

Tugas Besar – Beneath The Skin

IF2230 Sistem Operasi

PENGEMBANGAN SISTEM OPERASI

oleh

Erick Wijaya 13515057

Veren Iliana Kurniadi 13515078

Zacki Zulfikar Fauzi 13515147



PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG

2017

JAWABAN PERTANYAAN

1. *Apa perbedaan cara booting disk MBR dengan GPT? Apa cara boot yang dipakai dalam tugas ini?*

Pada disk MBR, informasi mengenai lokasi sistem file dan partisi sistem operasi hanya tersimpan pada sektor pertama, sedangkan pada disk GPT, informasi sistem file dan partisi sistem operasi disimpan lebih dari satu lokasi. Selama proses booting MBR akan mencai lokasi dimana terdapat *boot sector*. MBR mendukung proses booting dengan BIOS, sedangkan GPT mendukung booting dengan UEFI. Proses booting dalam tugas ini menggunakan BIOS 16 bit.

2. *Apakah mungkin menginstall OS ini ke floppy disk atau disk lain? Jelaskan.*

Instalasi OS tidak dapat dilakukan pada floppy disk karena cara booting disk berbeda antara MBR dan GPT. Instalasi OS tidak dapat dilakukan bila disk GPT sehingga tidak mendukung proses booting dengan BIOS sehingga OS tidak dapat diinstall ke disk lain.

3. *Apa perbedaan booting secara real mode dan protected mode? Mode apa yang dipakai dalam tugas ini?*

Booting secara real mode (real address mode) adalah metode booting yang menyediakan 1MiB memori dan akses tak hingga untuk setiap memori yang disediakan. Real mode tidak menyediakan proteksi memori maupun *multitasking*. Berbeda dengan real mode, protected mode memiliki akses penuh terhadap seluruh memori pada sistem dan tidak dibatasi dengan 1MiB memori. Protected mode juga menyediakan fitur *multitasking*, yang berarti sistem operasi mampu mengeksekusi banyak program sekaligus. Protected mode memiliki akses yang lebih cepat untuk prosesor 32-bit maupun drivers I/O 32-bit. Protected mode memiliki fitur *virtual memory*, yaitu fitur yang memperlakukan harddisk seperti memori fisik. Protected mode menjaga keamanan memori dengan mencegah akses invalid pada kode kernel dari program dan proses crash tidak memengaruhi keseluruhan sistem. Mode yang dipakai dalam tugas ini

adalah Real Mode sehingga *BIOS Interrupts* dapat dilakukan. Alasan mengapa digunakan Real Mode terdapat pada nomor 4.

4. *Mengapa BIOS sekarang masih ada yang dieksekusi secara 16-bit?*

BIOS masih dieksekusi secara 16-bit agar memiliki backward compatibility. Eksekusi BIOS dalam 16 bit dinamakan real mode. Processor x86 modern dapat menjalankan program yang dibuat untuk processor 8088 seperti MS-DOS tanpa harus beralih pada emulasi software. Oleh karena backward compatibility merupakan aspek yang penting dalam processor x86, hampir seluruh processor x86 masih menjalankan BIOS secara 16 bit (Real mode).

5. *Mengapa magic number yang digunakan oleh bootloader adalah 55 AA ? Apa akibatnya jika magic number tersebut diubah nilainya?*

