**LAPORAN PEMPBUATAN APLIKASI**

**“Konversi Desimal ke Biner”**

Diajukan untuk memenuhi salah satu tugas akhir mata kuliah Dasar Sistem Komputer yang diampu oleh :

Ali Tarmuji, S.T., M.Cs.

****

**DISUSUN OLEH :**

**NAMA : MUJADDID FATHI ATHO’ILLAH**

**NIM : 2200018385**

**KELAS : I**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS AHMAD DAHLAN**

**2022/2023**

# **KATA PENGANTAR**

Alhamdulillah, segala puji dan syukur bagi Allah Subhanahu wa Ta’ala yang telah melimpahkan kenikmatan serta keberkahan waktu, tenaga, dan pikiran sehingga saya dapat menyusun Laporan Pembuatan Aplikasi “Konversi Desimal ke Biner” tepat pada waktunya.

Tidak lupa saya sampaikan ucapan terima kasih kepada Bapak Ali Tarmuji, S.T., M.Cs. selaku dosen pembimbing yang telah membimbing kami dalam mengerjakan tugas laporan ini.

Saya menyadari bahwa masih ada kekurangan pada penulisan laporan ini. Oleh karena itu, kritik dan saran yang membangun saya harapkan dari para pembaca sekalian.

Akhir kata, Semoga laporan yang saya tulis ini dapat memberikan manfaat bagi para pembaca khususnya untuk menambah wawasan tentang dasar sistem komputer dan pemrograman assembly.

Yogyakarta, 2023

Mujaddid Fathi Atho’illah

# **DAFTAR ISI**

[**KATA PENGANTAR** i](#_Toc123472920)

[**DAFTAR ISI** ii](#_Toc123472921)

[**BAB I: DESKRIPSI APLIKASI** 1](#_Toc123472922)

[**1.1.** **Deskripsi Umum** 1](#_Toc123472923)

[**1.2.** **Flowchart** 1](#_Toc123472924)

[**BAB II: DESAIN ANTARMUKA DAN KODE PROGRAM** 3](#_Toc123472925)

[**2.1.** **Sketsa Antarmuka** 3](#_Toc123472927)

[**2.2.** **Kode Program** 3](#_Toc123472928)

[**2.3.** **Penjelasan Kode Program** 6](#_Toc123472929)

[**BAB III: PERCOBAAN PROGRAM** 10](#_Toc123472930)

[**BAB IV: PENUTUP** 14](#_Toc123472932)

# **BAB I**

**DESKRIPSI APLIKASI**

## **Deskripsi Umum**

Aplikasi “Konversi Desimal ke Biner” adalah aplikasi yang membantu pengguna untuk mengubah bilangan desimal ke bilangan biner. Aplikasi ini dibuat menggunakan bahasa assembly (TASM) dan memiliki fitur utama yaitu mengkonversi bilangan desimal ke bilangan biner.

Pengguna dapat mengakses aplikasi ini melalui browser web atau dengan mengunduh aplikasi ini ke perangkat dan mengakses aplikasi ini melalui DOSBOX atau aplikasi emulator/simulator pemrograman assembly seperti EMU8086, Microprocessor Simulator, dsb.

Aplikasi “Konversi Desimal ke Biner” ini sangat berguna bagi pengguna yang sering bekerja dengan bilangan biner, seperti dalam bidang kelistrikan atau teknik komputer. Dengan aplikasi ini, pengguna dapat dengan mudah mengubah bilangan desimal ke bilangan biner tanpa perlu menghitung secara manual.

## **Flowchart**

Berikut ini adalah cara kerja/alur program dari aplikasi “Konversi Desimal ke Biner”:

1. Mulai
2. Masukkan bilangan desimal yang ingin dikonversi
3. Hitung sisa pembagian bilangan tersebut dengan 2
4. Jika sisa pembagian adalah 0, simpan 0 sebagai bagian terakhir dari bilangan biner yang dihasilkan
5. Jika sisa pembagian adalah 1, simpan 1 sebagai bagian terakhir dari bilangan biner yang dihasilkan
6. Bagikan bilangan desimal dengan 2
7. Ulangi langkah 3-6 sampai bilangan desimal sama dengan 0
8. Cetak hasil konversi bilangan biner
9. Selesai

Algoritma flow chart program “Konversi Desimal ke Biner” adalah sebagai berikut:



