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*—The abstract is to be in fully-justified italicized text, at the top of the left-hand column as it is here, below the author information. The abstract is to be in 9-point, single-spaced type, and may be up to 8 cm long. Define all symbols used in the abstract. Do not cite references in the abstract. Do not delete the blank line immediately above the abstract; it sets the footnote at the bottom of this column. Leave two blank lines after the index terms, then begin the main text. All manuscripts must be in English.

*Keywords*—About four key words or phrases in alphabetical order, separated by commas.

# 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

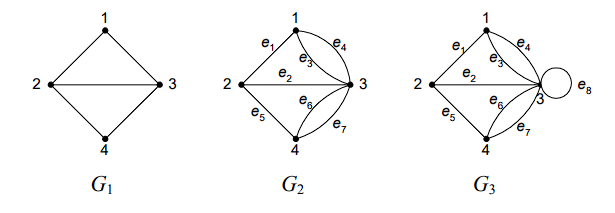
*G* = (*V*, *E*), dengan *V* dan *E :*

*V* = himpunan tidak-kosong dari simpul-simpul​

*=* {*v1,v2,...,vn*}​

*E* = himpunan sisi yang menghubungkan sepasang simpul

*=* {*e1,e2,...,en*}​



Gambar 2 Contoh graf,  
(a) graf sederhana, (b) graf ganda, dan (c) graf semu  
sumber : [Graf (bag.1) - IF2120 - Rinaldi Munir](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/19-Graf-Bagian1-2023.pdf)

## B. Ketetanggaan Simpul dan Sisi Beririsan

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

Tinjau graf G1 pada Gambar 1 :

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

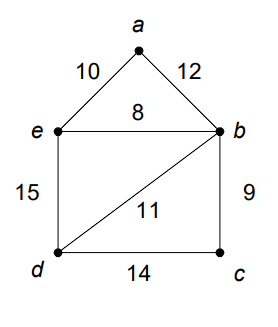
Untuk sembarang sisi *e* = (*vi,vj*) dapat dikatakan bahwa sisi *e* beririsan dengan simpul *vi ­­­­­* atau *e* beririsan dengan simpul *vj*.

Tinjau graf G1 pada Gambar 1 :

* Sisi (1,2) beririsan dengan simpul 1 dan simpul 2
* Sisi (1,3) beririsan dengan simpul 1 dan simpul 3
* Kedua Sisi tersebut tidak beririsan dengan simpul 4

## C. Graf Berbobot (Weighted Graph)

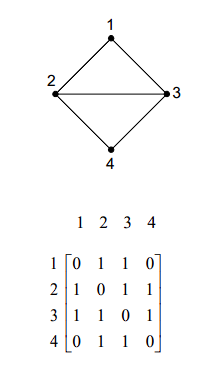
Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga (bobot). Bobot dari sisi dapat merepresentasikan suatu harga, jarak, dan beberapa unit lain relatif dari satu simpul ke simpul lainnya.



Gambar 3 Contoh graf berbobot  
sumber : [Graf (bag.1) - IF2120 - Rinaldi Munir](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/19-Graf-Bagian1-2023.pdf)

## 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.



Gambar 4 Representasi Graf dalam Matriks Ketetanggaan  
sumber : [Graf (bag.2) - IF2120 - Rinaldi Munir](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/20-Graf-Bagian2-2023.pdf)

## E. Algoritma Dijkstra

Algoritma Dijkstra merupakan suatu algoritma untuk menentukan lintasan terpendek dari 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.

