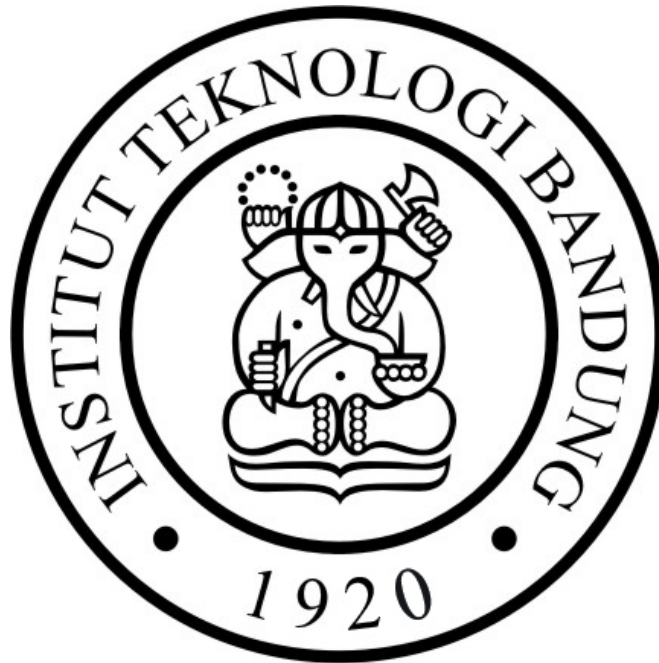


TUGAS BESAR
EL2008 Pemecahan Masalah dengan C
“Eksplorasi Minimisasi Logika”



Disusun oleh:
Fadiah Mumtaz Andevi (18320009)
Tanya Nuhaisy Wulandari (18320017)
Eunike Kristianti (18320019)

Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2022

Daftar Isi

Daftar Isi	2
Laporan Inti	3
Deskripsi Simulasi	3
Flowchart	4
Data Flow Diagram (DFD)	20
Kesimpulan dan Lesson Learned	21
Pembagian Tugas dalam Kelompok	22
Daftar Referensi	23

Laporan Inti

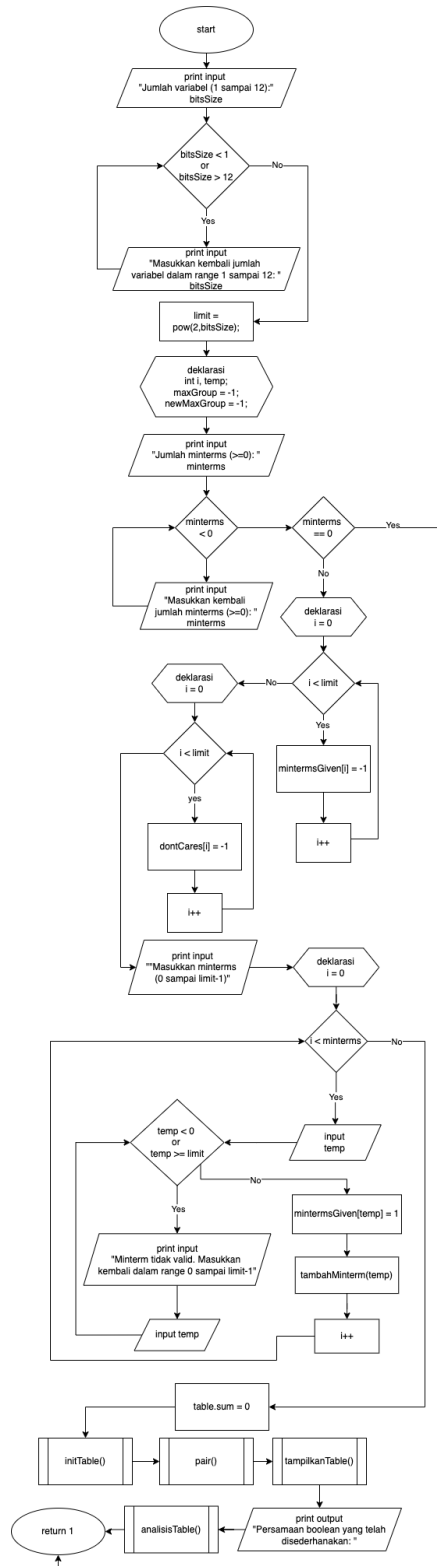
Deskripsi Simulasi

Minimisasi logika merupakan proses untuk menyederhanakan suatu fungsi boolean atau ekspresi algebra [1]. Terdapat tiga cara untuk menyederhanakan fungsi boolean yaitu dengan Teorema Aljabar Boolean, Metode *Karnaugh Map* (*K-Map*), dan dengan Metode Tabular. Setelah melakukan eksplorasi, diputuskan untuk membuat program minimisasi logika menggunakan bahasa C dengan metode tabular.

Program yang akan dibuat adalah sebagai berikut. Program akan menerima input dari pengguna berupa jumlah variabel, jumlah *minterms*, dan *minterms*. Jumlah variabel dari pengguna akan dibatasi dari satu sampai dua belas variabel. Sementara nilai *minterms* dibatasi dari nol sampai dua pangkat jumlah variabel dikurangi satu. Dari ketiga input tersebut, akan dicari *essential prime implicants*-nya yang kemudian akan menjadi solusi sekaligus output dari program ini. Dalam memproses input yang diberikan pengguna, program akan dibagi menjadi 4 langkah. Langkah pertama yaitu program akan mengubah *minterms* dari bentuk desimal menjadi bentuk binernya. Kemudian, bilangan-bilangan biner tersebut akan dikelompokkan ke dalam beberapa grup berdasarkan jumlah angka '1' dalam bilangan binernya. Setelah dikelompokkan, program akan membandingkan seluruh bilangan biner pada grup ke-n dengan seluruh bilangan biner pada grup ke-(n+1). Jika dua bilangan biner yang dibandingkan hanya memiliki satu perbedaan pada nilainya, misal "0001" dengan "0011", maka kedua bilangan biner tersebut akan dianggap sebagai *matched pair* dan satu angka yang berbeda tersebut akan digantikan menggunakan tanda underscore '_', sehingga menjadi "00_1". Itu merupakan langkah kedua. Setelah itu, seluruh *matched pair* dari grup ke-n akan dibandingkan dengan seluruh *matched pair* dari grup ke-(n+1). Jika dua *matched pair* yang dibandingkan hanya memiliki satu perbedaan pada nilainya, maka kedua *matched pair* tersebut akan dianggap *matched* dan satu angka yang berbeda tersebut akan digantikan menggunakan tanda underscore '_'. Itu merupakan langkah ketiga. Proses membandingkan tersebut akan terus dilakukan sampai tidak ditemukan lagi *matched pair*. Kemudian, dari angka yang tetap sama sampai akhir dalam setiap grup, akan ditentukan persamaan aljabar boolean nya untuk kemudian aljabar boolean tersebut akan disebut sebagai *prime implicants*. Pada langkah keempat, akan dibuat tabel *prime implicants* untuk menentukan *essential prime implicants*. *Essential prime implicants* tersebutlah yang kemudian akan ditampilkan menjadi output pada program sebagai *simplified boolean expression* atau persamaan boolean yang telah disederhanakan dari input yang diberikan pengguna. Akan tetapi, untuk mempermudah verifikasi hasil, maka setiap proses yang dilakukan, yang pada program disebut sebagai iterasi, akan ditampilkan juga sebagai output, termasuk dengan tabel *prime implicants* nya.