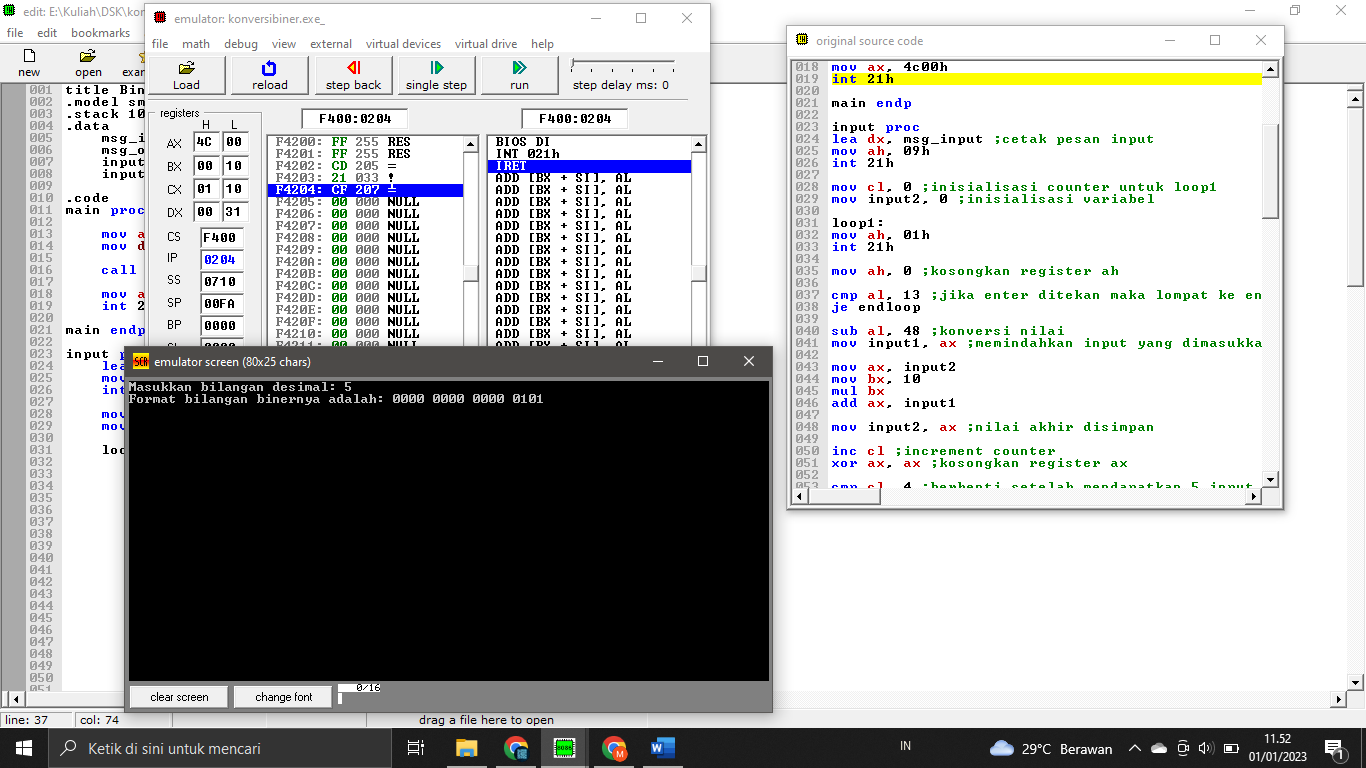
# **BAB II**

**DESAIN ANTARMUKA DAN KODE PROGRAM**



## **Sketsa Antarmuka**

Aplikasi “Konversi Desimal ke Biner” memiliki antarmuka yang sederhana dan mudah digunakan. Pengguna hanya memasukkan bilangan desimal yang ingin dikonversikan pada tempat yang disediakan . Setelah menekan tombol “enter” pada keyboard aplikasi akan menghitung konversi bilangan tersebut dan menampilkan hasil konversi ke dalam format bilangan biner.



## **Kode Program**

title BinaryConverter

.model small

.stack 100h

.data

msg\_input db "Masukkan bilangan desimal : ", "$"

msg\_output db "Format bilangan binernya adalah : ", "$"

input1 dw ?

input2 dw ?

.code

main proc

mov ax, @data

mov ds,ax

call input ;memanggil fungsi input

mov ax, 4c00h

int 21h

main endp

input proc

lea dx, msg\_input ;cetak pesan input

mov ah, 09h

int 21h

mov cl, 0 ;inisialisasi counter untuk loop1

mov input2, 0 ;inisialisasi variabel

loop1:

mov ah, 01h

int 21h

mov ah, 0 ;kosongkan register ah

cmp al, 13 ;jika enter ditekan maka lompat ke endloop

je endloop

sub al, 48 ;konversi nilai

mov input1, ax ;memindahkan input yang dimasukkan ke input variabel

mov ax, input2

mov bx, 10

mul bx

add ax, input1

mov input2, ax ;nilai akhir disimpan

inc cl ;increment counter

xor ax, ax ;kosongkan register ax

cmp cl, 4 ;berhenti setelah mendapatkan 5 input

jle loop1

xor bx,bx ;kosongkan register bx

xor ax,ax;kosongkan register ax

endloop:

clc ;kosongkan carry flag

mov cl, 0 ;reset counter

mov bl, 4 ;mengisi register bl dengan 4

mov dl, 0ah ;cetak baris baru

mov ah, 02h

int 21h

lea dx, msg\_output ;cetak pesan output

mov ah, 09h

int 21h

loop2:

cmp cl, bl ;jika penghitung tidak sama dengan bl (yang nilai awalnya 3) lompat ke nospace

jne nospace

mov dl, ' ' ;cetak spasi

mov ah, 02h

int 21h

add bl, 4 ;tambahkan 4 ke register bl

nospace:

shl input2, 1 ;bit bergeser ke kiri untuk menempatkan bit paling sesuai ke carriage flag

jc else\_block ;jc bernilai benar jika cf = 1

mov dl, '0' ;jika jc bernilai salah (tidak lompat ke else block) cetak 0

mov ah, 02h

int 21h

jmp cont ;akhiri perkondisian

else\_block:

mov dl, '1' ;cetak 1

mov ah, 02h

int 21h

cont:

