

**Laporan Tugas Besar Milestone 1
IF 2230 Sistem Operasi**

Booting, Kernel, File System, System Call, Eksekusi Program

Disusun oleh

Kelompok “Rika’OS”

Lukas Kurnia Jonathan / 13517006

Willy Santoso / 13517066

Rika Dewi / 13517147



**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG
2019**

DAFTAR ISI

DAFTAR ISI	1
BAB 1 JAWABAN PERTANYAAN	2
1.1 [Bootloader]	2
1.2 [Kernel]	2
1.3 [Write to Memory]	2
1.4 [Sector]	3
1.5 [Read/Write]	3
BAB 2 SCREENSHOT Pengerjaan	4
2.1 Menjalankan Sistem Operasi (Langkah 6)	4
2.2 Mengeksekusi Sistem Operasi (Langkah 8)	4
2.3 Bonus	4
2.3.1 Tampilan Logo OS	4
2.3.2 Kalukator Sederhana	4
PEMBAGIAN TUGAS	5
KESULITAN DAN FEEDBACK	6

BAB 1 JAWABAN PERTANYAAN

1.1 [Bootloader]

Apa perbedaan booting disk MBR dengan GPT? Cara boot yang mana yang digunakan dalam tugas ini?

- Master Boot Record (MBR) adalah *special boot sector* ketika awal mula booting disk yang memiliki informasi bootloader dari suatu operating system dan informasi partisi disk. MBR adalah metode lama namun tetap menjadi *default selection* ketika booting disk.
- Tipe baru adalah GUID Partition Table (GPT) menggunakan UEFI
- Perbedaan:
 - Perbedaan yang paling besar adalah MBR menyimpan data di satu tempat, jika data tersebut hilang / *corrupt* maka seluruh OS dapat down (perlu *repairing MBR*). Sedangkan GPT menyimpan *multiple copies* dari partisi dan boot sepanjang disk. Partisi ini dapat digunakan untuk *recovery* data. Selain itu, GPT memiliki *cyclic redundancy check* (CRC) untuk mengecek integritas dari data secara berkala.
 - MBR mensupport sampai 4 buah primary partition jika membutuhkan lebih harus menggunakan extended partition (ada limit sendiri), sedangkan GPT support 128 partisi (Windows limit) sehingga memudahkan tanpa harus menggunakan extended partition. Limit GPT didasarkan pada operating system yang dijalankan.
 - Pada MBR, maksimum disk size yang disupport adalah 2 TB. Partisi disimpan dengan besar 4 bytes (32 bit). Berbeda dengan GPT, partisi disimpan dengan besar 8 bytes (64 bit) atau secara teori setara 9,4 ZB. Akan tetapi, *practically* maksimum disk size tetap bergantung limit dari OS itu sendiri.
- Cara boot yang digunakan dalam tugas ini adalah Master Boot Record (MBR)

1.2 [Kernel]

Apa itu Kernel Panic dan mengapa bisa terjadi?

- Kernel Panic adalah suatu kondisi dimana kernel tidak dapat diproses dan di-*boot* dengan benar (*crash*). Hal ini merupakan tindakan preventif yang dilakukan oleh OS untuk proteksi dari dampak tindakan yang membuat “Kernel Panic” ini terjadi. Misal *overheating, hard drives corruption, hardware problems*, dll yang tidak diinginkan.
- Kernel panic dapat terjadi karena banyak hal, antara lain update yang gagal, kegagalan pekerjaan hardware, fungsi-fungsi dalam kernel gagal bekerja dengan benar.

1.3 [Write to Memory]

Apa yg terjadi jika code snippet ini dijalankan di kernel mode dan di usermode?

```
int main (void) {  
    *((int*)0) = 0x11fe;  
}
```

```
}
```

- `((int*)0)` berisi dengan nilai null, sehingga `*((int*)0)` adalah pointer yang menuk pada lokasi memori tidak valid untuk dialokasikan. Jika kode di atas dijalankan pada kernel mode, proses akan dijalankan pada memori sebenarnya. Hal ini karena kernel mode memiliki akses penuh terhadap memori. Hal ini dapat memengaruhi bootloader dan berpotensi menimbulkan risiko.
- Jika dijalankan pada user mode, maka kode di atas tidak dapat dijalankan. Hal ini karena proses akan berjalan pada memori virtual sehingga tidak dapat mengubah bagian memori.

