Lower_LT2 _SW	9127	3.5
85	Tangga_Lower_LT2 _SW	Tangga_Upper_LT2 _SW	7
86	Tangga_Lower_LT2 _SW	Tangga_Lantai_Dasa r_SW	21
87	Tangga_Lower_LT3 _SE	9131	3.5
88	Tangga_Lower_LT3 _SE	Tangga_Upper_LT2 _SE	14
89	Tangga_Lower_LT3 _SE	Tangga_Upper_LT3 _SE	7
90	Tangga_Lower_LT3 _NE	9132	3.5
91	Tangga_Lower_LT3 _NE	Tangga_Upper_LT2 _NE	14
92	Tangga_Lower_LT3 _NE	Tangga_Upper_LT3 _NE	7
93	Tangga_Lower_LT3 _NW	9135	3.5
94	Tangga_Lower_LT3 _NW	Tangga_Upper_LT2 _NW	14
95	Tangga_Lower_LT3 _NW	Tangga_Upper_LT3 _NW	7
96	Tangga_Lower_LT3 _SW	9136	3.5
97	Tangga_Lower_LT3 _SW	Tangga_Upper_LT2 _SW	14

98	Tangga_Lantai_LT3 _SW	Tangga_Upper_LT3 _SW	7
99	Tangga_Lantai_Dasa r_SE	9108	18
100	Tangga_Lantai_Dasa r_SE	Toilet_SE_LT1	14
101	Tangga_Lantai_Dasa r_SE	Tangga_Lantai_LT2 _SE	21
102	Tangga_Lantai_Dasa r_SE	Tangga_Lantai_Dasa r_NE	36
103	Tangga_Lantai_Dasa r_SE	Tangga_Lantai_Dasa r_SW	45
104	Tangga_Lantai_Dasa r_NE	9103	18
105	Tangga_Lantai_Dasa r_NE	Tangga_Lantai_LT2 _NE	21
106	Tangga_Lantai_Dasa r_NE	Tangga_Lantai_Dasa r_SE	36
107	Tangga_Lantai_Dasa r_NE	Tangga_Lantai_Dasa r_NW	45
108	Tangga_Lantai_Dasa r_NW	9104	18
109	Tangga_Lantai_Dasa r_NW	Toilet_NW_LT1	14
110	Tangga_Lantai_Dasa r_NW	Tangga_Lantai_LT2 _NW	21
111	Tangga_Lantai_Dasa r_NW	Tangga_Lantai_Dasa r_NE	45
112	Tangga_Lantai_Dasa r_NW	Tangga_Lantai_Dasa r_SW	36
113	Tangga_Lantai_Dasa r_SW	9107	18
114	Tangga_Lantai_Dasa r_SW	Tangga_Lantai_LT2 _SW	21
115	Tangga_Lantai_Dasa r_SW	Tangga_Lantai_Dasa r_SE	45
116	Tangga_Lantai_Dasa r_SW	Tangga_Lantai_Dasa r_NW	36



Gambar 7 Graf representasi GKUB ITB

sumber : [hannoobz/makalah-matdis2023 \(github.com\)](https://github.com/hannoobz/makalah-matdis2023) (dokumentasi penulis)

Graf representasi GKUB ITB tersebut kemudian dimasukkan ke dalam program Algoritma Dijkstra. Dalam penerapan ini, penulis menggunakan implementasi dengan bahasa C++. Graf direpresentasikan dengan menggunakan adjacency matrix.

Urutan penamaan simpul diurutkan dari simpul ruangan, simpul toilet, lalu simpul tangga. Pengurutan simpul ruangan dalam matriks diurutkan berdasarkan dokumen [Daftar Ruang Kuliah Umum - Direktorat Sarana dan Prasarana ITB Tahun 2011](#).