clc ;kosongkan carry flag

inc cl ;tambahkan counter dengan 1

cmp cl, 15 ;setelah 16 kali loop, loop akan berhenti

jle loop2

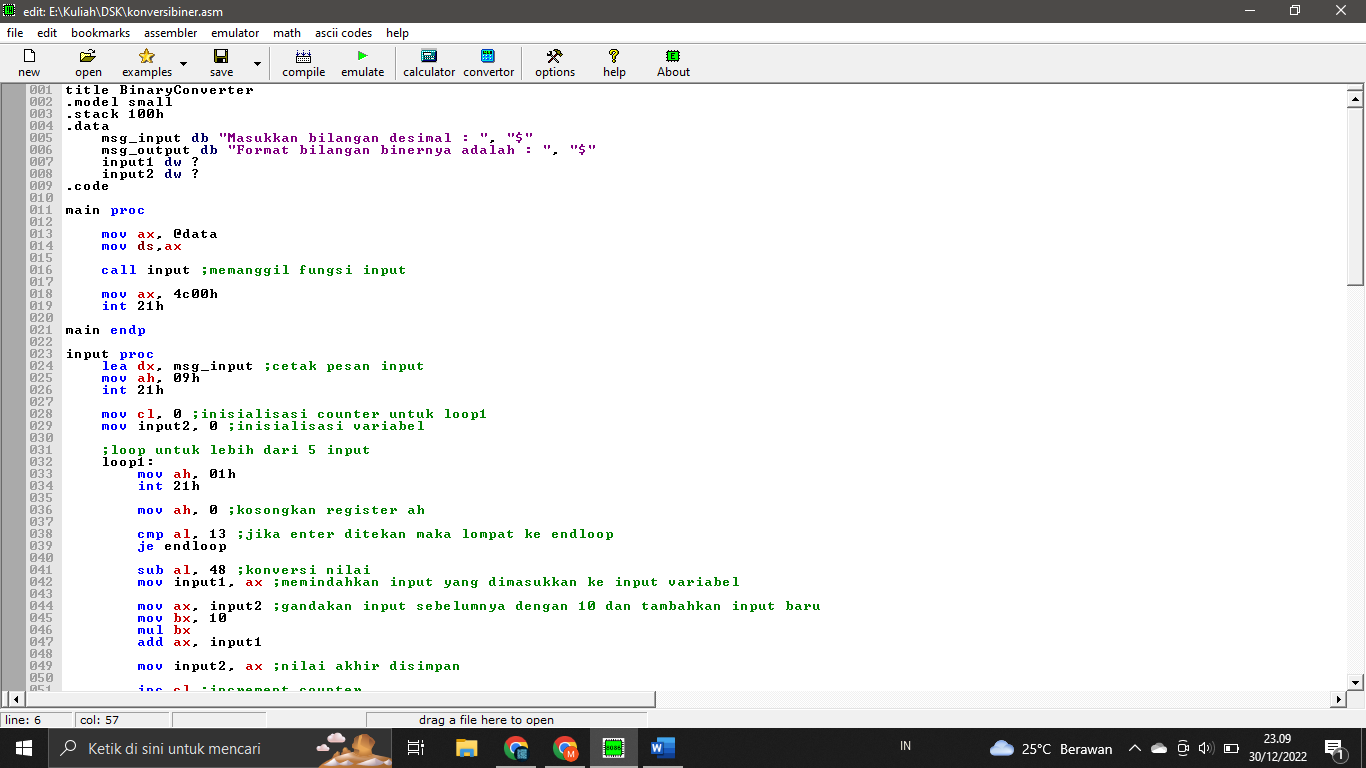
ret

input endp ;akhiri fungsi input

end main

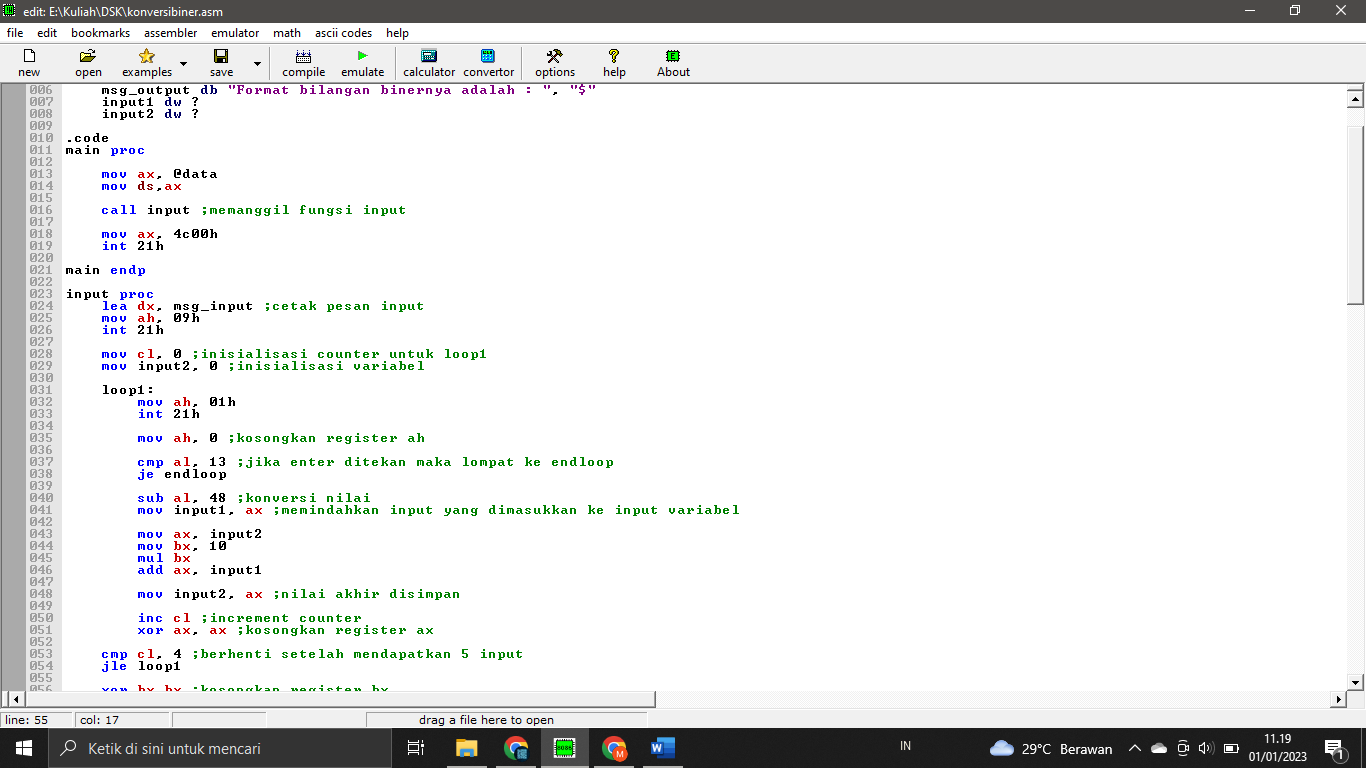
## **Penjelasan Kode Program**

Pesan yang ingin dicetak dideklarasikan terlebih dahulu ke dalam data berikut variabel-variabel yang akan digunakan dalam kode program