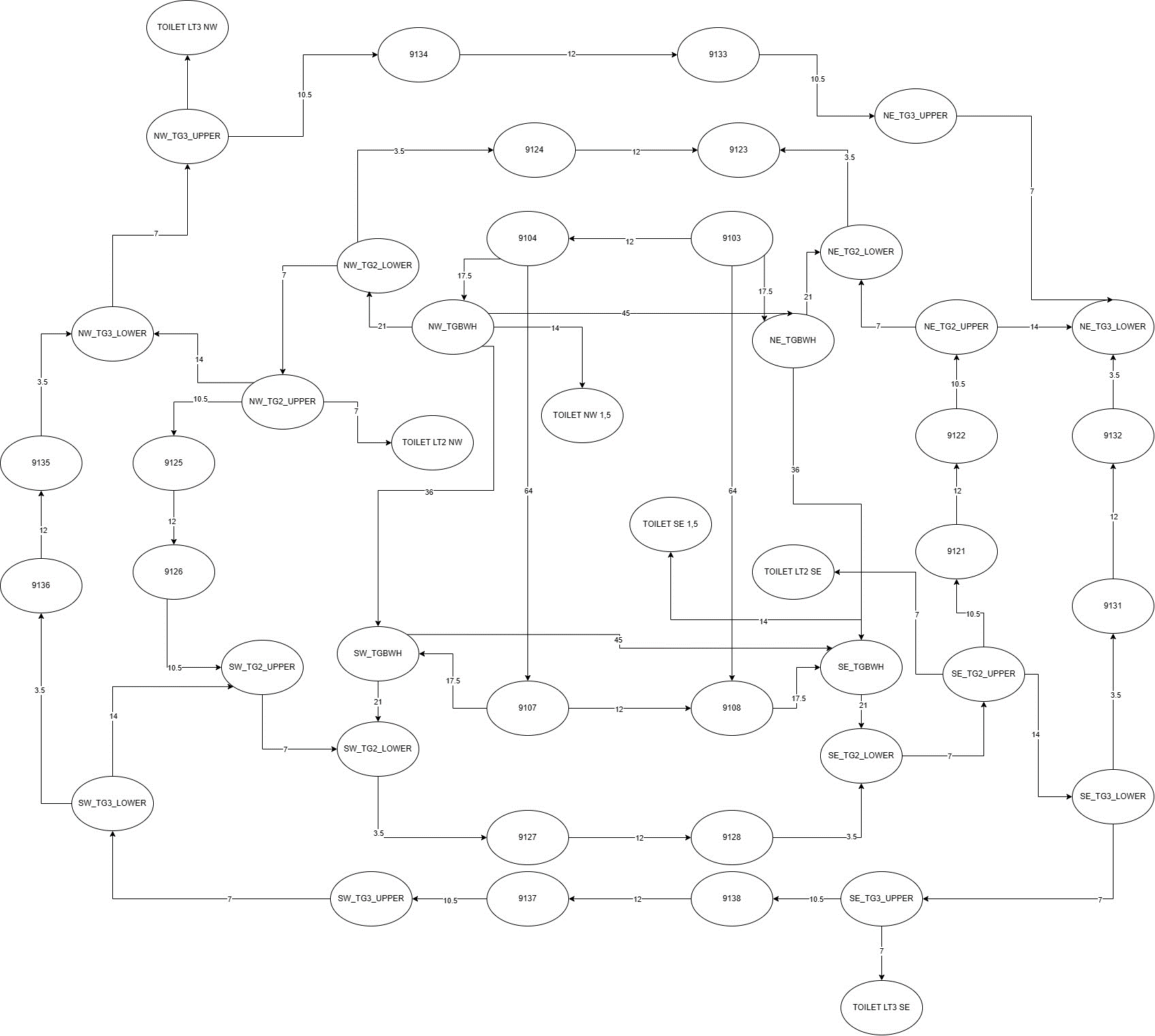
Algoritma lintasan terpendek disusun berdasarkan langkah per langkah. Pada langkah pertama, dibangun lintasan terpendek pertama, pada langkah kedua dibangun lintasan terpendek kedua, sampai seluruh langkah telah habis dan didapatkan lintasan terpendek. Berikut adalah langkah-langkah dalam menentukan 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) :
   1. Perbarui jarak terpendek menuju ruangan tersebut
   2. 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.

# 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.



Gambar 5 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 matriksn diurutkan berdasarkan dokumen [Daftar Ruang Kuliah Umum - Direktorat Sarana dan Prasarana ITB Tahun 2011](https://fa.itb.ac.id/wp-content/uploads/sites/13/2011/02/data-nama-ruang-kuliah-umum-di-sp.pdf).