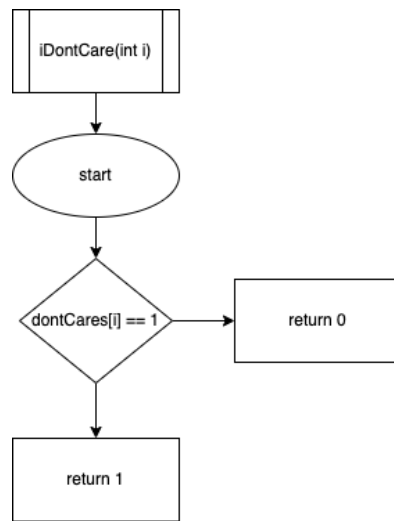
Flowchart

1. Main Program



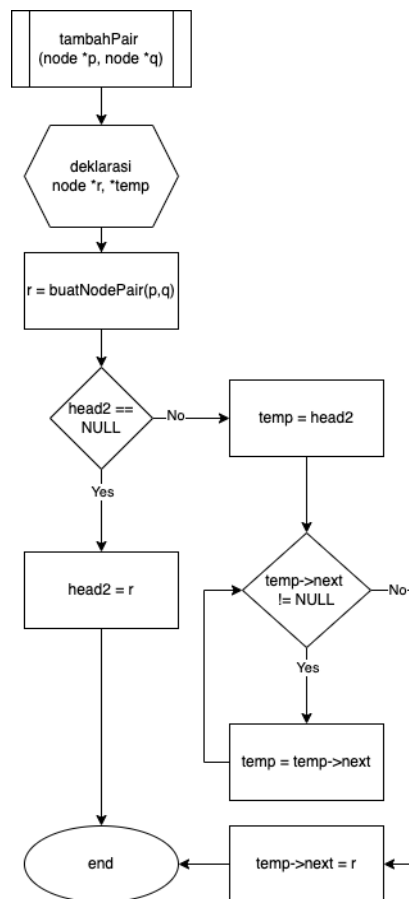
Gambar 2.1 Flowchart Main Program

2. Fungsi ifDontCare



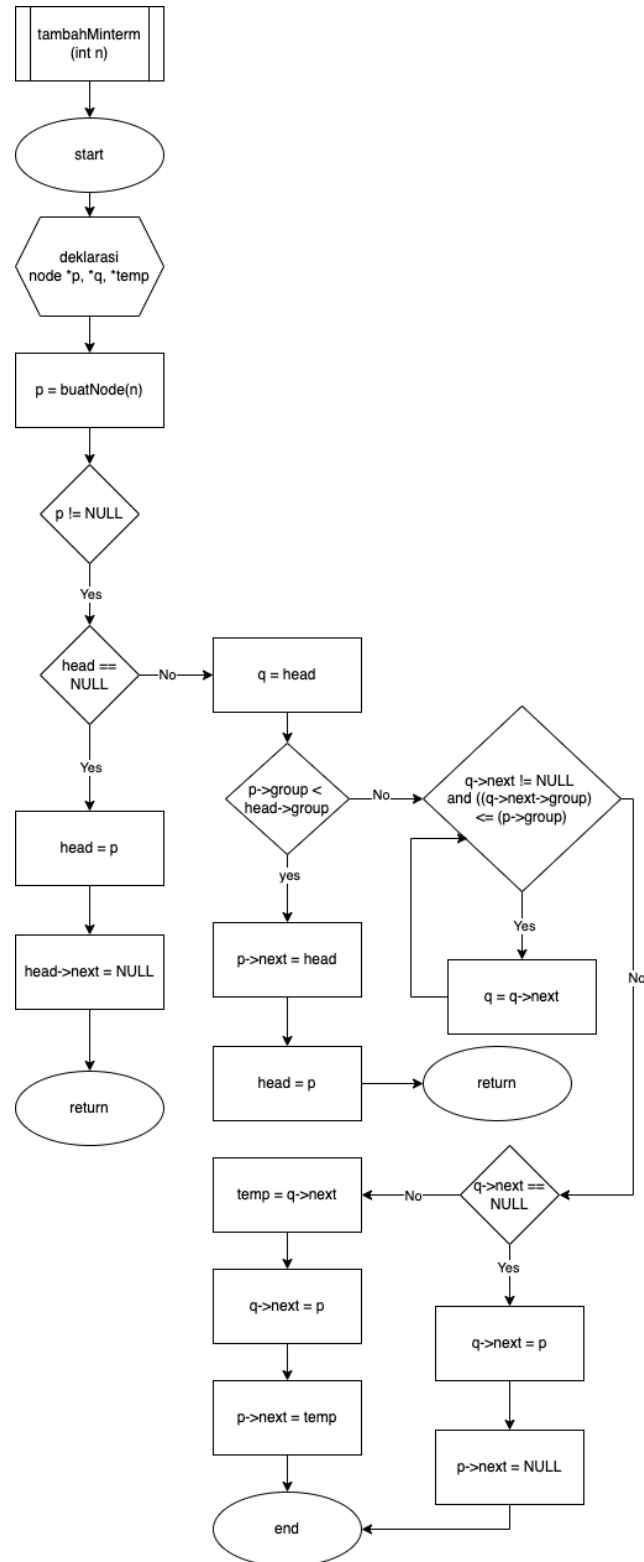
Gambar 2.2 Flowchart Fungsi ifDontCare