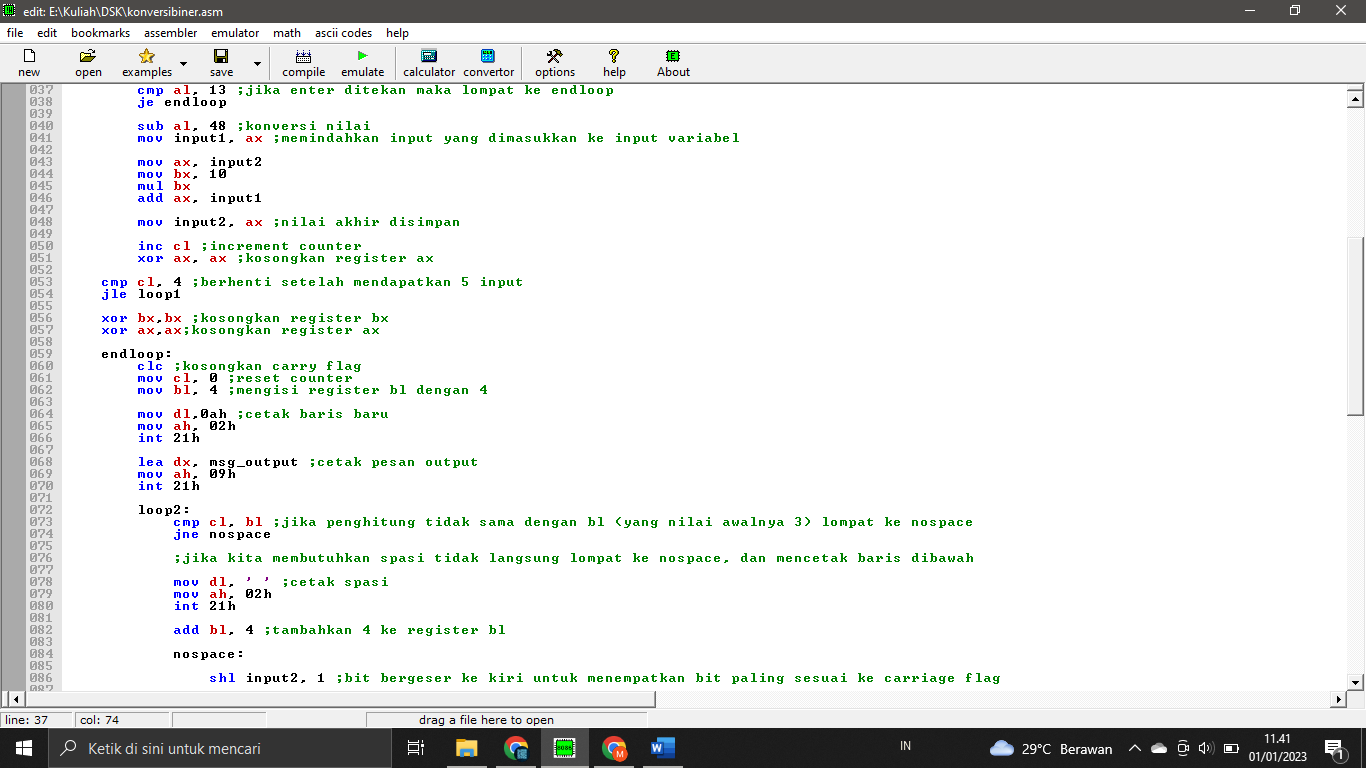
* Pesan untuk menginputkan bilangan desimal terdapat dalam variabel msg\_input
* Pesan untuk mencetak hasil proses konversi terdapat dalam variabel msg\_output
* Variabel input1 dan input2 akan digunakan untuk membantu proses penyimpanan nilai

Pertama-tama pesan input akan ditampilkan terlebih dahulu supaya pengguna tahu bahwa pengguna harus memasukkan bilangan yang ingin dikonversi. Kemudian bilangan yang dimasukkan pengguna akan disimpan ke dalam variabel input2.



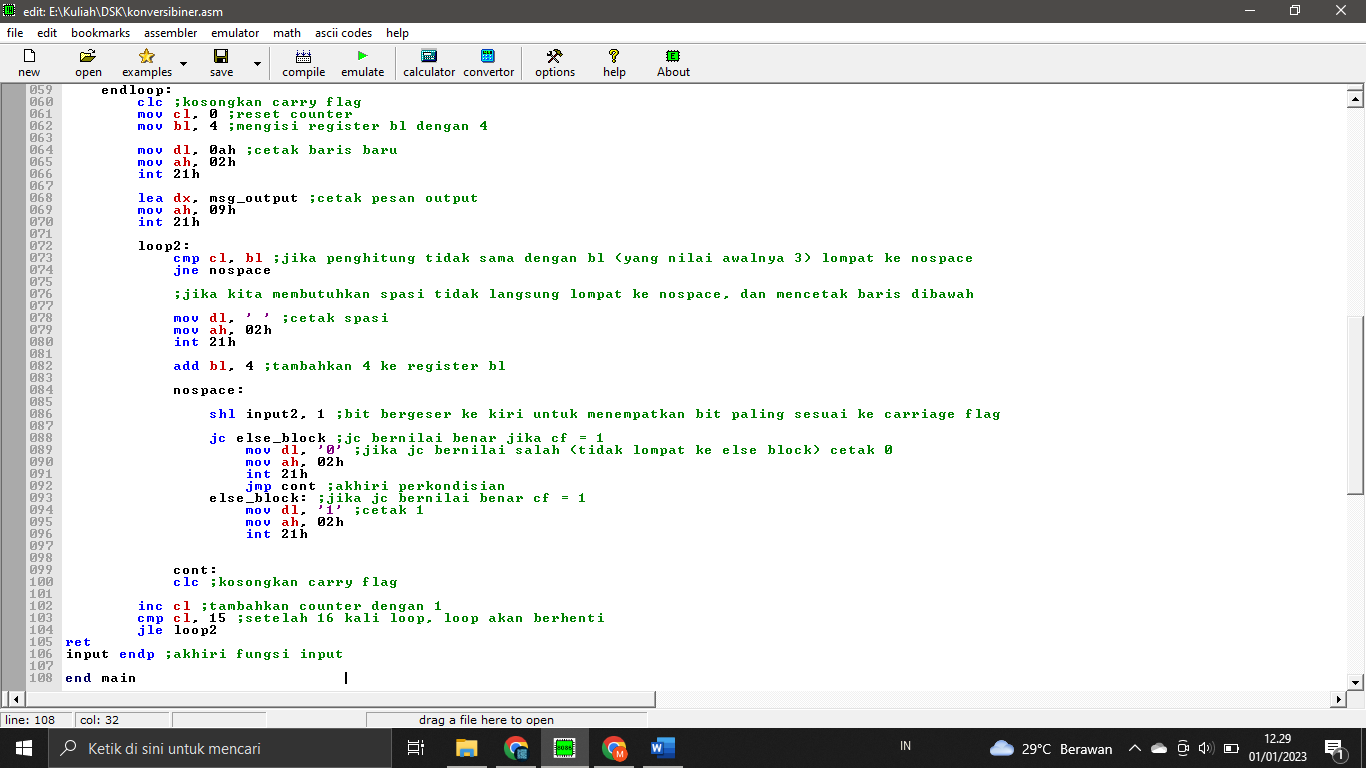
* Semua proses akan dilakukan dalam input proc.
* Menggunakan lea dx, msg\_input untuk mencetak pesan input yang telah dideklarasikan di atas
* Menggunakan mov ah, 09h untuk memasukkan nilai 09h ke dalam register ah supaya pesan dapat dicetak
* Menggunakan int 21h untuk menjalankan program di atasnya, yaitu mencetak pesan input
* Kemudian menggunakan looping untuk proses penginputan nilai, terlebih dahulu penghitung (counter) diinisialisasi dengan 0 dengan mov cl, 0
* Menggunakan mov input2, 0 untuk menginisalisasi bahwa nilai awal input2 adalah 0. Kemudian input2 akan digunakan untuk menyimpan nilai yang diinputkan pengguna
* Untuk proses penginputan nilai menggunakan loop1. Menggunakan mov ah, 01h supaya pengguna dapat memasukkan satu karakter nilai. Setelah intu menggunakan int 21h untuk mengeksekusinya. Kemudian register ah dikosongkan kembali dengan mov ah, 0
* Kemudian menggunakan cmp al, 13 untuk membandingkan, jika pengguna menekan tombol enter, maka langsung lompat ke endloop, yaitu baris kode ketika pengguna sudah selesai menginput nilai yang ingin dikonversikan
* Kemudian nilai yang dimasukkan akan disimpan dalam variabel input2 dan input1
* Kemudian menambahkan nilai penghitung (counter) dengan inc cl, untuk mengetahui berapa banyak angka yang diinput
* Menggunakan cl, 4 untuk membatasi nilai yang dimasukkan, yaitu nilai yang dimasukkan tidak boleh lebih dari 5 karakter. Jika belum sampai 5 karakter, maka akan kembali ke loop1 untuk mengulang proses penginputan nilai

