Penerapan Aljabar Boolean Pada Mesin Penukaran Koin Sederhana

Husnia Munzayana - 13521077
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13521077@std.stei.itb.ac.id

***Abstract*— Meskipun sebagian besar transaksi di era globalisasi sudah beralih menjadi transaksi secara digital, ternyata masih terdapat beberapa tempat yang mengharuskan kita menggunakan koin untuk melakukan transaksi. Misalnya, untuk menggunakan mesin *laundry* koin, mesin permainan *arcade*, mesin jaja otomatis, dan sebagainya. Koin dianggap tidak efisien untuk dibawa, sehingga tidak semua orang selalu sedia koin di dompetnya. Mesin penukar koin dikembangkan sebagai solusi untuk masalah tersebut. Mesin penukar koin memungkinkan kita mengubah uang kertas menjadi uang koin dengan nominal yang cenderung lebih kecil atau menukarkan koin khusus yang digunakan untuk menjalankan suatu alat. Mesin penukaran koin ini dapat menghitung jumlah uang yang dimasukkan pengguna, mendeteksinya, dan kemudian mengeluarkan jumlah koin yang sesuai. Perhitungan jumlah koin yang dikeluarkan alat ini dapat dikaitkan dengan penerapan prinsip aljabar boolean untuk kemudian direpresentasikan melalui rangkaian logika. Untuk lebih memahami bagaimana aljabar boolean dapat diterapkan pada mesin penukar koin, akan dilakukan studi lebih mendalam pada makalah ini.**

***Keywords*— Aljabar Boolean, Mesin Penukar Koin, Rangkaian Logika, Transaksi**

I. Pendahuluan

Menurut Kamus Besar Bahasa Indonesia (KBBI), uang adalah alat tukar atau standar pengukur nilai (kesatuan hitungan) yang sah, dikeluarkan oleh pemerintah suatu negara berupa kertas, emas, perak, atau logam lain yang dicetak dengan bentuk dan gambar tertentu[2]. Uang sebagai alat pembayaran yang sah secara fisik beredar dalam bentuk uang kertas maupun uang logam atau koin [3]. Biasanya, uang koin memiliki nilai mata uang yang relatif kecil sehingga beberapa orang mungkin merasa tidak akan membutuhkannya dan memilih untuk hanya membawa uang kertas saja.

Akan tetapi, dalam beberapa kondisi di tempat tertentu kita mungkin saja sangat membutuhkan uang koin untuk melakukan transaksi. Sebagai contoh, ketika kita hendak mencuci baju pada layanan binatu atau *laundry* di tempat umum dengan mesin yang mengharuskan pelanggannya untuk memasukkan beberapa uang koin sebelum menggunakan mesin tersebut. Selain itu, beberapa tempat seperti tempat permainan *arcade*, tempat parkir, atau mesin jaja otomatis biasanya juga secara khusus menerima uang koin. Meskipun hal tersebut dianggap kurang efektif, tetapi tidak dapat dipungkiri bahwa uang koin masih banyak diperlukan.

Gambar 1.1 Mesin *laundry* koin (Sumber: https://www.youtube.com/watch?v=NxI8QE9tOYU)

Untuk mengatasi permasalahan tersebut, tidak jarang pemilik tempat *laundry*, tempat permainan *arcade*, dan tempat-tempat lain yang mengkhususkan penggunaan uang koin tersebut menyediakan pula mesin penukar uang koin. Mesin penukar uang koin adalah alat yang digunakan untuk menukarkan uang kertas dengan nilai yang relatif besar menjadi uang koin dengan nilai yang lebih kecil. Adanya mesin penukar uang koin ini tentu sangat memudahkan pelanggan sehingga mereka tidak perlu repot mengumpulkan uang koin untuk dapat melakukan transaksi.