Langkah yang digunakan sama seperti algoritma dijkstra pada umumnya. Pertama, periksa simpul terdekat yang belum dikunjungi. Lalu, periksa jaraknya. Jika jarak lebih besar dibandingkan dengan jarak melalui rute yang diambil, ubah asal/parent dari simpul dengan asal yang lebih dekat. Langkah ini diulangi sampai tidak ada simpul yang tersisa, atau dalam konteks kasus ini tidak ada ruangan yang belum dikunjungi.

```

procedure Dijkstra(input G: weighted_graph, input a: initial_vertex, output
L: array[1..n] of real)
{ Mencari lintasan terpendek dari simpul a ke semua simpul lain di dalam
graf berbobot G.
Masukan: graf-berbobot yang terhubung, G = (V, E) dengan |V| = n
Luaran: L[1..n], L[i] berisi panjang terpendek dari simpul a ke simpul
vi
}
Deklarasi:
i: integer
u, v: vertex
S: set of vertex { himpunan solusi untuk mencatat simpul-simpul yang sudah
dipilih di dalam tur }

Algoritma
for i ← 1 to n do
    L(vi) ← ∞
endfor
L(a) ← 0 { jarak dari a ke a adalah 0 }
S ← {a}

for k ← 1 to n do
    u ← pilih simpul yang belum terdapat di dalam S dan memiliki L(u)
    minimum
    S ← S ∪ {u} { masukkan u ke dalam S }

    for setiap simpul v yang tidak terdapat di dalam S do
        { update jarak yang baru dari a ke v }
        if L(u) + G(u, v) < L(v) then
            { jarak dari a ke u ditambah bobot sisi dari u ke v lebih kecil
            dari jarak a ke v }
            L(v) ← L(u) + G(u, v) { jarak dari a ke v yang baru diganti dengan
            L(u) + G(u, v) }
        endif
    endfor
endfor

```

Pseudocode Algoritma Dijkstra

sumber : [Algoritma Greedy \(bag.2\) - IF2211 - Rinaldi Munir](#)

B. Implementasi Algoritma Dijkstra pada Program

Pertama, program akan menampilkan panduan singkat mengenai penomoran ruangan dan arti simbol atau karakter yang akan muncul dalam program.

```

Berikut adalah panduan penomoran ruangan
X adalah nomor lantai
N adalah North
E adalah East
S adalah South
W adalah West
LTX menandakan ruangan di lantai X

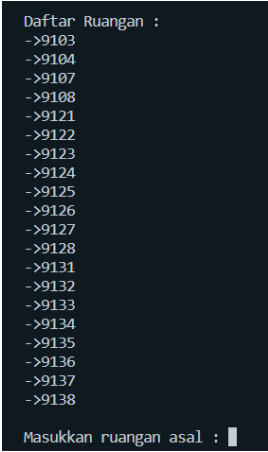
Utara1  91X4      91X3
#####
91X5 ##### 91X2
#####
#####
#####
91X6 ##### 91X1
#####
#####
91X7      91X8

```

Gambar 8 Panduan awal program

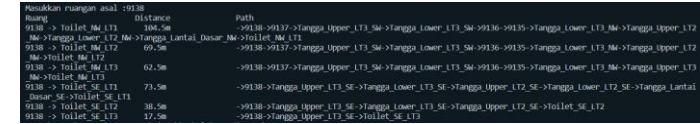
Kemudian, program akan menampilkan daftar ruang kelas yang ada di GKUB ITB dan meminta input titik awal mulai

kepada pengguna. Hanya terdapat tiga ruang kelas yang dihapus, yaitu ruang 9114, 9115, dan 9116 karena jalur menuju ruangan yang linier dan ruangan jarang dipakai.



Gambar 9 Daftar ruangan dan input titik awal

Setelah pengguna memasukkan nama ruangan titik awal, program akan menampilkan seluruh toilet yang ada GKUB ITB dan juga menampilkan jalur untuk menuju toilet tersebut.



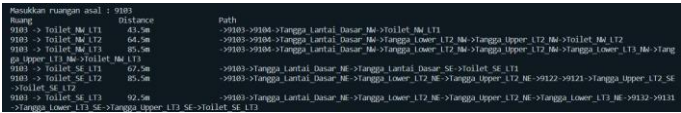
Gambar 10 Keluaran hasil pencarian rute toilet

Pengguna dapat mengikuti alur yang ditampilkan oleh program untuk mencari toilet terdekat. Pengguna dapat memilih beberapa alternatif toilet karena program juga akan tetap menampilkan toilet lain selain toilet yang terdekat dari ruangan awal.

IV. HASIL IMPLEMENTASI DAN PEMBAHASAN

A. Deskripsi & Ujicoba Hasil Pencarian Toilet

1. Ruang 9103



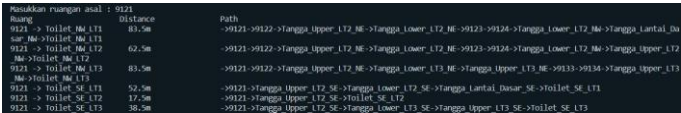
Gambar 11 Hasil ujicoba untuk ruang kuliah 9103

Ruang 9103 berada pada lantai 1 area timur laut GKUB ITB. Toilet terdekat dari ruang 9103 berada di dekat tangga sebelah tangga dasar menuju kelas lantai 2 dan 3. Tangga tersebut mengarah ke toilet yang terletak di antara lantai 1 dan lantai 2. Posisi tangga ini dapat membuat orang melewati keberadaan toilet ini.