Setelah proses penginputan selesai, maka nilai yang diinputkan akan diproses. Sebelum proses utama dilakukan, terlebih dahulu program akan menampilkan pesan output.



* Menggunakan mov cl, 0 untuk menginisialisasi ulang penghitung (counter) supaya dapat digunakan untuk membantu proses konversi
* Menggunakan mov dl, 0ah untuk mencetak baris baru di bawah pesan input
* Kemudian menggunakan mov ah, 02h dan int21 untuk mengeksekusi kode tersebut
* Menggunakan lea dx, msg\_output untuk memanggil pesan output
* Menggunakan mov ah, 09h untuk mencetak pesan tersebut
* Menggunakan int 21h untuk mengeksekusi baris kode dia atasnya yaitu mencetak pesan output

Setelah itu program akan melakukan proses konversi nilai yang diinputkan pengguna ke dalam format bilangan binernya dan menampilkannya ke dalam layar.



* Menggunakan perulangan dalam loop2 untuk mencetak format bilangan biner sebanyak 20 digit.
* Membandingkan nilai cl dan bl menggunakan cmp cl, bl. Jika penghitung (counter) tidak sama dengan register bl yang nilai awalnya adalah 3, maka lompat ke baris nospace. Jika sama maka baris kode dibawahnya akan menambahkan spasi.
* Pada baris nospace, shl input2, 1 supaya bit bergeser ke kiri untuk menempatkan bit paling sesuai ke carriage flag.
* Jika cf (carriage flag) bernilai 1, maka lompat ke baris else\_block untuk mencetak angka 1.
* Jika cf (carriage flag) tidak sama dengan 1 maka baris di bawahnya akan mencetak angka 0.
* Setelah pencetakan karakter 0/1 maka cf akan kembali dikosongkan.
* Penghitung (counter) akan ditambahkan 1 dengan inc cl. Dan akan dibandingkan dengan cmp, 15. Jika nilai cl tidak sama dengan 15, maka loop2 akan dilakukan kembali hingga cl bernilai 15.