Gambar 1.2 Mesin Penukar Uang Koin (Sumber: https://www.youtube.com/watch?v=NxI8QE9tOYU)

Perancangan mesin penukaran uang koin ini tentunya sangat kompleks. Dibutuhkan alat pemindai mata uang untuk mengenali besaran uang yang dimasukkan pengguna serta rangkaian validasi untuk menentukan jumlah koin yang akan dikeluarkan. Sistem validasi ini secara sederhana dapat dikaitkan dengan penerapan konsep aljabar boolean. Konsep aljabar boolean sangat berguna dan memiliki banyak peranan dalam penyelesaian masalah kehidupan. Pada makalah ini, akan dibahas secara detail penerapan konsep aljabar boolean pada mesin penukaran uang koin.

II. TEORI DASAR

A. Pengantar Aljabar Boolean

Aljabar boolean adalah cabang ilmu aljabar mengenai aturan dasar logika yang didasarkan pada pernyataan logika bernilai benar atau salah. Pada tahun 1854, George Boole, seorang matematikawan berkebangsaan Inggris. Beliau memperkenalkan aljabar boolean sebagai suatu sistem analisis matematis mengenai logika. Aturan-aturan dasar logika tersebut diperkenalkan melalui bukunya, *The Laws of Thought* [5]. Dalam buku *Schaum's Outline of Discrete Mathematics*, Aljabar boolean didefinisikan sebagai berikut :

Misalkan B adalah himpunan tidak kosong dengan dua operator biner, + dan ·[1], dan sebuah operator uner, ′[2], dan dua elemen berbeda 0 dan 1:

$$< B, +, \cdot, ', 0, 1 >$$

Maka, B disebut aljabar boolean jika untuk setiap a, b, c ϵ B berlaku aksioma berikut:

a. Hukum Komutatif
   (i) $a + b = b + a$
   (ii) $a \cdot b = b \cdot a$
b. Hukum Distributif
   (i) $a + (b \cdot c) = (a + b) \cdot (a + c)$
   (ii) $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$
c. Hukum Identitas
   (i) $a + 0 = a$
   (ii) $a \cdot 1 = a$
d. Hukum Komplemen
   (i) $a + a' = 1$
   (ii) $a \cdot a' = 0$ [4]

Aljabar boolean yang didefinisikan di atas juga sering disebut sebagai aljabar boolean dua-nilai dengan himpunan B terdiri dari dua buah elemen, yaitu 0 dan 1, dan memenuhi kaidah operator biner dan uner:

$\boldsymbol{a}$	$\boldsymbol{b}$	$\boldsymbol{a+b}$
0	0	0
0	1	1
1	0	1
1	1	1

Tabel 2.1.1 Operasi Biner +

$\boldsymbol{a}$	$\boldsymbol{b}$	$\boldsymbol{a \cdot b}$
0	0	0
0	1	0
1	0	0
1	1	1

Tabel 2.1.2 Operasi Biner ·

$\boldsymbol{a}$	$\boldsymbol{a'}$
0	1
1	0

Tabel 2.1.3 Operasi Uner ′

Elemen-elemen himpunan B tersebut dapat dikombinasikan dengan operator biner dan uner sehingga dapat membentuk ekspresi boolean.

B. Hukum-hukum Aljabar Boolean

Dalam menentukan nilai kebenaran atau hasil sebuah ekspresi boolean, kita perlu menyederhanakan ekspresi boolean tersebut. Penyederhanaan ekspresi boolean ini dapat dilakukan dengan menerapkan hukum-hukum aljabar boolean sebagai berikut:

a. Hukum identitas:
   (i) $a + 0 = a$
   (ii) $a \cdot 1 = a$
b. Hukum idempoten:
   (i) $a + a = a$
   (ii) $a \cdot a = a$
c. Hukum komplemen:
   (i) $a + a' = 1$
   (ii) $aa' = 0$
d. Hukum dominansi:
   (i) $a \cdot 0 = 0$
   (ii) $a + 1 = 1$
e. Hukum involusi:
   (i) $(a')' = a$
f. Hukum penyerapan:
   (i) $a + ab = a$
   (ii) $a(a + b) = a$
g. Hukum komutatif:
   (i) $a + b = b + a$
   (ii) $ab = ba$
h. Hukum asosiatif:
   (i) $a + (b + c) = (a + b) + c$
   (ii) $a\,(b\,c) = (a\,b)\,c$
i. Hukum distributif:
   (i) $a + (bc) = (a + b)\,(a + c)$
   (ii) $a\,(b + c) = a\,b + a\,c$
j. Hukum De Morgan:
   (i) $(a + b)' = a'b'$
   (ii) $(ab)' = a' + b'$
k. Hukum 0/1
   (i) $0' = 1$
   (ii) $1' = 0$

