# PRAKTIKUM SISTEM OPERASI MODUL 2



**Disusun oleh :**

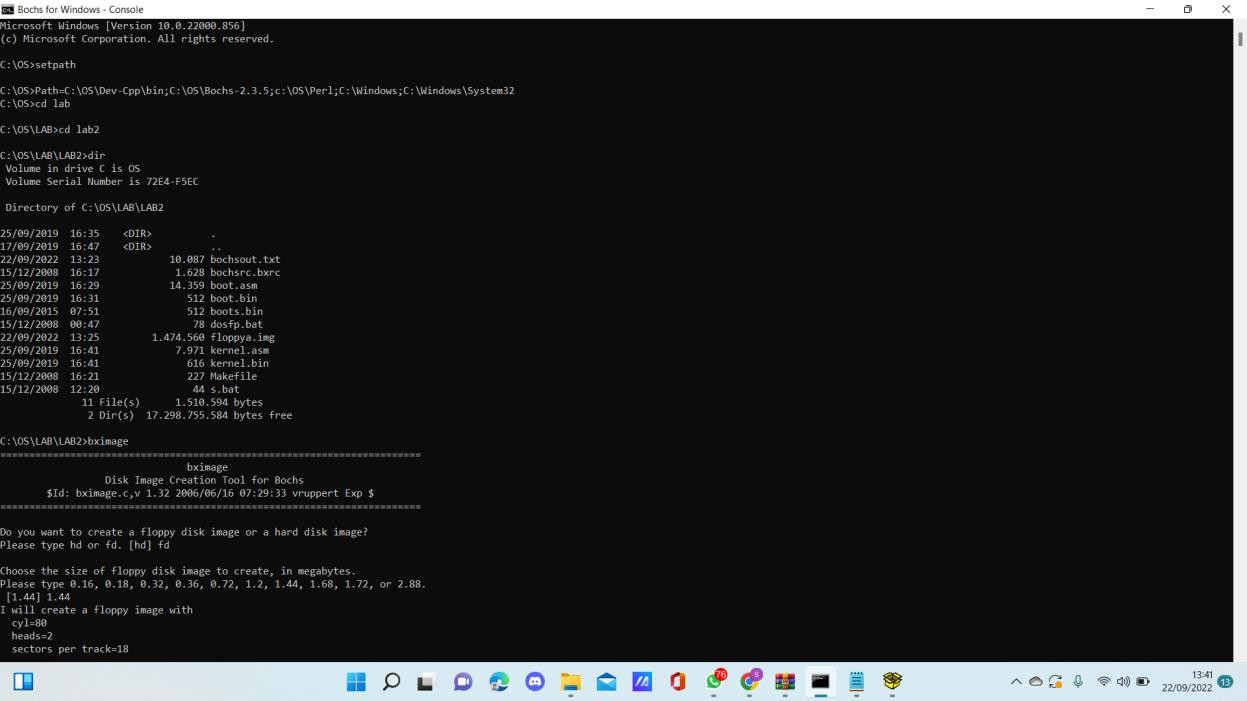
# RAMA AZIZ WIBOWO

# L200210263

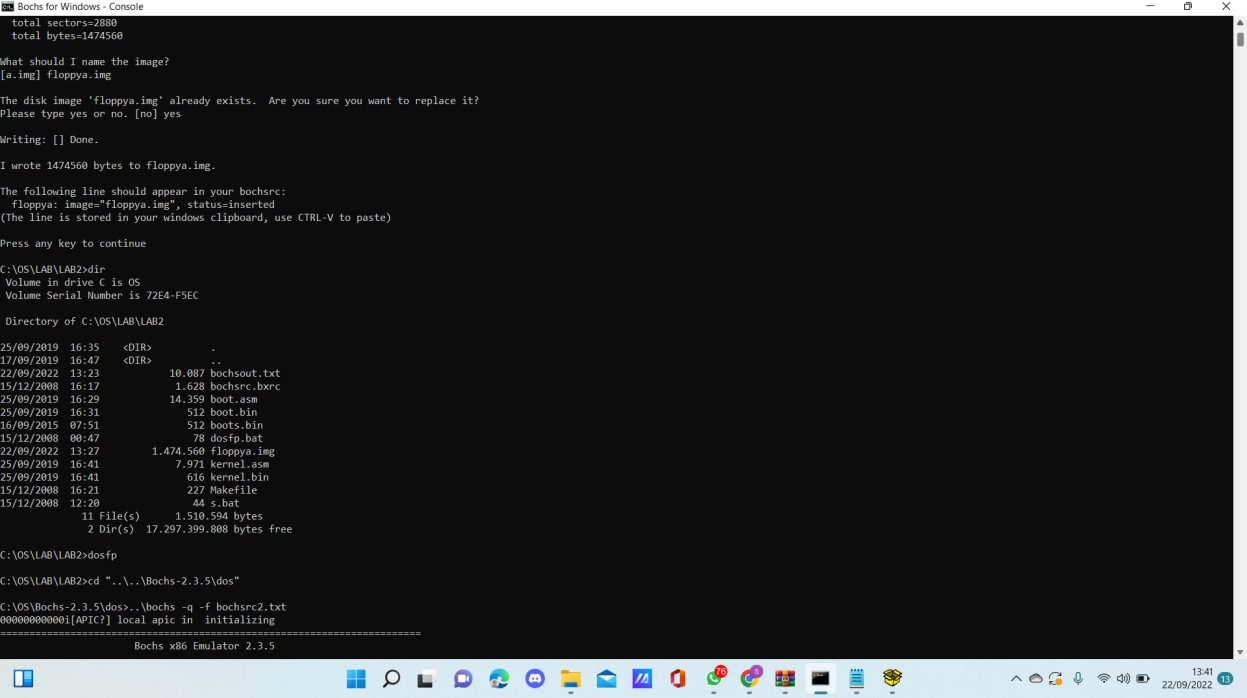
# KELAS E

**PROGAM STUDI TEKNIK INFORMATIKA FAKULTAS ILMU KOMUNIKASI DAN INFORMATIKA**

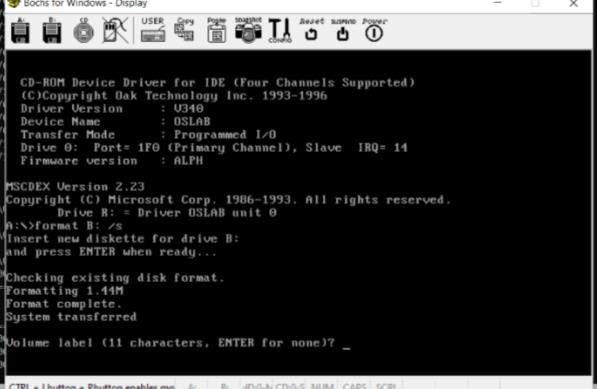