1.4 [Sector]

Apa kelebihan dan kekurangan jika setiap sector pada suatu Hard Disk berukuran besar, misalkan 64MB?

- Kelebihannya dirasakan jika file-file yang disimpan pada Harddisk juga berukuran besar. File besar itu akan disimpan pada beberapa sektor saja, sehingga untuk mengakses file tersebut tidak diperlukan akses ke banyak sektor sekaligus. Dampaknya adalah akses ke file yang menjadi lebih cepat.
- Kekurangannya dirasakan jika file-file yang disimpan pada Harddisk berukuran kecil. Ketika disimpan dalam Harddisk, file akan disimpan dalam sektor dan setiap sektornya hanya boleh berisi satu file. Misal file yang digunakan hanya memakai 1% dari ukuran sektor. Maka sisa 99% dari ukuran sektor akan terbuang sia-sia. Dampaknya adalah ketidakefisienan memori.

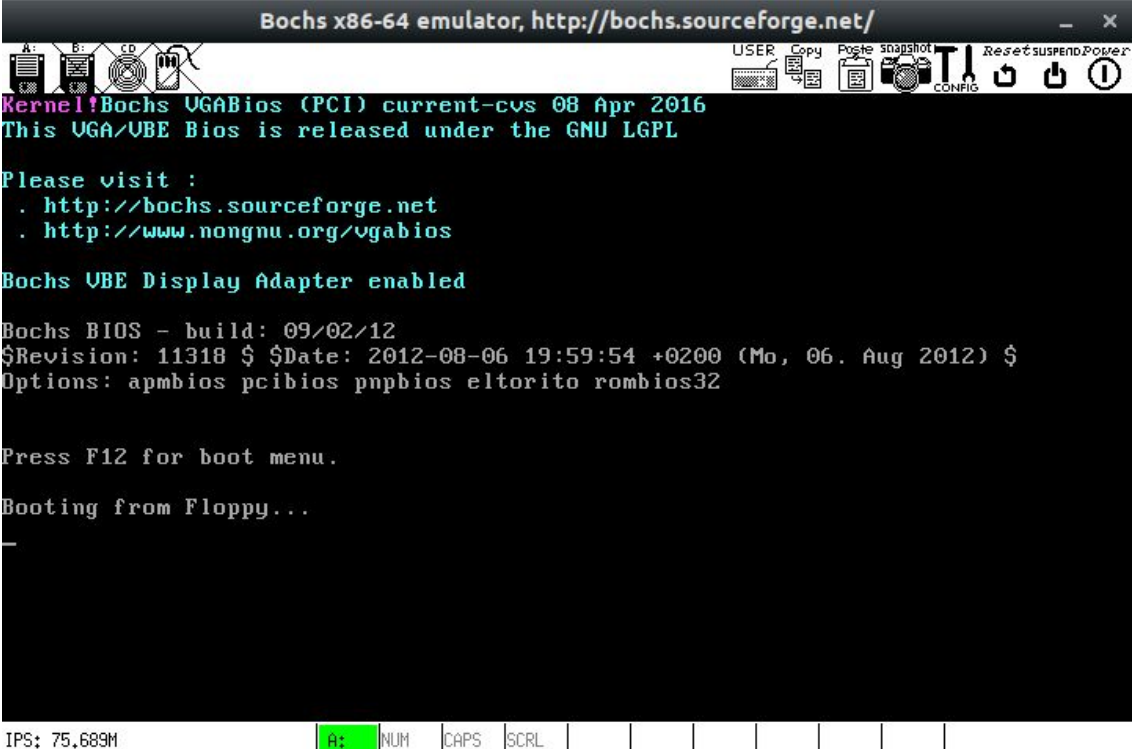
1.5 [Read/Write]

Melakukan file I/O pada kernel cenderung dikatakan pelanggaran standard practice, mengapa?