[1] Tanda dot (·) menyatakan perkalian, sama seperti tanda bintang (*)
[2] Tanda petik satu (′) menyatakan komplemen, sama seperti tanda (¯)

C. Fungsi Boolean

Dalam aljabar boolean, sebuah variabel atau peubah boolean dapat bernilai 0 (salah) atau 1 (benar). Himpunan variabel boolean ini dapat dipetakan pada sebuah ekspresi boolean melalui fungsi boolean, contohnya :

$$f(x, y, z) = xy + y'z$$

dengan x, y, dan z adalah variabel boolean. Setiap variabel dalam fungsi boolean ini, termasuk bentuk komponennya, disebut sebagai literal. Terdapat tiga operasi fungsi boolean, yaitu penjumlahan, perkalian, dan operasi komplemen.

Fungsi boolean dapat diekspresikan dalam bentuk kanonik, yaitu setiap suku(term) dalam ekspresi boolean mengandung literal yang lengkap. Terdapat dua jenis bentuk kanonik:

a. SOP (*Sum of Product*)
   Sum of Product atau penjumlahan dari hasil kali (lambang: $\sum$) adalah bentuk ekspresi boolean yang menjumlahkan setiap sukunya, dengan setiap suku dalam bentuk hasil kali variabel boolean (*minterm*, lambang: m).
b. POS (*Product of Sum*)
   Product of Sum atau perkalian dari hasil jumlah (lambang: $\prod$) adalah ekspresi boolean mengalikan setiap sukunya, dengan setiap suku dalam bentuk hasil jumlah variabel boolean (*maxterm*, lambang: M).

D. Rangkaian Logika

Fungsi boolean juga dapat direpresentasikan dalam bentuk rangkaian logika. Rangkaian logika memiliki tiga gerbang logika dasar, yaitu gerbang AND, OR, dan NOT. Kombinasi gerbang dasar ini dapat menghasilkan gerbang-gerbang lain, seperti NAND, NOR, XOR, dan XNOR.

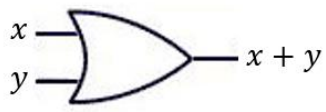
a. AND
   Gerbang logika AND akan menghasilkan nilai 0 atau *false* jika terdapat salah satu *input* bernilai 0 atau *false* dan menghasilkan nilai 1 atau *true* jika semua *input* bernilai 1 atau *true*.

Gambar 2.4.1 Gerbang Logika AND

x	y	xy
0	0	0
0	1	0
1	0	0
1	1	1

Tabel 2.4.1 Tabel Kebenaran AND

b. OR
   Gerbang logika OR akan menghasilkan nilai 0 atau *false* jika semua *input* bernilai 0 atau *false* dan menghasilkan nilai 1 atau *true* jika terdapat salah satu *input* bernilai 1 atau *true.*

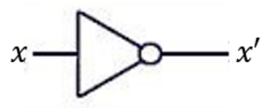


Gambar 2.4.2 Gerbang Logika OR

x	y	x + y
0	0	0
0	1	1
1	0	1
1	1	1

Tabel 2.4.2 Tabel Kebenaran OR

c. NOT
   Gerbang logika NOT berfungsi sebagai pembalik keadaan. Jika *input* bernilai 1 atau *true,* maka akan keluaran yang dihasilkan bernilai 0 atau *false.*

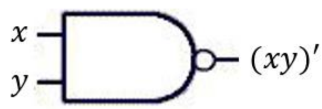


Gambar 2.4.3 Gerbang Logika NOT

x	x′
0	1
1	0

Tabel 2.4.3 Tabel Kebenaran NOT

d. NAND
   Gerbang logika NAND adalah kombinasi gerbang logika AND dan NOT. Nilai keluaran yang dihasilkan gerbang logika ini berkebalikan dengan AND.