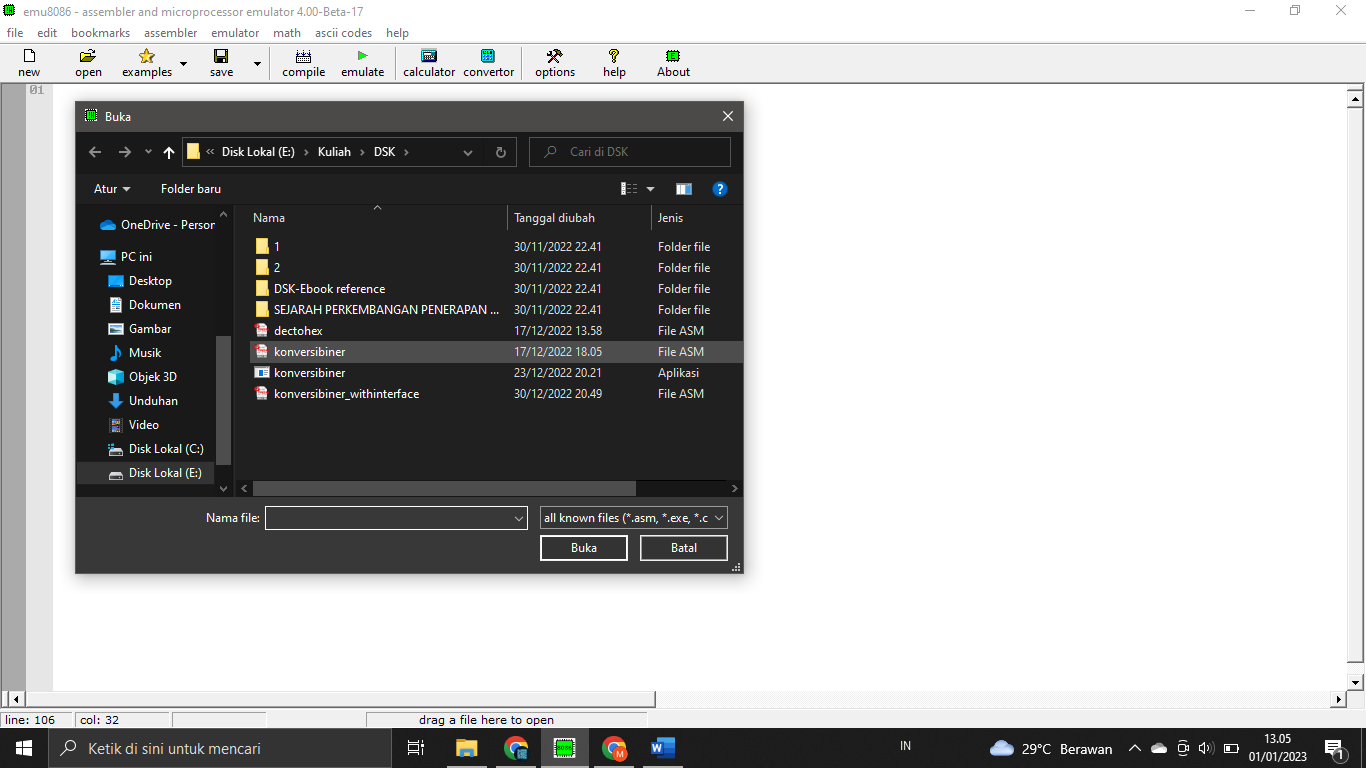
# **BAB III**

**PERCOBAAN PROGRAM**

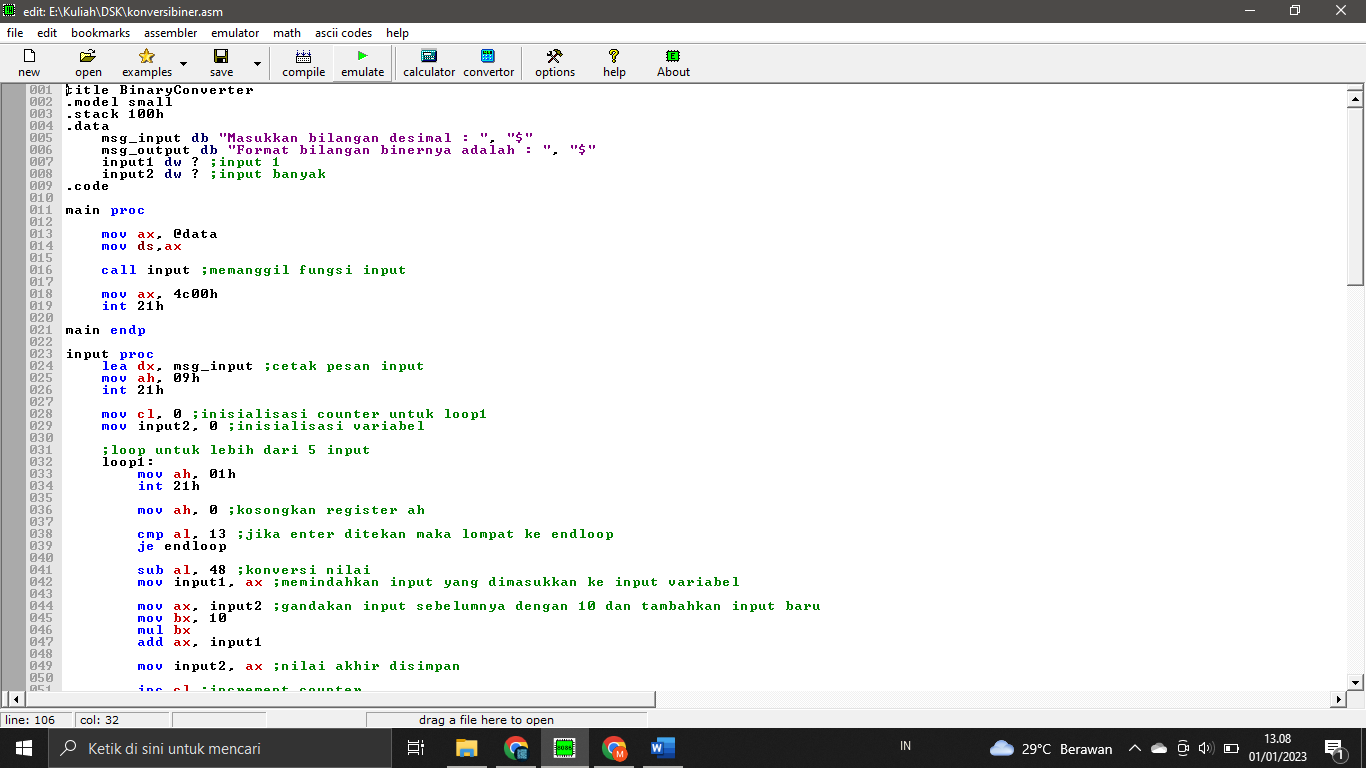


Program dapat dijalankan dengan berbagai aplikasi/media seperti yang sudah saya sebutkan di atas. Dalam percobaan ini, saya menggunakan aplikasi EMU8086. Langkah-langkahnya adalah sebagai berikut.

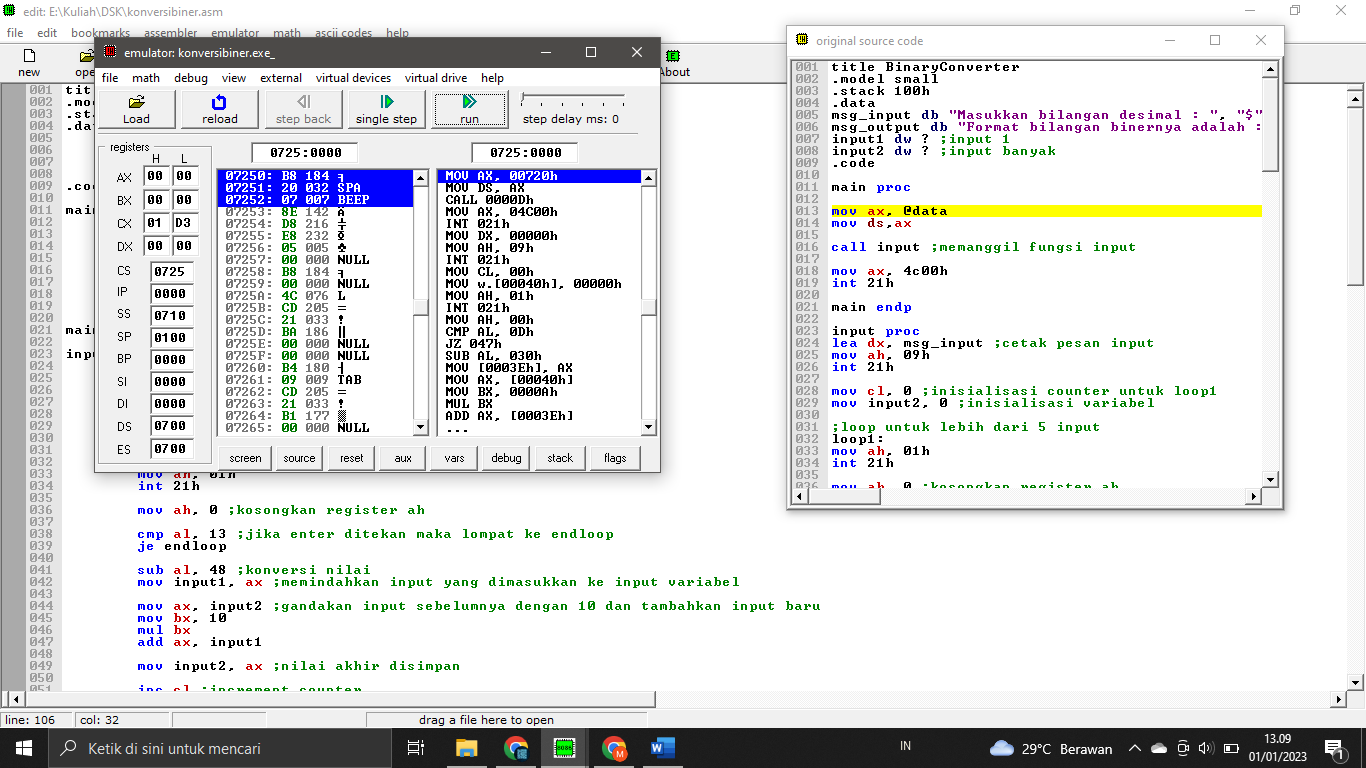
1. Membuka file konversibiner.asm.