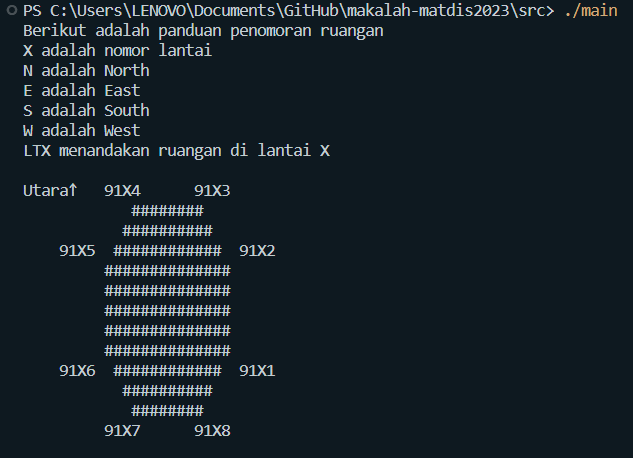
Langkah yang digunakan sama seperti algoritma dijkstra pada umumnya. Pertama, periksa simpul terdekat yang belum dikunjungi. Lalu, periksa jaraknya. Jika jarak lebih besar dibandingkan dengan jarak melalui rute yang diambil, ubah asal/parent dari simpul dengan asal yang lebih dekat. Langkah ini diulangi 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 ← {}  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](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag2.pdf)

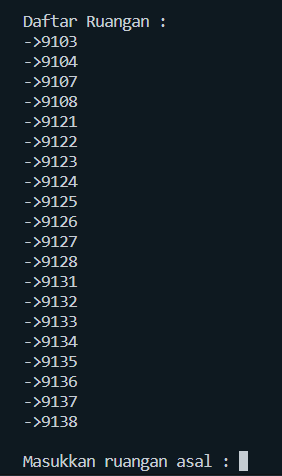
## B. Implementasi Algoritma Dijkstra pada Program

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



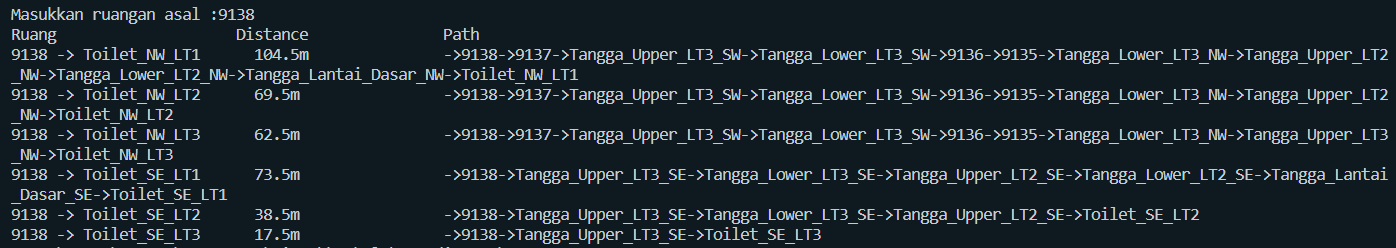
Gambar 6 Panduan awal program

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 7 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 8 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.

# V. Conclusion

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.

# VII. Acknowledgment

The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank ... .” Instead, write “F. A. Author thanks ... .” Sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page.

# References

1. G. O. Young, “Synthetic structure of industrial plastics (Book style with paper title and editor),” in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.
2. W.-K. Chen, *Linear Networks and Systems* (Book style)*.* Belmont, CA: Wadsworth, 1993, pp. 123–135.
3. H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.
4. B. Smith, “An approach to graphs of linear forms (Unpublished work style),” unpublished.
5. E. H. Miller, “A note on reflector arrays (Periodical style—Accepted for publication),” *IEEE Trans. Antennas Propagat.*, to be published.
6. J. Wang, “Fundamentals of erbium-doped fiber amplifiers arrays (Periodical style—Submitted for publication),” *IEEE J. Quantum Electron.*, submitted for publication.
7. C. J. Kaufman, Rocky Mountain Research Lab., Boulder, CO, private communication, May 1995.

# 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, 3 Desember 2023

Ttd (scan atau foto ttd)

M. Hanief Fatkhan Nashrullah / 13522100