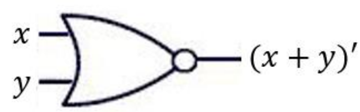


Gambar 2.4.4 Gerbang Logika NAND

x	y	(xy)′
0	0	1
0	1	1
1	0	1
1	1	0

Tabel 2.4.4 Tabel Kebenaran NAND

e. NOR
   Gerbang logika NOR adalah kombinasi gerbang logika OR dan NOT. Nilai keluaran yang dihasilkan gerbang logika ini berkebalikan dengan OR.

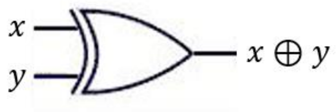


Gambar 2.4.5 Gerbang Logika NOR

x	y	(x + y)′
0	0	1
0	1	0
1	0	0
1	1	0

Tabel 2.4.5 Tabel Kebenaran NOR

f. XOR
Gerbang logika XOR atau *Exclusive* OR membutuhkan dua buah *input*. Gerbang logika ini akan menghasilkan nilai 1 atau *true* jika kedua *input* memiliki nilai yang berbeda, dan akan bernilai 0 atau *false* jika kedua *input* memiliki nilai yang sama.

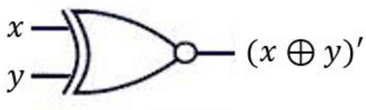


Gambar 2.4.6 Gerbang Logika XOR

x	y	x ⊕ y
0	0	0
0	1	1
1	0	1
1	1	0

Tabel 2.4.6 Tabel Kebenaran XOR

g. XNOR
Gerbang logika XNOR adalah kombinasi gerbang logika XOR dan NOT. Nilai keluaran yang dihasilkan gerbang logika ini berkebalikan dengan XOR.



Gambar 2.4.7 Gerbang Logika XNOR

x	y	(x ⊕ y)′
0	0	1
0	1	0
1	0	0
1	1	1

Tabel 2.4.7 Tabel Kebenaran XNOR

D. Peta Karnaugh

Terdapat tiga metode dalam penyederhanaan fungsi boolean, antara lain penyederhanaan secara aljabar menggunakan hukum aljabar boolean, metode Peta Karnaugh, dan metode Quine-Mccluskey atau metode tabulasi. Peta Karnaugh (K-map) adalah metode penyederhanaan fungsi boolean secara grafis. Metode ini pertama kali diperkenalkan pada tahun 1953 oleh Maurice Karnaugh untuk meminimalisasi sirkuit secara manual.

Peta Karnaugh digunakan untuk menemukan suku-suku yang dapat digabungkan sehingga dapat menghasilkan fungsi boolean yang melibatkan variabel seminimal mungkin. Setiap elemen bernilai 1 pada Peta Karnaugh merepresentasikan *minterm*. Setiap kotak dapat dikatakan bertetangga atau bersisian jika representasi *minterm* berbeda 1 literal, sebagai contoh untuk fungsi boolean dengan 4 peubah:

wx / yz	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_8	m_9	m_{11}	m_{10}
10	m_{12}	m_{13}	m_{15}	m_{14}

Tabel 2.5 Peta Karnaugh Fungsi Boolean 4 Peubah

Minimalisasi pada Peta Karnaugh dilakukan dengan menggabungkan kotak yang bernilai 1 dan saling bersisian membentuk pasangan (2 elemen), *kuad* (4 elemen), atau oktet (8 elemen). Elemen Peta Karnaugh bertanda X merepresentasikan kondisi *don't care*. Kondisi ini terjadi ketika nilai dari peubah dalam fungsi boolean tidak akan berpengaruh terhadap keluaran fungsi tersebut. Dalam menyederhanakan Peta Karnaugh, simbol X ini dapat direpresentasikan sebagai 0 atau 1 sesuai kebutuhan pengelompokan.