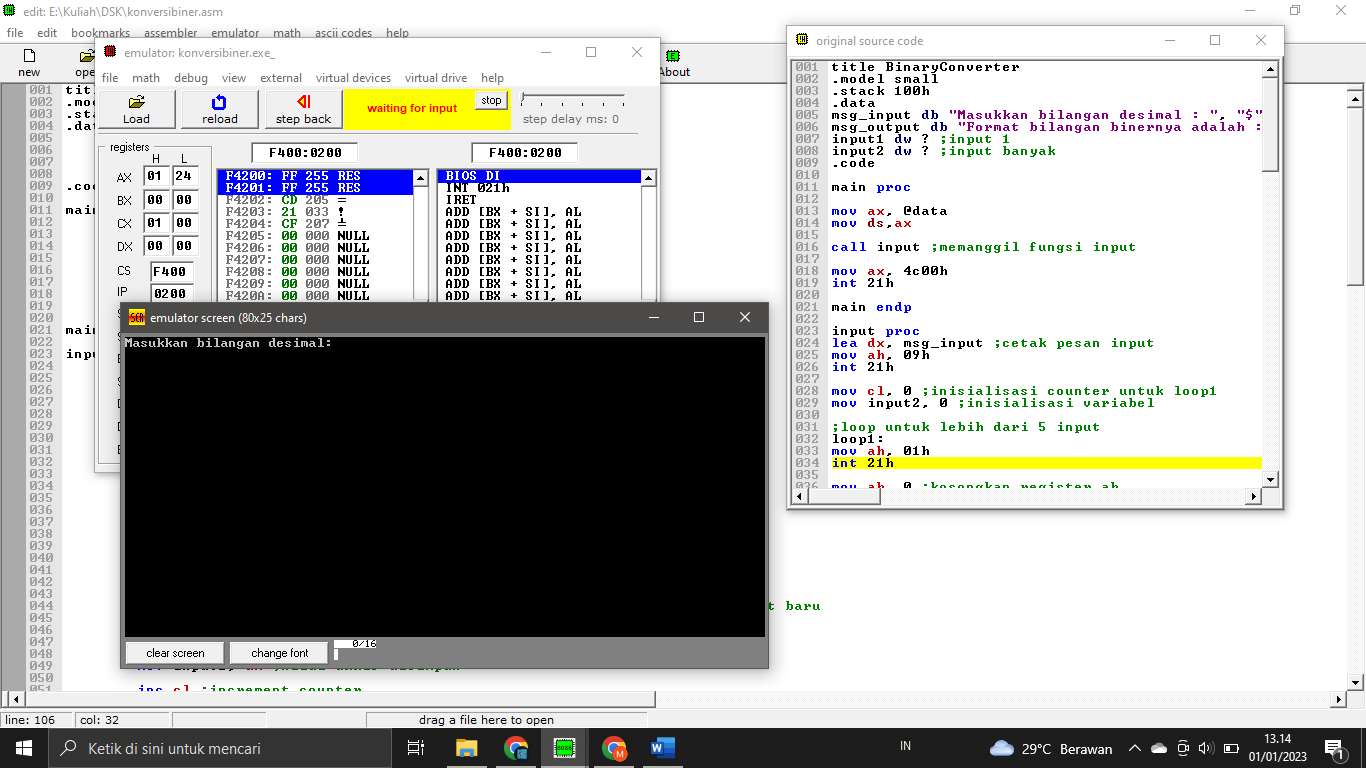
1. Setelah program terbuka, klik emulate pada menu program maka akan tampil jendela emulator.



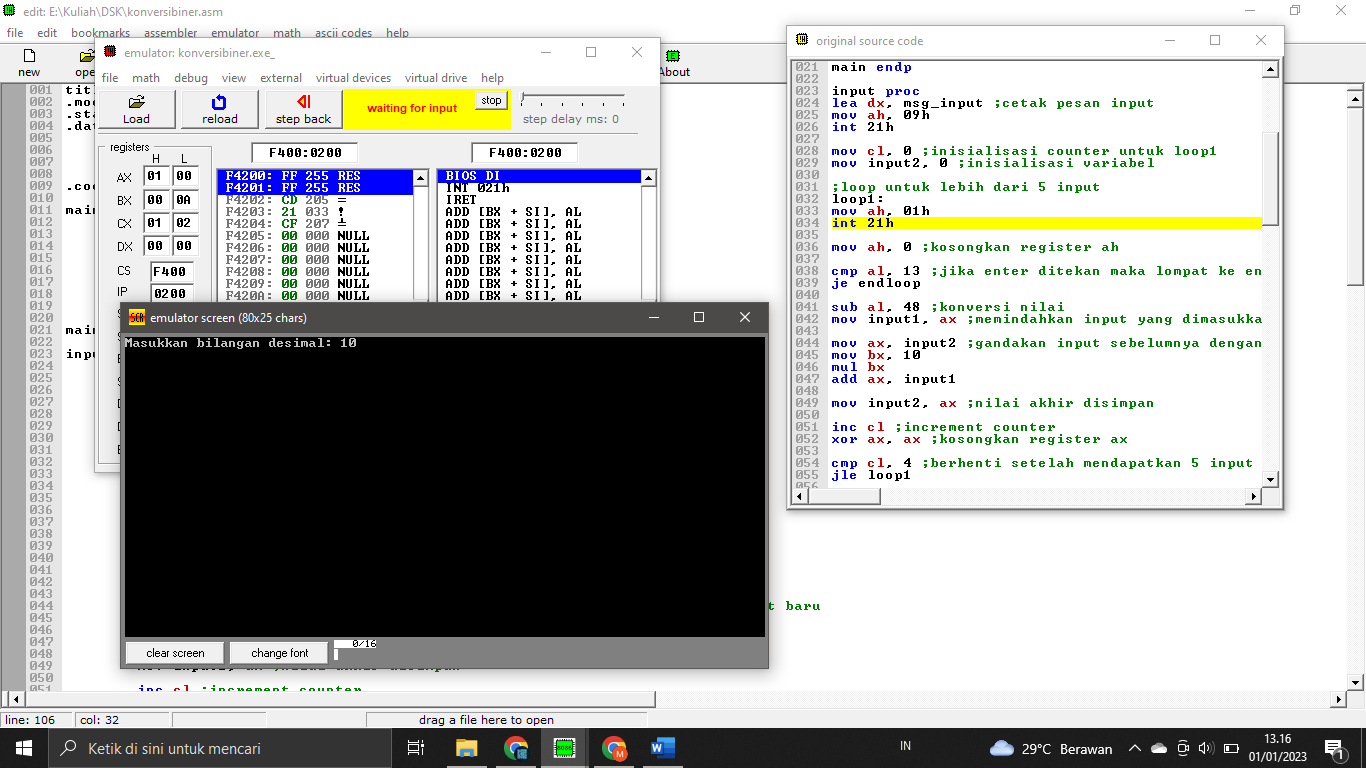
1. Setelah jendela emulator terbuka, klik run untuk menjalankan program.