3. Fungsi tambahPair



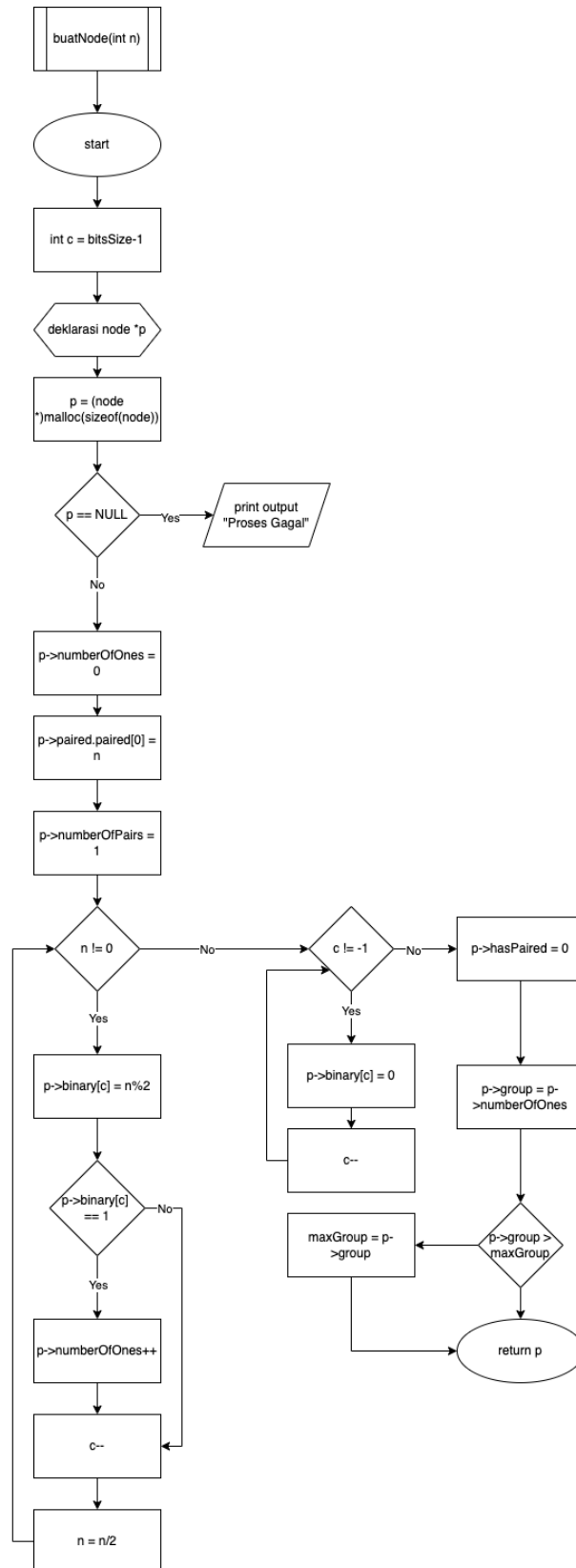
Gambar 2.3 Flowchart Fungsi tambahPair

4. Fungsi tambahMinterm



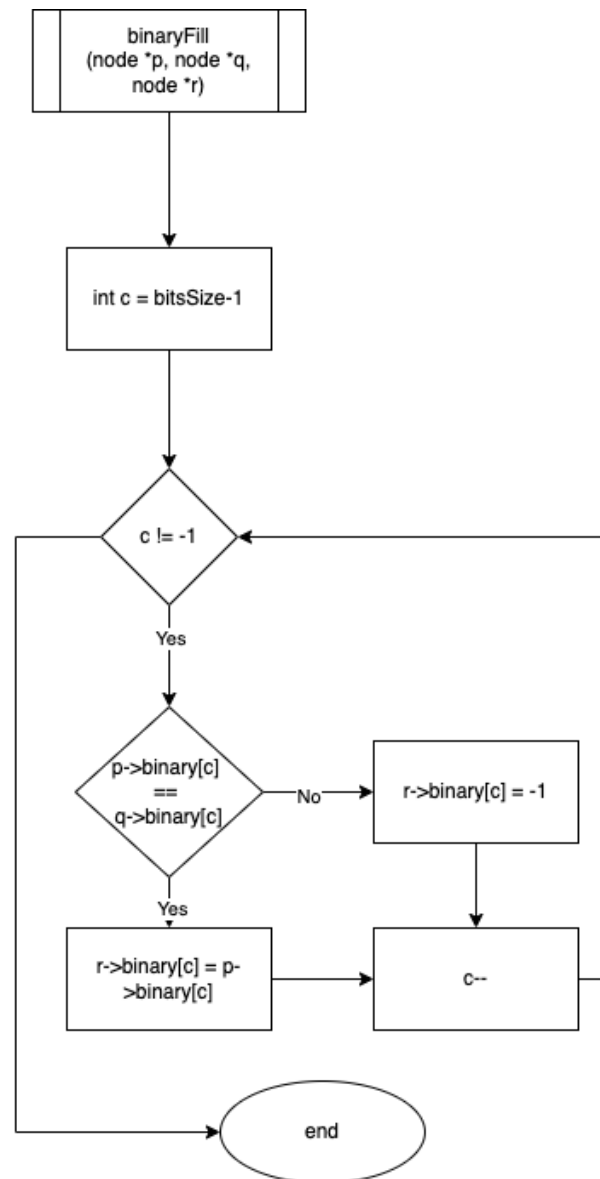
Gambar 2.4 Flowchart Fungsi tambahMinterm