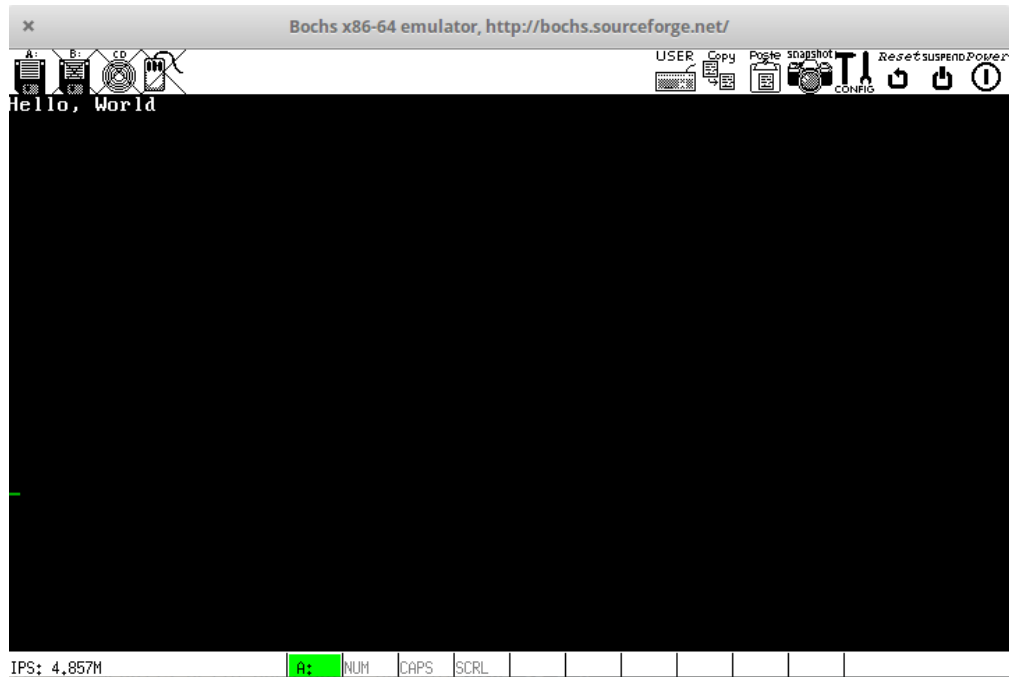
Magic number 55 AA digunakan dalam bootloader karena sistem operasi menggunakan MBR (Master Boot Record). MBR memiliki kode 55 AA pada 2 byte terakhirnya. Jika magic number tersebut diubah nilainya maka booting tidak akan berjalan dengan normal karena magic number lah yang memberitahu komputer cara booting dan menjelaskan partisi dalam hard drive yang digunakan.

KESULITAN YANG DIHADAPI

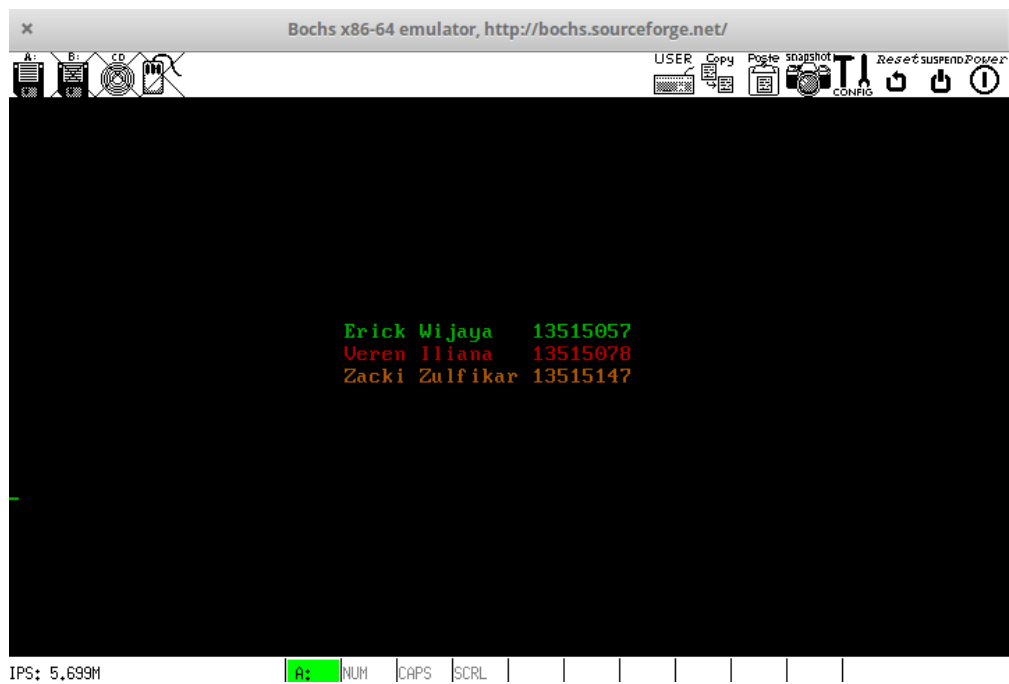
Kendala yang dihadapi kelompok kami adalah ketika mencoba mencetak sebuah huruf 'A' pada kernel. Beberapa kali kelompok kami memodifikasi kode program, hasilnya huruf 'A' dapat tampil pada kernel. Akan tetapi, ketika ditambahkan fungsi *printString*, huruf 'A' yang harusnya ditampilkan pada kernel justru hilang. Sebenarnya kelompok kami sudah melakukan perhitungan memori dengan benar untuk fungsi *printString* sesuai dari panduan yang sudah diberikan. Kelompok kami kemudian menyadari bahwa sintaks program C untuk dikompilasi bcc sedikit berbeda dengan biasanya. Setelah kelompok kami memindahkan implementasi fungsi *printString* dibawah fungsi *main*, fungsi *printString* akhirnya dapat mencetak tulisan pada kernel. Setelah itu, kelompok kami mulai menyusun logo dan membuat fungsi-fungsi lain seperti *clearscreen* dan *putLogo*.