# UNIVERSITAS MUHAMMADIYAH SURAKARTA TAHUN 2020/2021



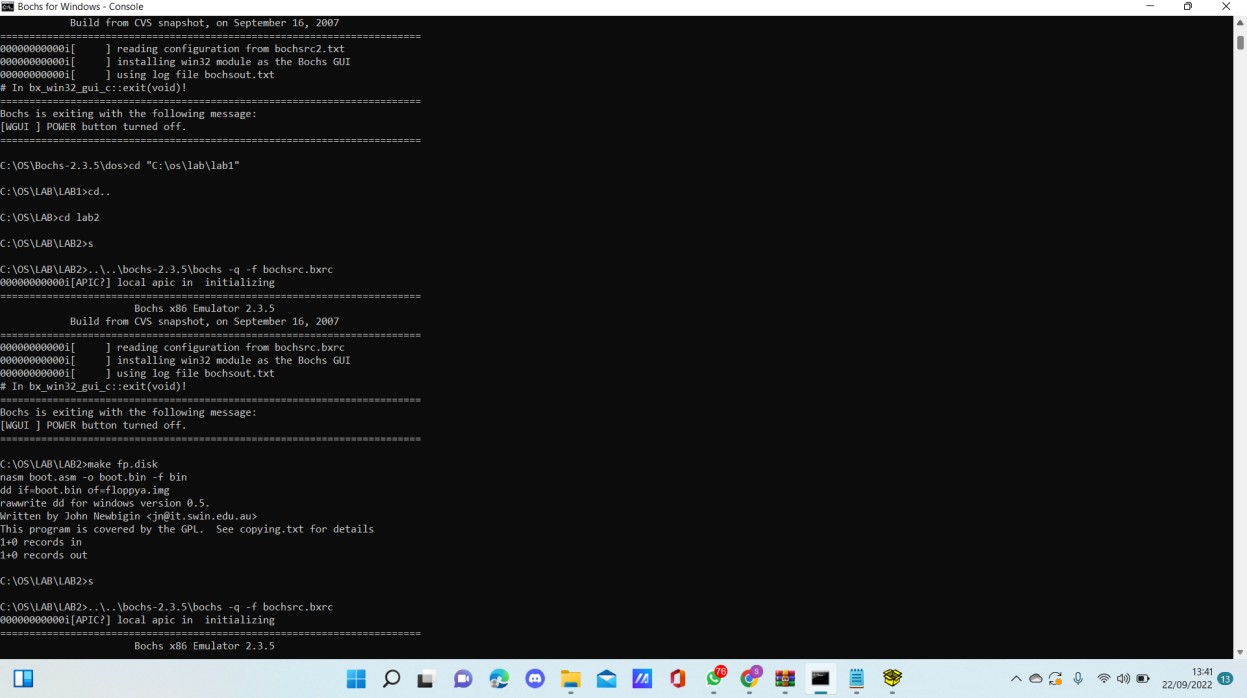
*Gambar 1. Menjalankan setpath, lalu membuka direktori Lab2. Dilanjutkan dengan menyiapkan file floppya.img dengan menjalankan perintah bximage.*