5. Fungsi buatNodePair



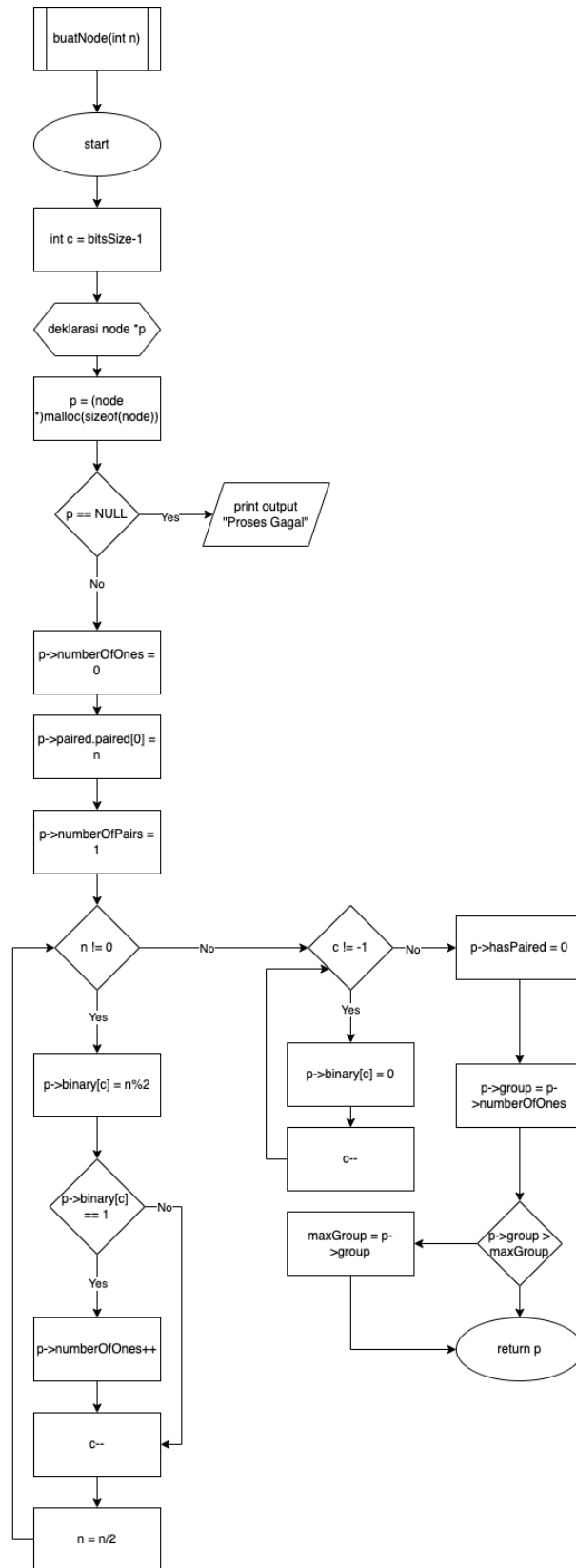
Gambar 2.5 Flowchart Fungsi buatNode

6. Fungsi binaryFill



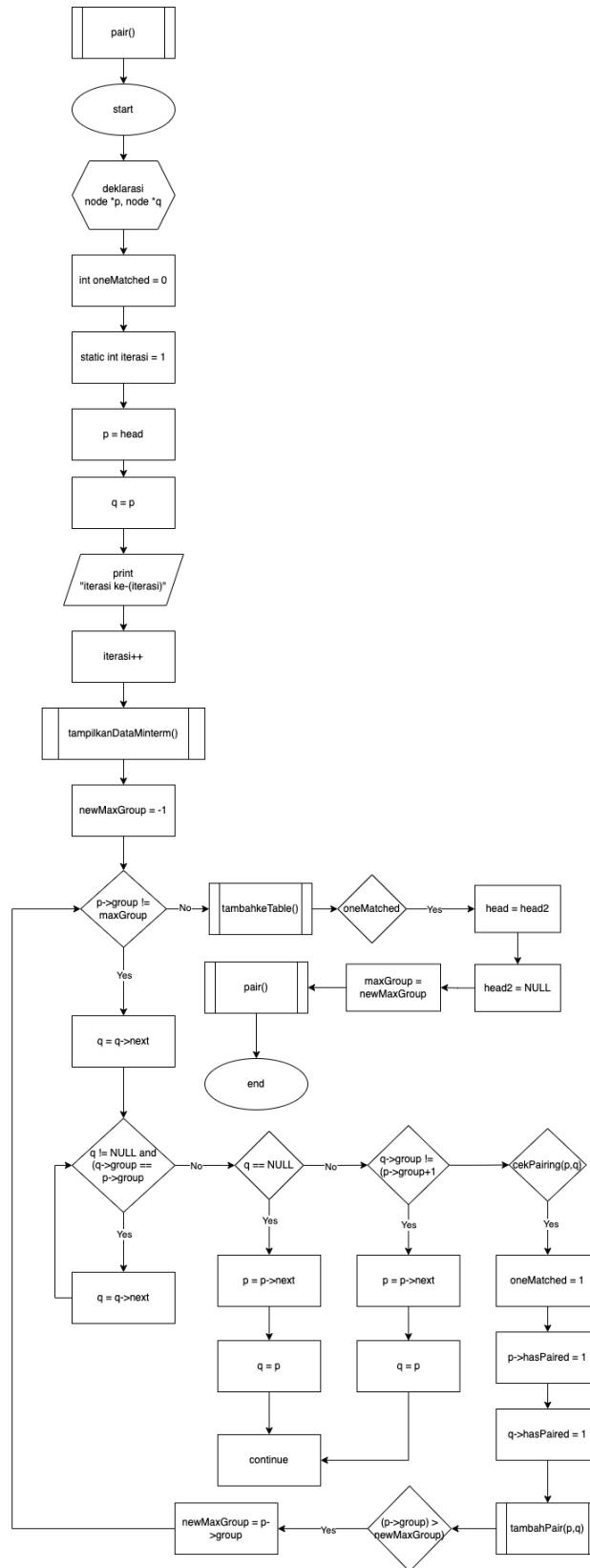
Gambar 2.6 Flowchart Fungsi binaryFill

7. Fungsi buatNodePair



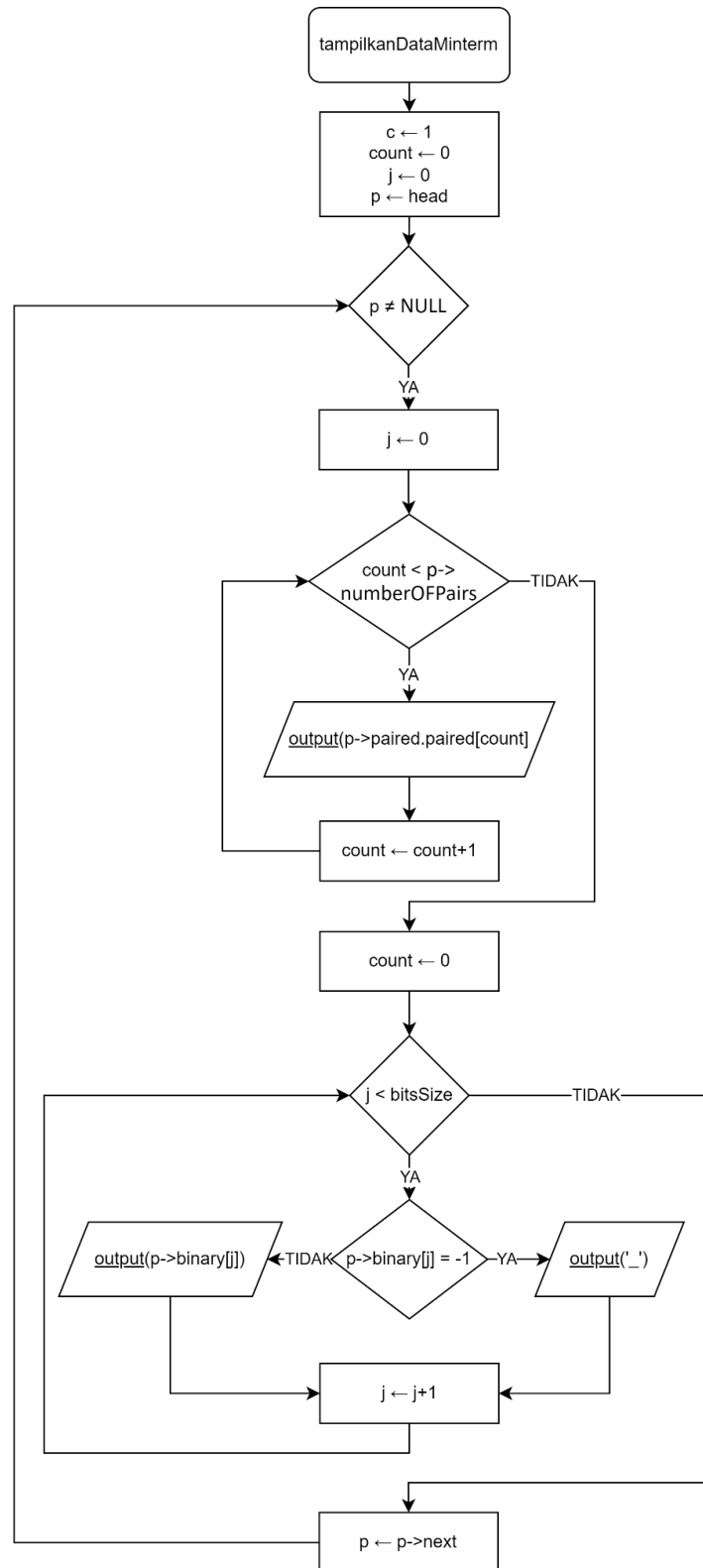