Jalur Terdekat :
->9103->9104->Tangga_Lantai_Dasar_NW->Toilet_NW_LT1

Jarak :
43.5 meter

2. Ruang 9121



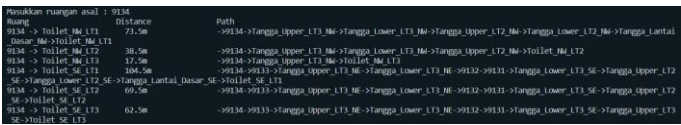
Gambar 12 Hasil ujicoba untuk ruang kuliah 9121

Ruang 9121 berada di lantai 2 area tenggara GKUB ITB. Ruangan ini dapat diakses melalui tangga tenggara GKUB ITB. Untuk ruangan yang menghadap timur atau barat dan berada di lantai 2, diperlukan untuk naik 4 set tangga (12 anak tangga setiap set dan berbalik arah setelah 1 set) dan turun 1 set tangga untuk mencapai ruang 9121 dari lantai 1. Setelah menaiki 4 set tangga dan sebelum turun 1 set tangga menuju 9121, terdapat toilet di dekat posisi tersebut dan mengarah ke pusat bangunan. Toilet tersebut merupakan toilet terdekat dari ruang 9121

Jalur Terdekat :
->9121->Tangga_Upper_LT2_SE->Toilet_SE_LT2

Jarak :
17.5 meter

3. Ruang 9134



Gambar 13 Hasil ujicoba untuk ruang kuliah 9134

Ruang 9134 merupakan ruangan yang berada pada lantai 3 area barat laut GKUB ITB. Hampir sama dengan ruang 9121, diperlukan untuk turun 1 set tangga tetapi dengan perbedaan jumlah naik tangga sebanyak 6 set terlebih dahulu. Setelah menaiki 6 set tangga, dapat terlihat toilet di arah pusat bangunan. Toilet tersebut merupakan toilet terdekat dari ruang 9134.

Jalur Terdekat :
->9134->Tangga_Upper_LT3_NW->Toilet_NW_LT3

Jarak :
17.5 meter

B. Pembahasan

Berdasarkan tiga kasus ujicoba dengan ruangan awal 9103, 9121, dan 9134, program sudah menampilkan toilet terdekat dari ruangan tersebut. Untuk jarak yang ditampilkan, kemungkinan masih terdapat miscalculasi karena keterbatasan alat untuk menghitung jarak dari satu titik ke titik lain. Namun, untuk jalur yang diambil sudah cukup tepat.

Pada ruang awal 9103, program mengarahkan untuk melewati 9104, dan kemudian menuju dekat tangga bagian bawah. Di sebelah tangga terdapat tangga lain menuju toilet. Toilet tersebut merupakan toilet terdekat dan toilet yang dapat diakses dengan menaiki tangga lebih sedikit dibanding toilet lainnya.

Pada ruang awal 9121, ruangan tersebut sudah berada di bagian tenggara gedung. Toilet di GKUB hanya ada pada bagian barat laut dan tenggara dari gedung. Hanya diperlukan naik 1 set tangga ke arah tenggara untuk menuju toilet terdekat. Keluaran

program sudah sesuai dengan kondisi GKUB.

Terakhir, hampir sama dengan 9121, ruang 9134 berada pada bagian barat daya. Bagian ini merupakan bagian yang dekat dengan toilet. Hanya perlu naik 1 set tangga dan toilet sudah dapat terlihat dari posisi tersebut. Keluaran program sudah sesuai dengan kondisi GKUB.

V. KESIMPULAN

Berdasarkan penerapan yang telah dilakukan, algoritma dijkstra cocok untuk diterapkan dalam kasus pencarian toilet di GKUB ITB. Ruangan yang memiliki akhiran nomor 1, 8, 5, dan 4 cenderung memerlukan langkah jalur yang lebih pendek karena posisinya berada pada bagian yang sama dengan toilet di GKUB ITB, yaitu pada bagian barat laut atau bagian tenggara dari gedung ini. Ruangan dengan nomor tersebut umumnya hanya perlu menaiki 1 set tangga untuk mencapai toilet.