III. Penerapan Aljabar Boolean Pada Mesin Penukaran Uang Koin

Mesin penukaran uang koin memiliki sistem yang cukup kompleks, sehingga untuk mengilustrasikan penerapan aljabar boolean pada mesin penukaran koin, akan diterapkan beberapa batasan:

a. Mesin hanya dapat menukarkan uang menjadi uang koin pecahan Rp500.
b. Mesin maksimal dapat mengeluarkan 6 koin dalam satu kali penukaran.
c. Pembayaran hanya dapat dilakukan dengan uang kertas nominal Rp1.000 saja, Rp2.000 saja, atau kombinasi keduanya.

Dalam merepresentasikan sistem mesin penukaran koin sederhana menjadi bentuk fungsi boolean, akan digunakan beberapa variabel, antara lain:

a. Variabel w dan x merepresentasikan jumlah uang pecahan Rp2.000 dalam bentuk biner
b. Variabel y dan z merepresentasikan jumlah uang pecahan Rp1.000 dalam bentuk biner

Oleh karena pecahan koin yang dapat dikeluarkan mesin senilai Rp500 dan masukkan uang yang diterima mesin dalam kelipatan Rp1000, maka akan digunakan tiga fungsi boolean:

a. f(w,x,y,z) merepresentasikan dua koin pertama atau didapatkan keluaran 2 koin.
b. g(w,x,y,z) merepresentasikan dua koin kedua atau didapatkan keluaran 4 koin.
c. h(w,x,y,z) merepresentasikan dua koin ketiga atau didapatkan keluaran 6 koin.

Pada fungsi f(w,x,y,z), agar dihasilkan nilai *true* atau 1 dibutuhkan minimum uang senilai Rp1000. Maka, jumlah uang yang digunakan maksimum 1 buah pecahan Rp1.000 atau maksimum 1 buah uang pecahan Rp2.000. Jumlah uang yang lebih dari jumlah uang maksimum dengan total nominal lebih dari Rp1.000 dianggap sebagai kondisi *don't care* (X) atau tidak dihiraukan nilainya.

Pada fungsi g(w,x,y,z), agar dihasilkan nilai *true* atau 1 dibutuhkan minimum uang senilai Rp2.000. Maka, jumlah uang yang digunakan maksimal 2 buah uang pecahan Rp1.000 atau maksimum 1 buah uang pecahan Rp2.000. Jumlah uang yang lebih dari jumlah uang maksimum dengan total nominal lebih dari Rp2.000 dianggap sebagai kondisi *don't care* (X) atau tidak dihiraukan nilainya.

Pada fungsi h(w,x,y,z), agar dihasilkan nilai *true* atau 1 dibutuhkan minimum uang senilai Rp3.000. Maka, jumlah uang yang digunakan maksimal 3 buah uang pecahan Rp1.000 atau maksimum 2 buah uang pecahan Rp2.000. Jumlah uang yang lebih dari jumlah uang maksimum dengan total nominal lebih dari Rp3.000 dianggap sebagai kondisi *don't care* (X) atau tidak dihiraukan nilainya.

Representasi sistem mesin penukaran koin sederhana tersebut dalam tabel adalah sebagai berikut:

Jumlah uang Rp2.000	Jumlah uang Rp1.000	w	x	y	z	f(w,x,y,z)	g(w,x,y,z)	h(w,x,y,z)
0	0	0	0	0	0	0	0	0
0	1	0	0	0	1	1	0	0
0	2	0	0	1	0	X	1	0
0	3	0	0	1	1	X	X	1
1	0	0	1	0	0	1	1	0
1	1	0	1	0	1	X	X	1
1	2	0	1	1	0	X	X	1
1	3	0	1	1	1	X	X	X
2	0	1	0	0	0	X	X	1
2	1	1	0	0	1	X	X	X
2	2	1	0	1	0	X	X	X
2	3	1	0	1	1	X	X	X
3	0	1	1	0	0	X	X	X
3	1	1	1	0	1	X	X	X
3	2	1	1	1	0	X	X	X
3	3	1	1	1	1	X	X	X