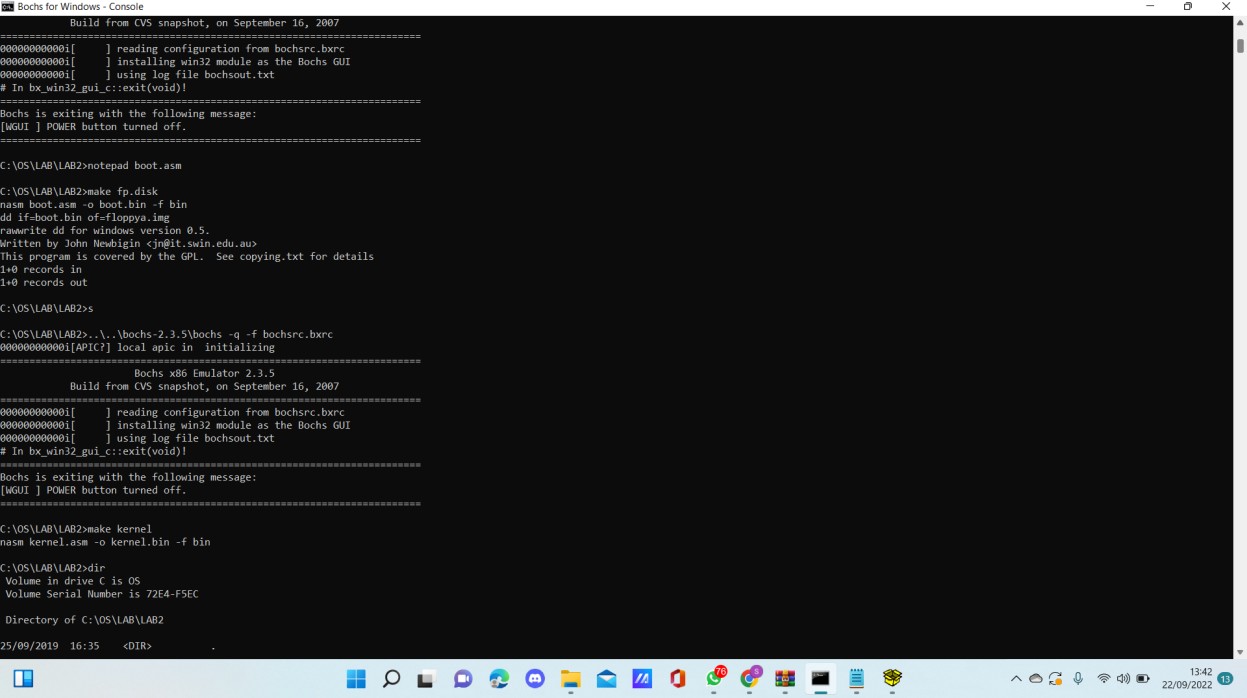
*Gambar 2. Memastikan file floppya.img. Dilanjutkan dengan menjalankan perintah dosfp.*



*Gambar 3. Selanjutnya pada halaman dosfp, klik penyimpanan B dan atur untuk disimpan pada direktori LAB2.*