Gambar 2.7 Flowchart Fungsi buatNodePair

8. Fungsi pair



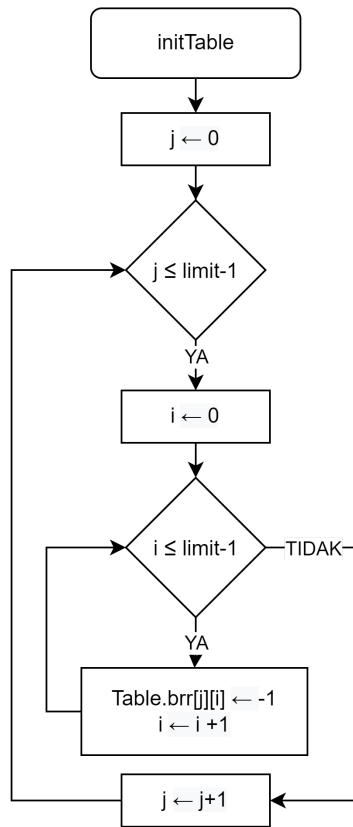
Gambar 2.8 Flowchart Fungsi pair

9. Fungsi tampilkanDataMinterm



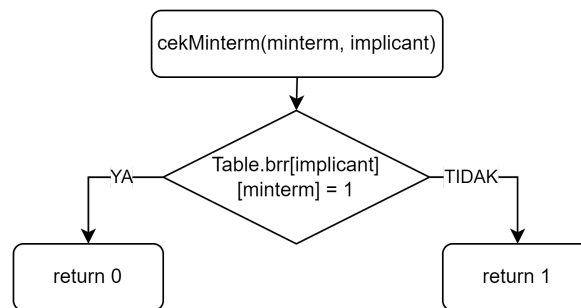
Gambar 2.9 Flowchart Fungsi tampilkanDataMinterm