Tabel 3.1 Sistem Mesin Penukar Koin Sederhana

Dari tabel sistem mesin penukaran koin sederhana tersebut kita dapat merepresentasikan setiap fungsi boolean dalam Peta Karnaugh kemudian menyederhanakan fungsi boolean tersebut sehingga didapatkan Peta Karnaugh sebagai berikut:

a. Fungsi Boolean f(w, x, y, z)

wx / yz	**00**	**01**	**11**	**10**
00	0	1	X	X
01	1	X	X	X
11	X	X	X	X
10	X	X	X	X

Tabel 3.2 Peta Karnaugh Fungsi Boolean f

b. Fungsi Boolean g(w, x, y, z)

wx / yz	**00**	**01**	**11**	**10**
00	0	0	X	1
01	1	X	X	X
11	X	X	X	X
10	X	X	X	X

Tabel 3.3 Peta Karnaugh Fungsi Boolean g

c. Fungsi Boolean h(w, x, y, z)

wx / yz	**00**	**01**	**11**	**10**
00	0	0	1	0
01	0	1	X	1
11	1	X	X	X
10	X	X	X	X

Tabel 3.4 Peta Karnaugh Fungsi Boolean h

Dari hasil penyederhanaan fungsi boolean tersebut, didapatkan hasil :

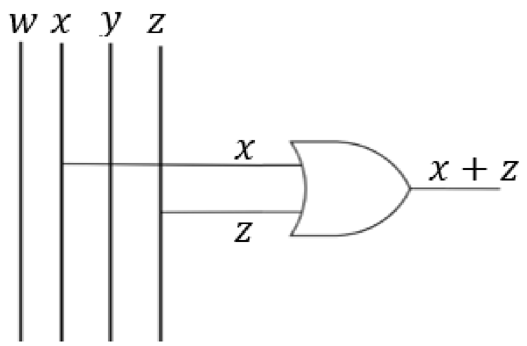
$$f(w,x,y,z) = x + z$$

$$g(w,x,y,z) = x + y$$

$$h(w,x,y,z) = w + xy + xz + yz$$

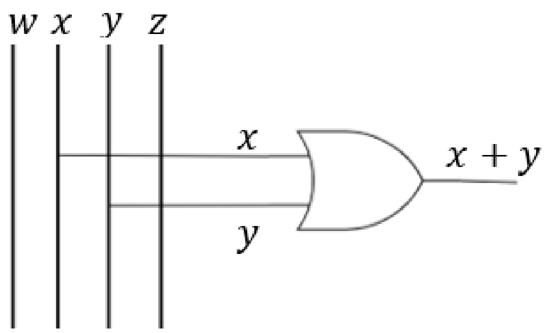
Jika fungsi boolean tersebut direpresentasikan dalam rangkaian logika akan didapatkan rangkaian logika sebagai berikut:

a. Fungsi Boolean f(w, x, y, z)



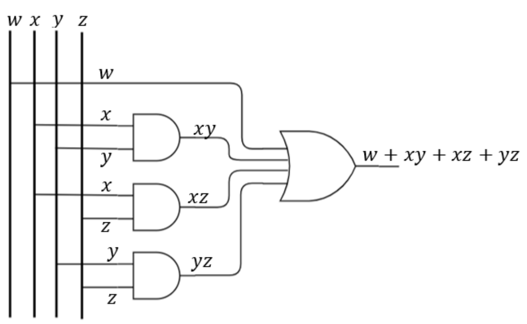
Gambar 3.1 Rangkaian Logika Fungsi Boolean f

b. Fungsi Boolean g(w, x, y, z)



Gambar 3.2 Rangkaian Logika Fungsi Boolean g

c. Fungsi Boolean f(w, x, y, z)