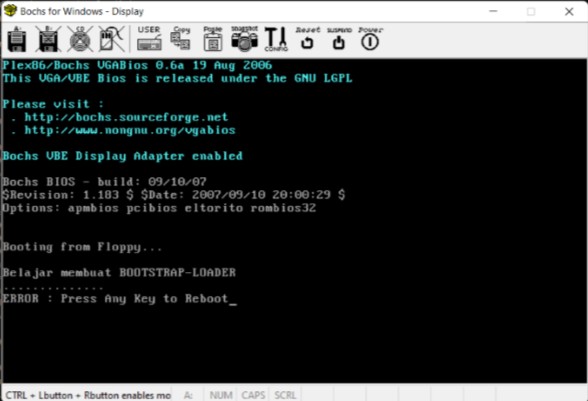
*Gambar 4. Selanjutnya turn off halaman dosfp, lalu ketik S pada halaman command prompt. Setelah proses untuk memastikan bahwa file floppya.img sekarang ada pada penyimpanan A, lalu ketikan perintah make fp.disk. Setelah itu lanjut ketik S lagi pada halaman commad prompt.*



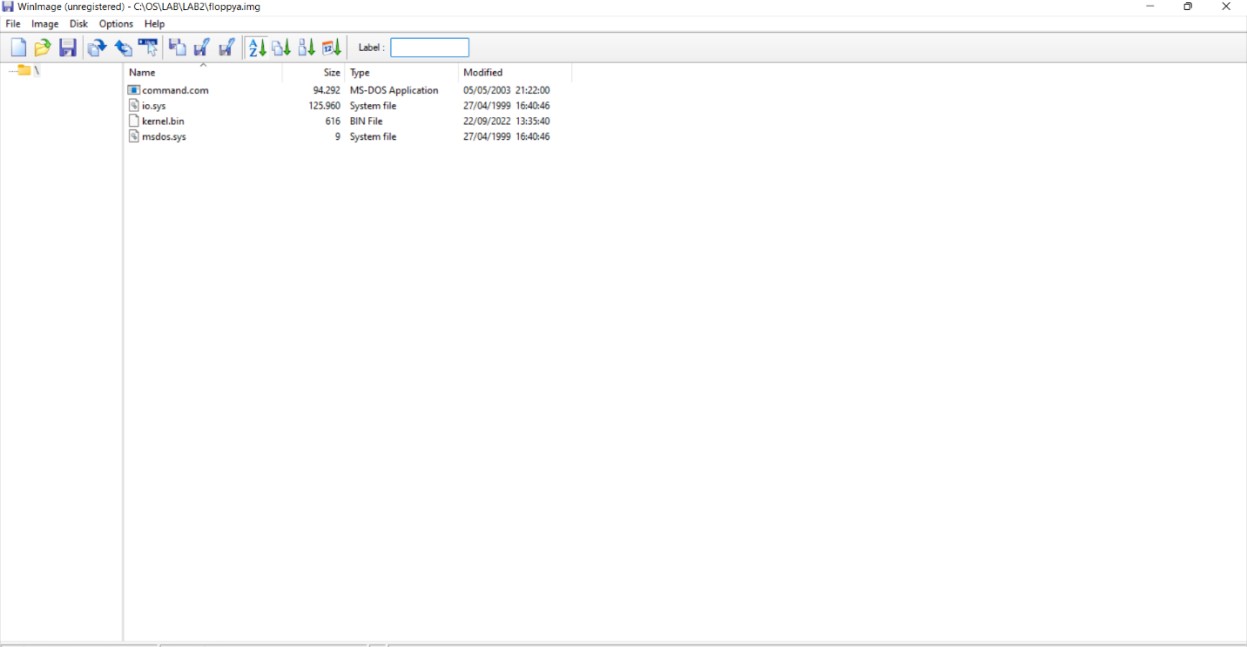
*Gambar 5. Selanjutnya yaitu membuka halaman notepad, dengan mengtikkan perintah notepad boot.asm untuk mengagganti teks menjadi “Belajar membuat BOOTSTRAP-LOADER”. Lalu ketikkan make fp.disk untuk mengganti teks pada halaman dosfp. Setelah itu lanjut untuk membuat kernel, dengan memasukkan perintah ake kernel pada halaman command prompt.*