SCREENSHOT

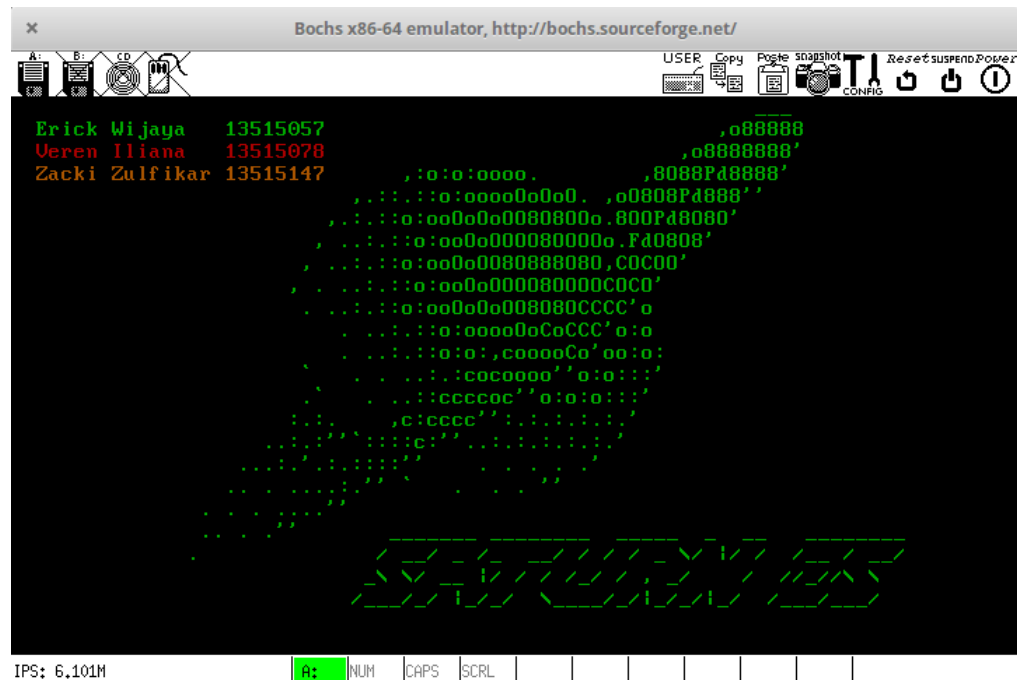
1. Hello World



2. Print String



3. Logo OS



PEMBAGIAN TUGAS

Nama	Tugas	Persentase
Erick Wijaya	<ul style="list-style-type: none">- Menjawab pertanyaan 2 dan 3- Membantu membuat prosedur printString- Membuat prosedur clearscreen dan putLogo	33,3%
Veren Iliana Kurniadi	<ul style="list-style-type: none">- Menjawab pertanyaan 1- Membantu membuat prosedur printString- Mendesain logo OS	33.4%
Zacki Zulfikar Fauzi	<ul style="list-style-type: none">- Menjawab pertanyaan 4 dan 5- Membuat prosedur printString- Menulis shell script	33.3%

FEEDBACK

Menurut kelompok kami, tugas besar milestone pertama ini sudah cukup baik karena tugas ini secara sukses memperkenalkan kami dengan cara membuat sistem operasi yang simple. Instruksi yang diberikan sudah cukup jelas dan mudah diikuti walaupun kami sempat kebingungan karena tugas ini menggunakan bahasa C lama. Secara keseluruhan tugas ini sudah sangat membantu dalam mengajarkan kami membuat sistem operasi walaupun masih tergolong sangat simple.