10. Fungsi initTable



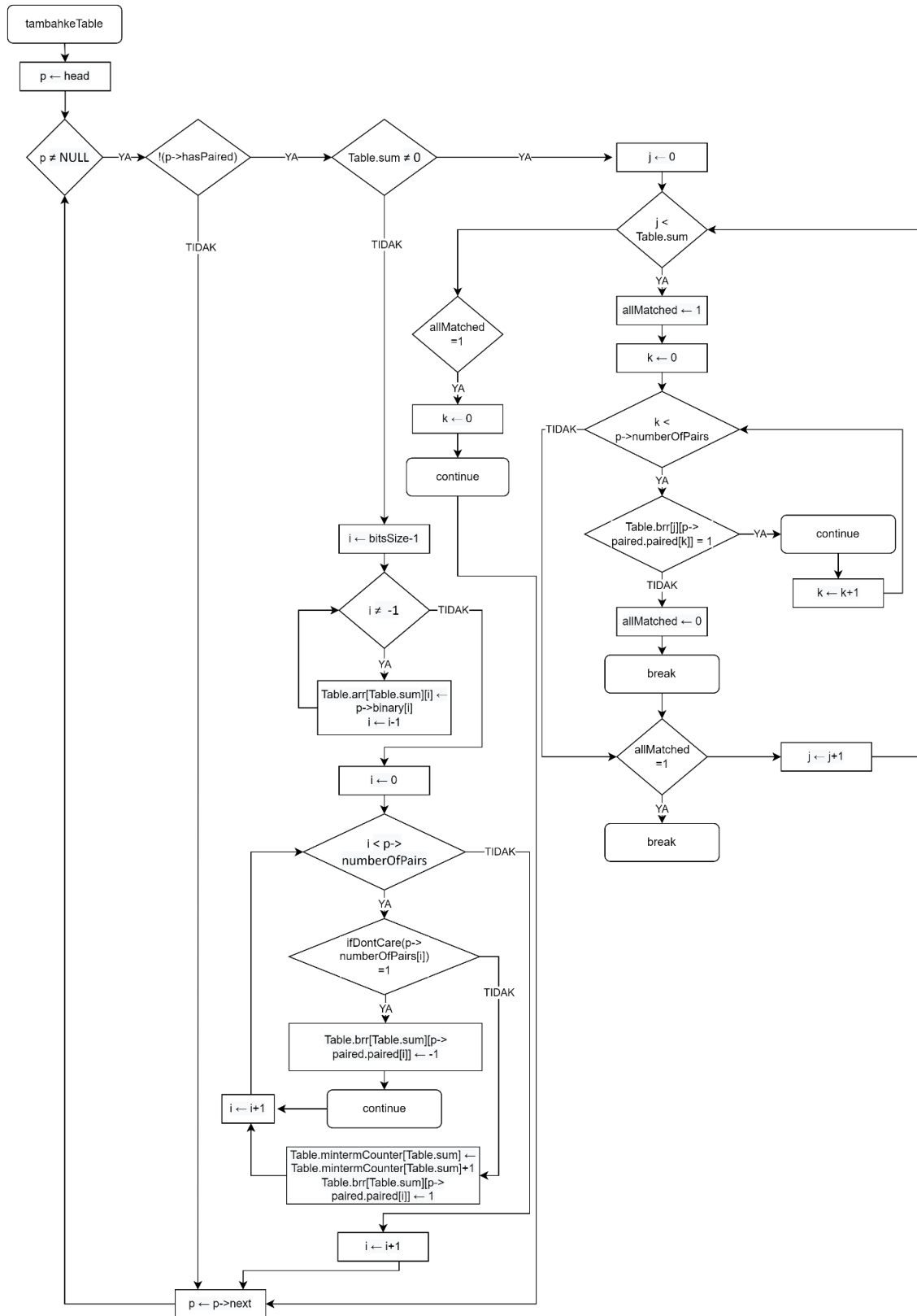
Gambar 2.10 Flowchart Fungsi initTable

11. Fungsi cekMinterm



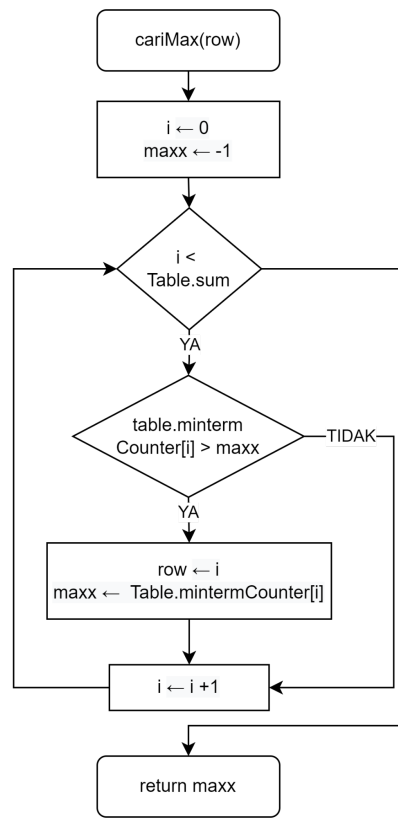
Gambar 2.11 Flowchart Fungsi cekMinterm

12. Fungsi TambahkeTable



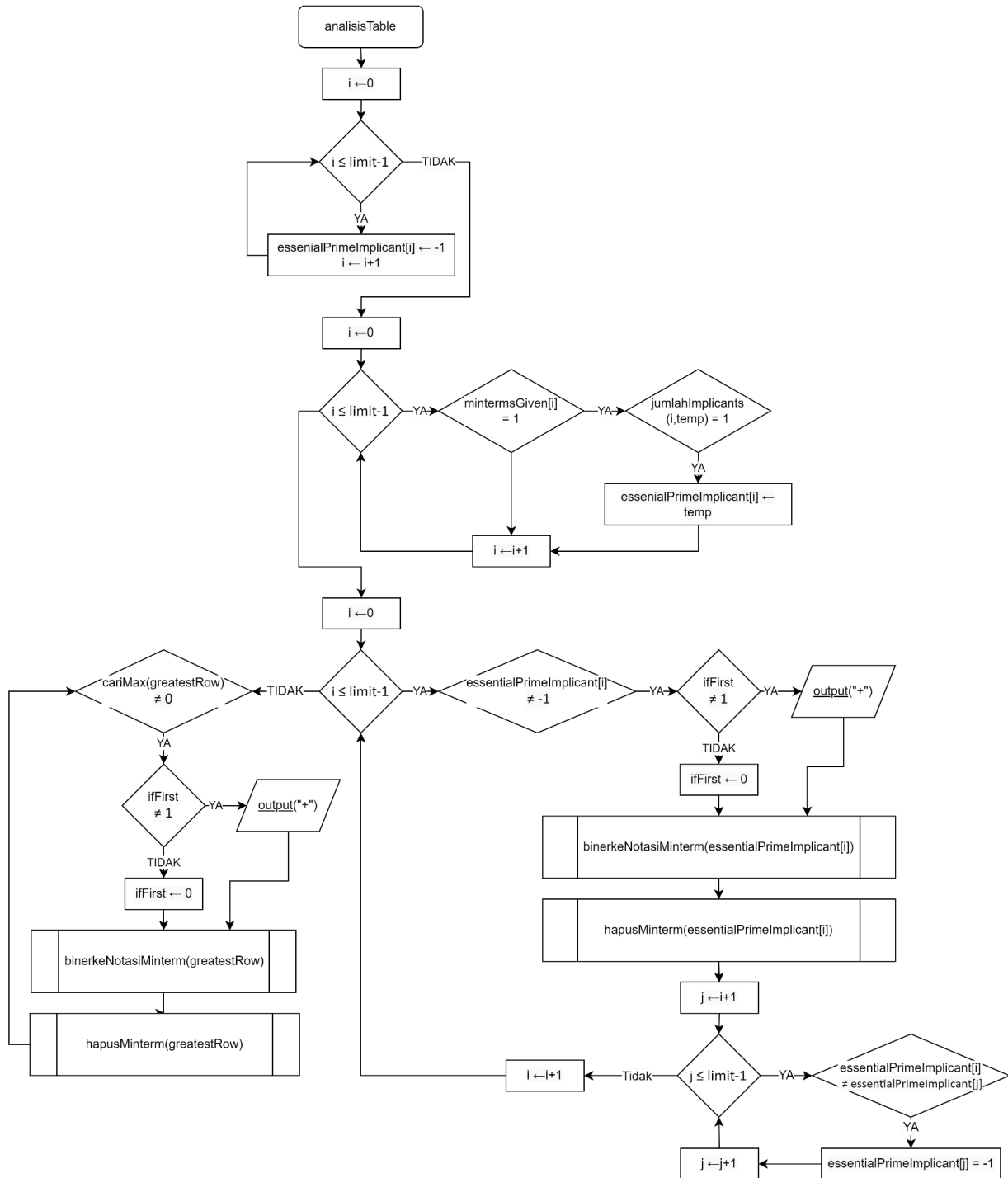
Gambar 2.12 Flowchart Fungsi TambahkeTable