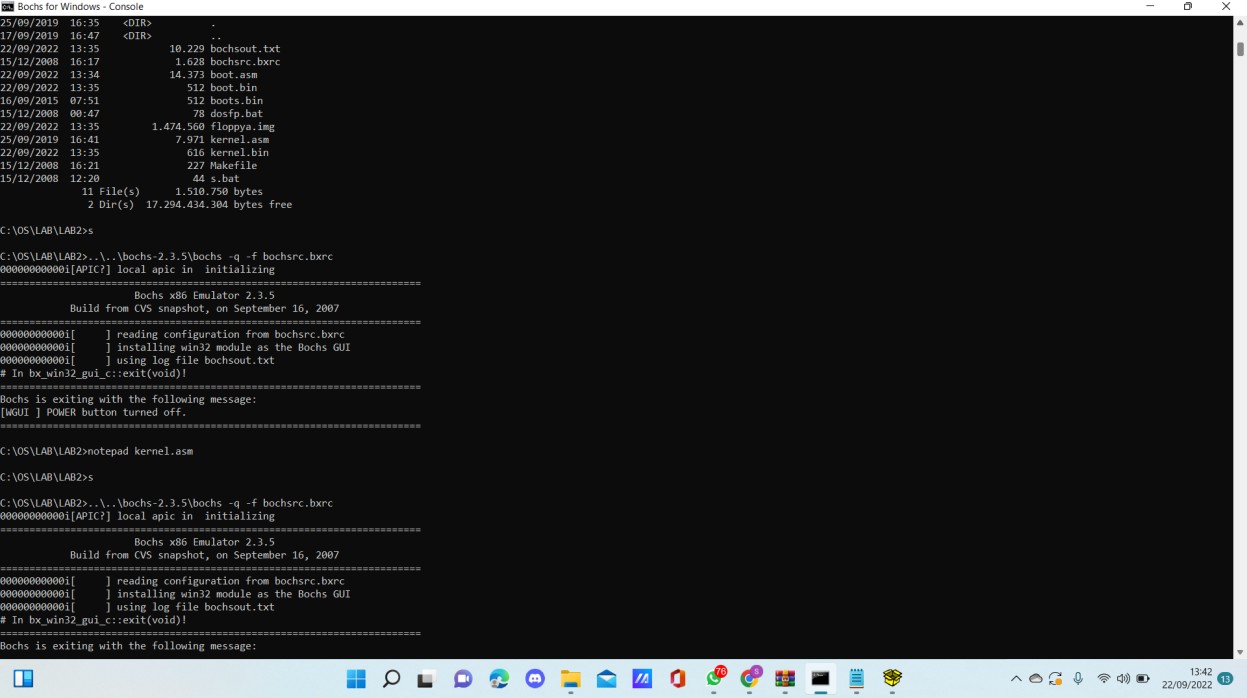
*Gambar 6. Mengganti teks pada halaman notepad boot.asm*



*Gambar 7. Hasil ditampilkan pada halaman dosfp.*



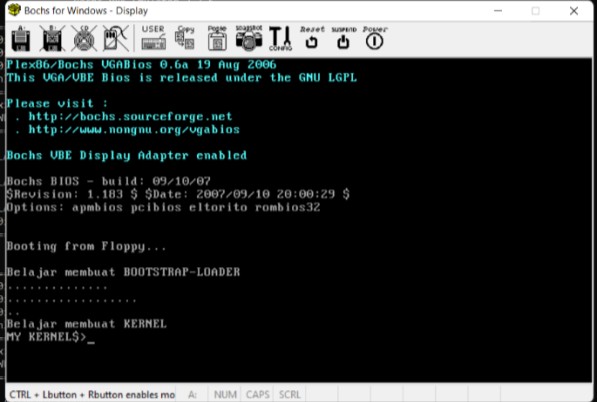
*Gambar 8. Setelah menjalankan perintah make kernel, lalu kita memindahkan file kernel.bin engan menggunakan winImage untuk dipindahkan ke file floppya.img.*



*Gambar 9. Setelah selelsai memindahkan file kernel.bin ke dalam filoe floppya.img, maka selanjutnya yaitu membuka notepad kernel.asm.*



*Gambar 10. Pada halaman notepad, kita diperintahkan untuk mengganti teks menjadi “Belajar membuat KERNEL”.*