Ruangan dengan akhiran 2, 7, 6, 3 memerlukan langkah tambahan untuk mencari toilet. Umumnya ruangan tersebut memilih jalur ke ruangan dengan akhiran 1, 8, 5, atau 4 yang terdekat, kemudian memilih jalur yang sama seperti ketika input ruangan awal adalah salah satu dari ruangan yang berakhiran 1, 8, 5 atau 4.

Dari keseluruhan, hampir dapat dipastikan bahwa toilet terdekat dari ruang kelas merupakan toilet yang posisinya lebih tinggi dari ruang kelas tersebut. Toilet yang berada 1 lantai di bawah memerlukan jarak yang lebih besar dibandingkan dengan toilet yang berada di lantai yang sama atau 1 lantai lebih tinggi. Dengan demikian, langkah terbaik untuk mencari toilet di GKUB ITB adalah dengan mendekati ruangan berakhiran 1, 8, 5, atau 4, dan menaiki 1 set tangga terdekat dari ruangan tersebut.

VII. UCAPAN TERIMA KASIH

Penulis mengucapkan rasa syukur kepada Allah Subhanahu wa ta'ala karena telah memberikan kesehatan dan kelancaran kepada penulis dalam membuat makalah berjudul "Penerapan Algoritma Dijkstra Dalam Pencarian Toilet Terdekat di GKU Barat ITB". Penulis juga ingin mengucapkan terima kasih kepada orangtua yang telah mendorong dan mendukung penulis untuk bisa berkuliah di Institut Teknologi Bandung. Terakhir, penulis juga ingin mengucapkan terima kasih kepada Dr. Fariska Zakhralatifa Ruskanda, S.T., M.T. dan Dr. Ir. Rinaldi, M.T. selaku dosen dari mata kuliah IF2120 Matematika Diskrit yang telah memberikan materi dan arahan mengenai penerapan matematika dalam kehidupan sehari-hari. Ide tentang pencarian toilet ini telah lama ingin penulis wujudkan karena pengalaman penulis yang kesulitan dalam mencari toilet di GKUB ITB. Penulis berharap makalah ini berguna bagi mahasiswa angkatan selanjutnya yang mungkin akan melaksanakan kegiatan perkuliahan di GKUB ITB.

REFERENSI

- [1] R. Munir, "Graf (Bag.1)," 2023. [Online]. Available: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/19-Graf-Bagian1-2023.pdf>.

- [2] R. Munir, "Graf (Bag.2)," 2023. [Online]. Available: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/19-Graf-Bagian2-2023.pdf>.
- [3] R. Munir, "Algoritma Greedy (Bagian 2)," 2021. [Online]. Available: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag2.pdf).
- [4] Direktorat Sarana dan Prasarana, "Daftar Ruang Kuliah Umum," [Online]. Available: <https://fa.itb.ac.id/wp-content/uploads/sites/13/2011/02/data-nama-ruang-kuliah-umum-di-sp.pdf>.
- [5] J. Wengrow, "A Common-Sense Guide to Data Structures and Algorithms, Second Edition: Level Up Your Core Programming Skills," 2020.
- [6] GeeksforGeeks, "How to find Shortest Paths from Source to all Vertices using Dijkstra's Algorithm," <https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/>, Last updated: November 23, 2023, Accessed: December 9, 2023.
- [7] P. Frana, "An Interview with Edsger W. Dijkstra," *Communications of the ACM*, vol. 53, no. 8, pp. 41–47, Aug. 2010, doi:10.1145/1787234.1787249.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 10 Desember 2023



M. Hanief Fatkhan Nashrullah / 13522100

LAMPIRAN

1. *Source code* : [hannoobz/makalah-matdis2023 \(github.com\)](https://github.com/hannoobz/makalah-matdis2023)
2. Graf GKUB ITB : [makalah-matdis2023/doc/graf.pdf at main · hannoobz/makalah-matdis2023 \(github.com\)](https://github.com/hannoobz/makalah-matdis2023/blob/main/doc/graf.pdf)
3. Adjacency Matrix : [makalah-matdis2023/doc/adjacency matrix.xlsx at main · hannoobz/makalah-matdis2023 \(github.com\)](https://github.com/hannoobz/makalah-matdis2023/blob/main/doc/adjacency%20matrix.xlsx)