13. Fungsi cariMax



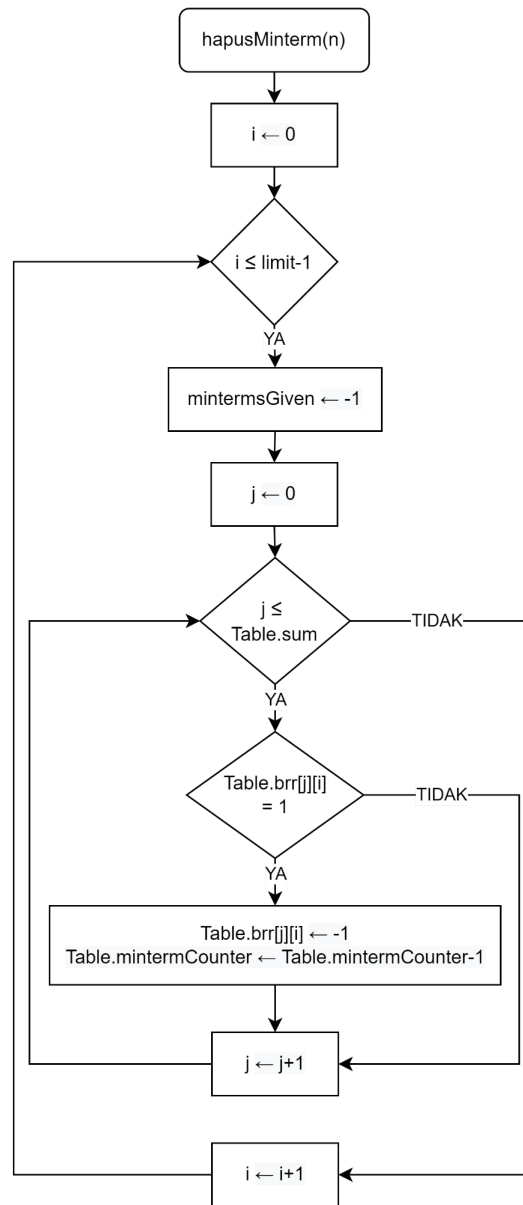
Gambar 2.13 Flowchart Fungsi cariMax

14. Fungsi analisisTable



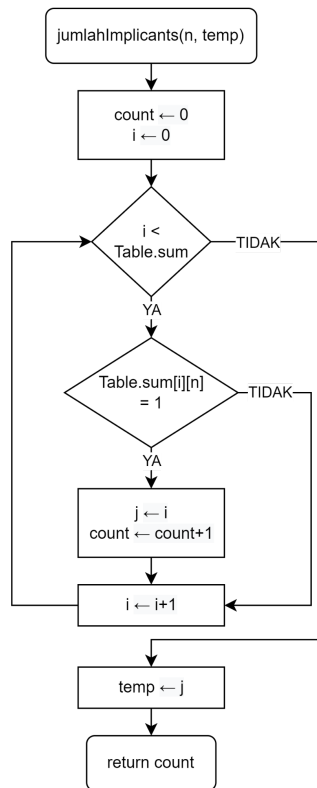
Gambar 2.14 Flowchart Fungsi analisisTable

15. Fungsi hapusMinterm



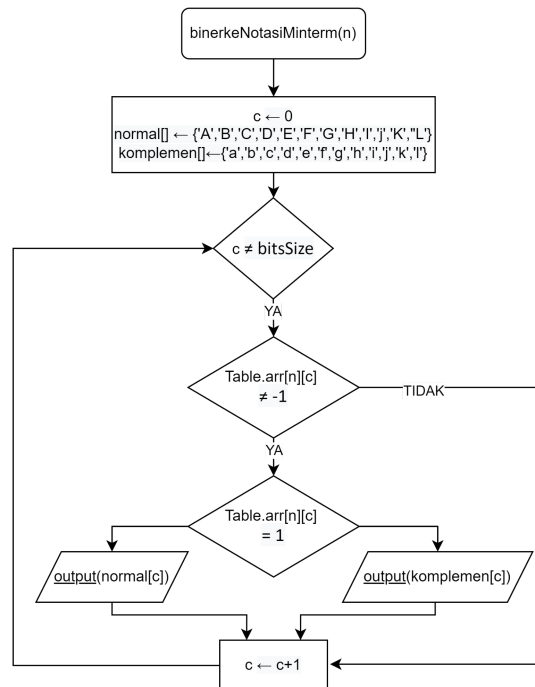
Gambar 2.15 Flowchart Fungsi hapusMinterm

16. Fungsi jumlahImplicants



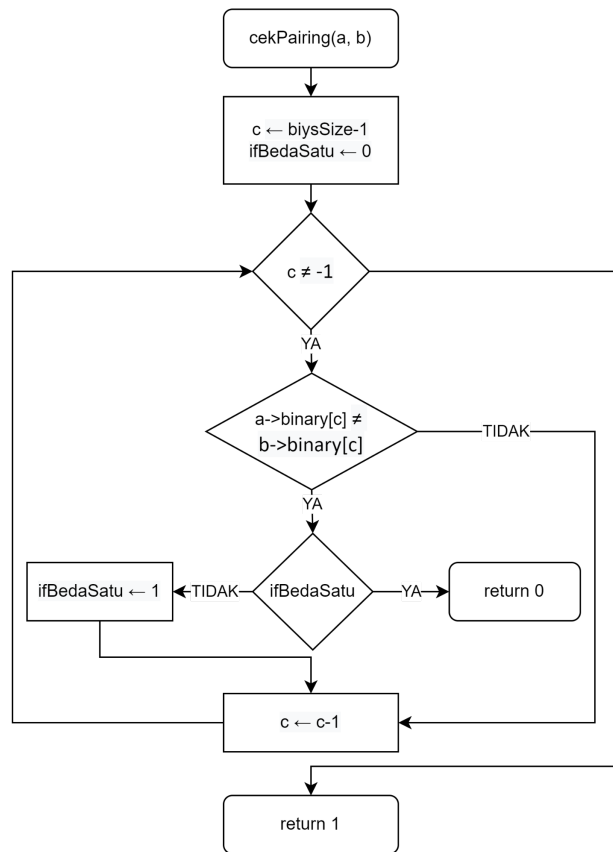
Gambar 2.16 Flowchart Fungsi jumlahImplicants