*Gambar 11. Setelah mengganti teks pada halaman notepad, selanjutnya ulangi proses kompilasi seperti yang tadi untuk mengganti file kernel.bin yang terbaru, lalu ketikkan perintah make kernel pada halaman command prompt dan selanjutnya cek hasilnya pada halaman dosfp dengan mengetikkan perintah S pada halaman command prompt.*

# TUGAS MODUL 2

1. Pelajari cara kerja program “boot.asm” buatlah algoritma dari program tersebut dalam bentukFlowchart. Untuk memudahkan dalam memahami proses boot buatlah dua jenis algoritma, pertama buat algoritma yang bersifat global dan kedua buat algotitma yang bersifat lebih detail

Cara kerja program ‘boot.asm’ secara global:

* 1. Pertama nyalakan BIOS, pada tahap ini prosesor mencar alamat tertentu pada BIOS.
  2. Selanjutnya BIOS mengambil alih untuk menjadi system operasi pertama pada computer,setelah itu BIOS melakukan pengecekan terhadap seluruh galat (sumber variasi data) yang terdapat dalam memori. Jika terdapat masalah error dalam program maka proses tidak akan berlanjut, namun aka nada peringatan tentang masalah di dalam program.
  3. BIOS akan mencari kartu grafis yang tertanam pada computer dan jika sudah menemukankartu tersebut BIOS akan menjalankan kartu grafis tersebut. Dan tidak lupa untuk pengecekan BIOS terhadap ROM.
  4. Jika seluruh proses pengecekan oleh BIOS Sudha dilakukan, setelah itu BIOS yang akanmencari system operasi yang sudah terinstal setelah itu memuatnya pada memori serta segera mengeksekusinya.

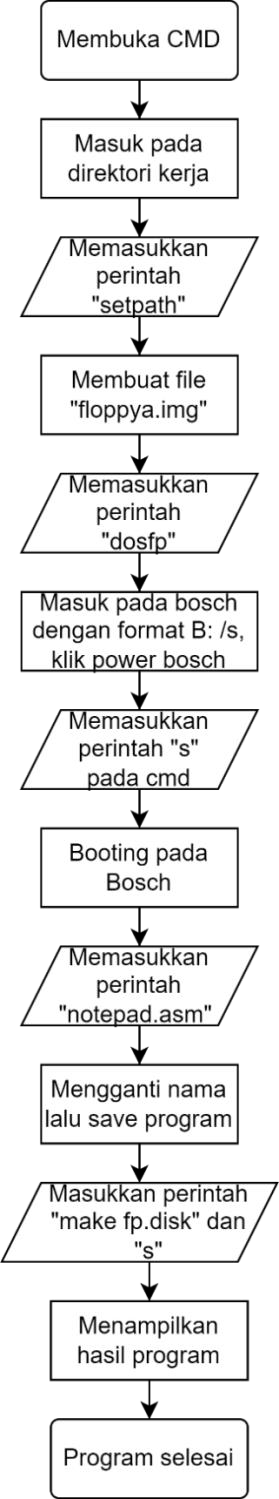
Cara kerja program ‘boot.asm’secara detail :

1. Membuka command prompt atau cmd
2. Memberikan perintah ‘setpath’
3. Mengatur disk ke direktori LAB2
4. Membuat file ‘floppya.img’ dengan perintah ‘bximage’ dan menyelesaikan prosesnya e.Menjalankan perintah ‘dosfp’
5. Setelah muncul jendela bochs kemudian berikan perintah ‘format B: /S’ dan selesaikanprosesnya
6. Tutup jendela bochs dengan klik tombol Power
7. Jalankan perintah ‘s’ pada command prompt, maka akan muncul kembali jendela bochs,dan kemudian tutup jendela bochs dengan klik tombol Power
8. Memberikan perintah ‘make fp.disk’
9. Memberikan perintah ‘notepad boot.asm’
10. Muncul jendela notepad , lalu sunting isi file ‘boot.asm’ tersebut pada bagian variablemsgLoading dan msgFailure
11. Menyimpan hasil suntingan
12. Menjalankan kembali perintah ‘make fp.disk’
13. Memberikan perintah ’s’
14. Maka akan muncul jendela bochs (hasil)
15. Selesai