Gambar 3.2 Rangkaian Logika Fungsi Boolean h

IV. Simpulan

Uang sebagai alat pembayaran yang sah secara fisik beredar dalam bentuk uang kertas maupun uang logam atau koin. Walaupun dianggap kurang efisien, terdapat tempat-tempat yang mengharuskan kita melakukan transaksi dengan uang koin, seperti transaksi pada mesin binatu dengan koin, mesin jaja, atau permainan *arcade.* Mesin Penukaran Uang Koin dikembangkan untuk membantu manusia untuk dapat menukarkan uang dengan pecahan yang besar menjadi uang koin. Mesin mula-mula mendeteksi besaran uang yang dimasukkan pengguna, kemudian melakukan perhitungan dan mengeluarkan jumlah uang koin yang sesuai. Desain dari perhitungan jumlah koin ini dapat diimplementasikan dalam bentuk rangkaian logika dengan menerapkan salah satu konsep yang dipelajari pada mata kuliah IF2120 Matematika Diskrit, yaitu Aljabar Boolean.

V. Ucapan Terima Kasih

Pertama-tama, penulis panjatkan puji syukur kepada Tuhan Yang Maha Esa atas limpahan berkah, rahmat, dan karunia-Nya, makalah ini dapat diselesaikan dengan baik. Penulis juga mengucapkan terima kasih kepada Ibu Dr. Nur Ulfa Maulidevi S.T., M.Sc. selaku dosen pengajar mata kuliah Matematika Diskrit K01 yang telah dengan sabar memberikan banyak ilmu serta memberi motivasi selama perkuliahan. Terima kasih pula pada orang tua dan teman-teman yang selalu memberi dukungan dan semangat sehingga penulis dapat menyelesaikan tugas makalah ini dengan lancar. Penulis menyadari bahwa penulis masih jauh dari kata sempurna, sehingga penulis memohon maaf apabila terdapat kesalahan dalam penulisan makalah ini, baik yang disengaja maupun tidak disengaja. Penulis sangat terbuka terhadap masukan, kritikan, dan saran yang bersifat membangun terhadap makalah ini. Akhir kata, penulis berharap makalah ini dapat bermanfaat bagi banyak orang.

VI. Tautan Video

Video penjelasan terkait makalah ini diunggah dalam Youtube "IF2120 Penerapan Aljabar Boolean pada Mesin Penukaran Koin Sederhana" yang dapat diakses melalui tautan berikut :

https://youtu.be/XLT3L8iK_2g

Daftar Pustaka

[1] Aljabar Boolean - Pengertian, Hukum, dan Contoh Soal Aljabar Boolean ~ Studi Elektronika. (n.d.). Diakses tanggal 2 Desember 2022 pukul 18.00, dari https://www.webstudi.site/2019/02/aljabar-boolean.html

[2] Badan Pengembangan dan Pembinaan Bahasa, Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia. (2016). KBBI Daring. Diakses pada 2 Desember 2022 pukul 20.45, dari https://kbbi.kemdikbud.go.id

[3] Bank Indonesia. (n.d.). Glosarium. Diakses tanggal 3 Desember 2022 pukul 20.25, dari https://www.bi.go.id/id/glosarium.aspx

[4] Lipschutz, Seymour, and Marc Lars Lipson. 2007. *Schaum's Outline of Discrete Mathematics*. 3rd ed. New York: McGraw Hill.

[5] Munir, Rinaldi. 2020. *Matematika Diskrit*. 7th ed. Bandung: Informatika Bandung.

[6] Noura Alajmi, Mariam Al-Yehya, Saba AlAsqah, Abdulaziz Al-Dulaimi. 2019. *Coin Exchange Machine.* Electrical and Computer Engineering Department, American University of Kuwait. Diakses tanggal 3 Desember 2022 pukul 22.17, dari https://dspace.auk.edu.kw/handle/11675/8035

[7] Schardijn, Amy, "AN INTRODUCTION TO BOOLEAN ALGEBRAS" (2016). Electronic Theses, Projects, and Dissertations. 421. https://scholarworks.lib.csusb.edu/etd/421

[8] Setiawan, R. (2021, July 30). Gerbang Logika dan Tabel Kebenaran. Dicoding Blog. Diakses pada 2 Desember 2022 pukul 20.00, dari https://www.dicoding.com/blog/gerbang-logika-dan-tabel-kebenaran/

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 2022

Husnia Munzayana
13521077