17. Fungsi binerkeNotasiMinterm



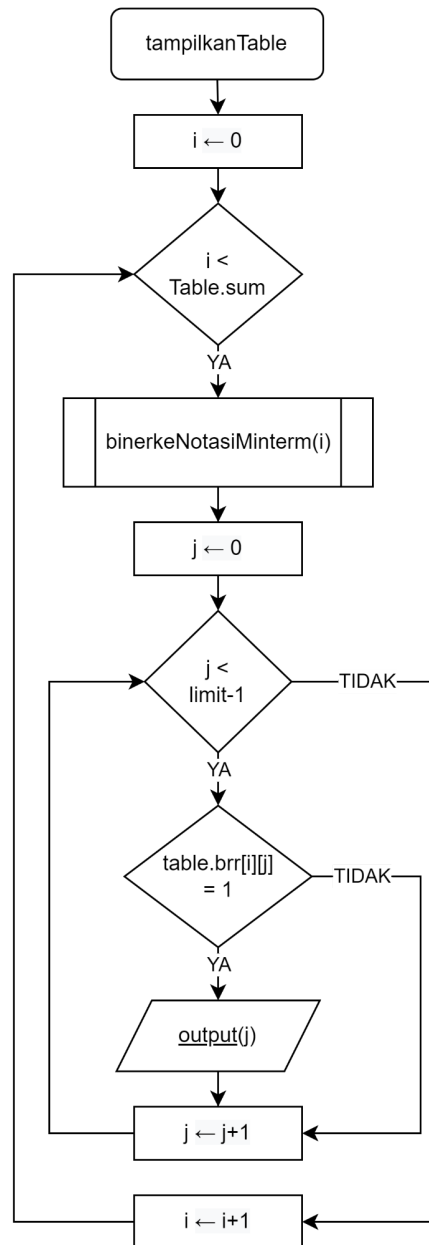
Gambar 2.17 Flowchart Fungsi binerkeNotasiMinterm

18. Fungsi cekPairing



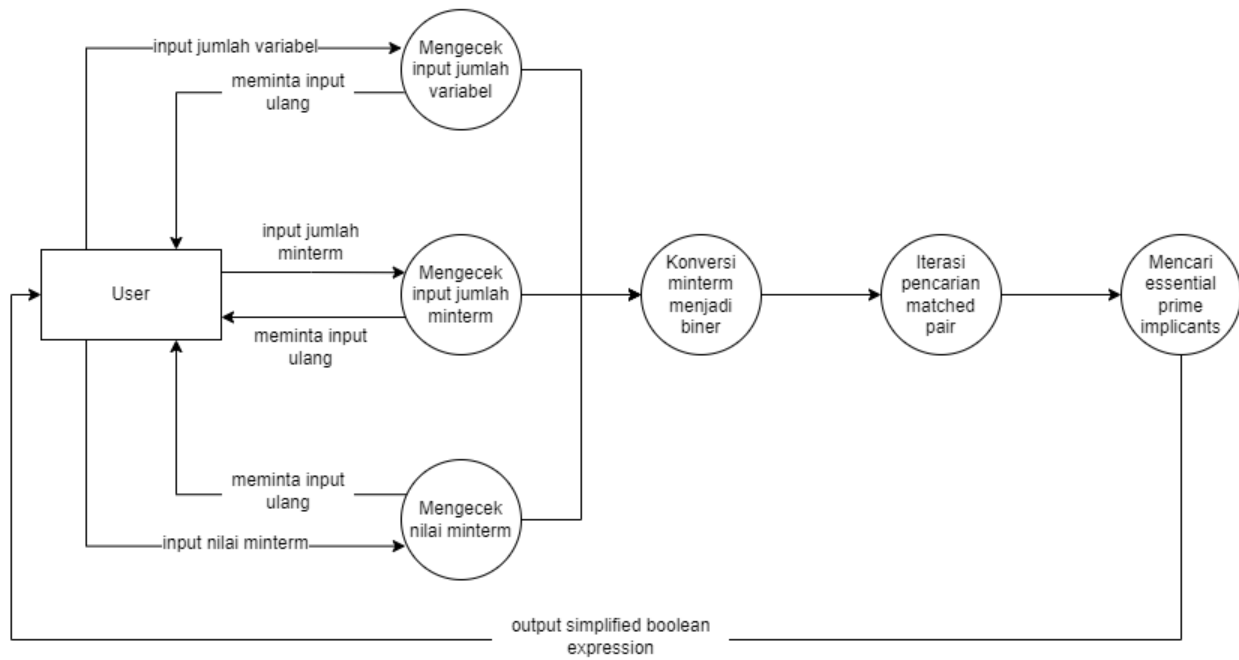
Gambar 2.18 Flowchart Fungsi cekPairing

19. Fungsi tampilkanTable



Gambar 2.19 Flowchart Fungsi tampilkanTable

Data Flow Diagram (DFD)



Gambar 2.20 Data Flow Diagram program

Kesimpulan dan Lesson Learned

Metode Tabular merupakan salah satu metode yang digunakan untuk menyederhanakan suatu fungsi boolean. Dari ketiga metode penyederhanaan fungsi boolean, metode ini dapat digunakan untuk fungsi dengan jumlah variabel yang cukup banyak dibanding dengan metode lainnya. Metode Tabular dapat diimplementasikan dalam pemrograman bahasa C. Program yang telah dibuat menerima input berupa jumlah variabel, jumlah minterm dan minterm sedangkan untuk output yaitu berupa tabel *prime implicants* dan fungsi boolean yang telah disederhanakan. Jumlah variabel untuk program ini sendiri sudah dibatasi hanya hingga 12 variabel. Dari pengujian yang telah dilakukan, output yang didapatkan sudah sesuai dengan output yang diharapkan. Salah satu kekurangan dari program ini adalah program menerima input berupa minterm. Sehingga, dari fungsi boolean yang akan disederhanakan, perlu dikonversi secara manual terlebih dahulu kedalam bentuk mintermnya.

Pembagian Tugas dalam Kelompok

Fadiyah Mumtaz Andevi (18320009)

- Pembuatan Source Code
- Deskripsi Simulasi
- Flowchart
- Kesimpulan dan Lesson Learned

Tanya Nuhaisy Wulandari (18320017)

- Pembuatan Source Code
- Deskripsi Simulasi
- Flowchart
- Kesimpulan dan Lesson Learned

Eunike Kristianti (18320019)

- Pembuatan Source Code
- Deskripsi Simulasi
- Data Flow Diagram (DFD)
- PPT

Daftar Referensi

- [1] <https://www.geeksforgeeks.org/minimization-of-boolean-functions/>, 24 April 2022, 15.30.
- [2] <http://freesourcecode.net/cprojects/102643/sourcecode/McQuicksy.c#.YlGuFchBw2w>, 23 April 2022, 20.00.