Flowchart Cara kerja program ‘boot.asm’ secara global



Flowchart Cara kerja program ‘boot.asm’secara detail



1. Lakukan hal yang sama untuk program “kernel.asm”

Cara Kerja kernel.asm bersifal Kernel Global

* 1. Mengcompile source code ‘kernel.asm’ yang hasilnya adalah ‘kernel.bin’.
  2. Memindahkan hasil compile ke bootsector ‘floppya.img’.
  3. Menuju ke direktori floppya.img yang berada di derektori LAB2.
  4. Menjalankan program WinImage.
  5. Memasukkan (Inject) kernel.bin ke dalam WinImage (floppya.img)
  6. Menyimpan file.
  7. Menyunting file ‘kernel.asm’.
  8. Mengcompile sourch code ‘kernel.asm’ yang menghasilkan ‘kernel.bin’.
  9. Memindahkan hasil compile ke bootsector ‘floppya.img’.

Cara kerja program kernel secara detail :

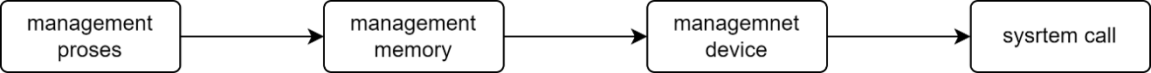
* + 1. Membuka command prompt atau cmd
    2. Memberikan perintah ‘setpath’
    3. Mengatur disk ke direktori LAB2
    4. Membuat file ‘floppya.img’ dengan perintah ‘bximage’ dan menyelesaikanprosesnya. Menjalankan perintah ‘dosfp’
    5. Setelah muncul jendela bochs kemudian berikan perintah ‘format B:

/S’ danselesaikan prosesnya

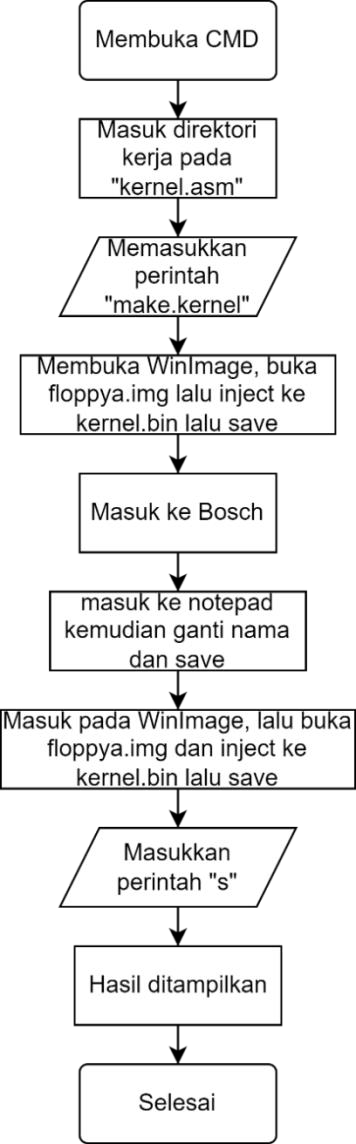
* + 1. Tutup jendela bochs dengan klik tombol Power
    2. Jalankan perintah ‘s’ pada command prompt, maka akan muncul kembali jendelabochs, dan kemudian tutup jendela bochs dengan klik tombol Power
    3. Memberikan perintah ‘make fp.disk’
    4. Memberikan perintah ‘notepad kernel.asm’
    5. Muncul jendela notepad , lalu sunting isi file ‘Welcome to MY KERNEL’ menjadi ‘Belajar membuat KERNEL’. Teks tersebut disimpan pada variable strWelcomeMsg
    6. Menyimpan hasil suntingan
    7. Menjalankan kembali perintah ‘make fp.disk’
    8. Memberikan perintah ’s’
    9. Maka akan muncul jendela bochs (hasil)
    10. Selesai

cara lain kerja kernel.asm secara Global

1. Proses management, pada bagian ini terjadi pengaturna antar proses program denganhardware. Pengaturnan masuk atau keluarnya untuk melakukan sebuah proses
2. Memory management, akan ada pengaturan memori oleh program
3. Device management, berfungsi untuk menjaid jembatan penggunaan dari hardware yangada pada system. Hardware akan dikenali dan digunakan oleh program dan system operasi
4. System call, pada bagian ini kernel akan mengatur antara hubungan dari program dansystem.



Flowchart Cara Kerja kernel.asm bersifal Kernel Global



Flowchart Cara Kerja kernel.asm bersifal Kernel Detail