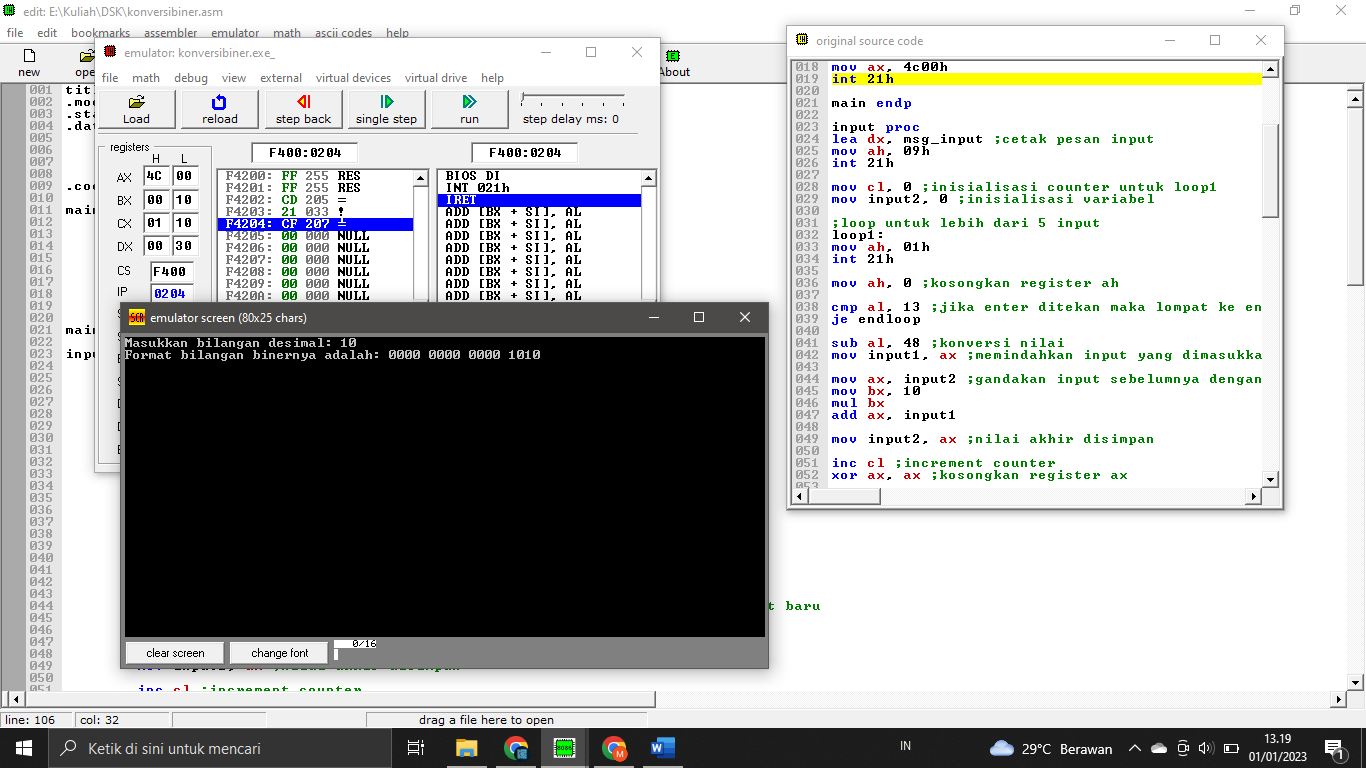
1. Kemudian akan tampil jendela output yang akan menampilkan pesan untuk menginputkan bilangan yang ingin dikonversi.



1. Masukkan bilangan yang ingin dikonversikan (maksimal 5 digit), kemudian tekan enter.



1. Setelah enter, maka hasil dari konversi bilangan akan diproses oleh program dan ditampilkan ke layar dalam format bilangan biner seperti berikut.



1. Program selesai.

# **BAB IV**

**PENUTUP**

Setelah melakukan proses analisis dan uji coba terhadap program “Konversi Desmal ke Biner” yang telah saya buat, dapat disimpulkan bahwa program tersebut dapat berjalan dengan baik. Selama proses uji coba, program tersebut dapat memproses input yang diberikan pengguna dengan tepat, serta menampilkan hasil akhir yang sesuai aturan konversi bilangan desimal ke bilangan biner. Hal ini menunjukkan bahwa program tersebut sudah dapat berjalan dengan baik dan sesuai fungsi yang diinginkan.

Meskipun program “Konversi Desimal ke Biner” yang saya buat dapat berfungsi sesuai yang diharapkan, saya menyadari terdapat kekurangan yang perlu disempurnakan untuk menghindari berbagai macam kesalahan/bug yang mungkin terjadi. Dengan demikian, di masa yang akan datang perlu dilakukan perbaikan-perbaikan pada program ini agar dapat memberikan pengalaman yang memuaskan bagi pengguna.

Saya berharap aplikasi ini dapat bermanfaat bagi pengguna dan terus memberikan performa yang baik dalam jangka panjang. Selain itu, saya juga terbuka untuk menerima saran dan masukan dari pengguna untuk meningkatkan aplikasi ini di masa yang akan datang.