- Karena melakukan file I/O pada kernel dapat memicu banyak masalah. Masalah yang ditimbulkan antara lain interpreting data, buffer overflow, akses ke data yang dilindungi, dan masalah kebijakan/policy.

BAB 2 SCREENSHOT Pengerjaan

2.1 Menjalankan Sistem Operasi (Langkah 6)



The screenshot shows the Bochs x86-64 emulator window. The title bar reads "Bochs x86-64 emulator, http://bochs.sourceforge.net/". The main display area shows the following text:

```
Kernel!Bochs UGABios (PCI) current-cvs 08 Apr 2016
This UGA/UBE Bios is released under the GNU LGPL

Please visit :
. http://bochs.sourceforge.net
. http://www.nongnu.org/ugabios

Bochs UBE Display Adapter enabled

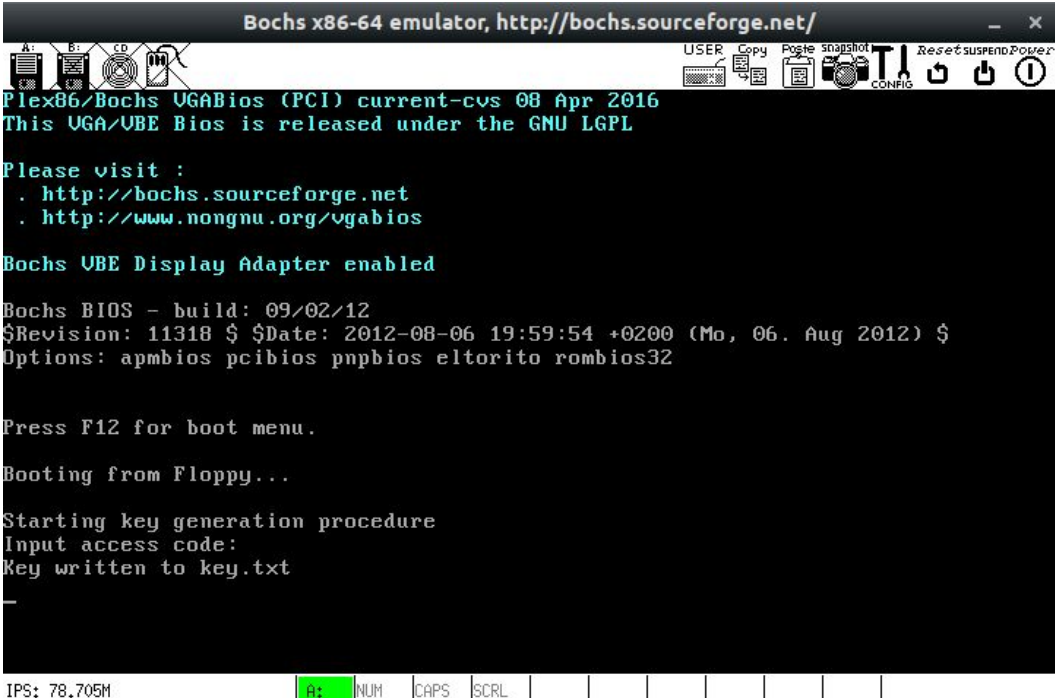
Bochs BIOS - build: 09/02/12
$Revision: 11318 $ $Date: 2012-08-06 19:59:54 +0200 (Mo, 06. Aug 2012) $
Options: apmbios pcibios pnpbios eltorito rombios32

Press F12 for boot menu.

Booting from Floppy...
```

At the bottom of the window, there is a status bar showing "IPS: 75.689M" and a keyboard layout indicator with "A:" highlighted in green, followed by "NUM", "CAPS", "SCRL", and several empty slots.

2.2 Mengeksekusi Sistem Operasi (Langkah 8)



The screenshot shows the Bochs x86-64 emulator window. The title bar reads "Bochs x86-64 emulator, http://bochs.sourceforge.net/". The main display area shows the following text:

```
Plex86/Bochs UGABios (PCI) current-cvs 08 Apr 2016
This UGA/UBE Bios is released under the GNU LGPL

Please visit :
. http://bochs.sourceforge.net
. http://www.nongnu.org/ugabios

Bochs UBE Display Adapter enabled

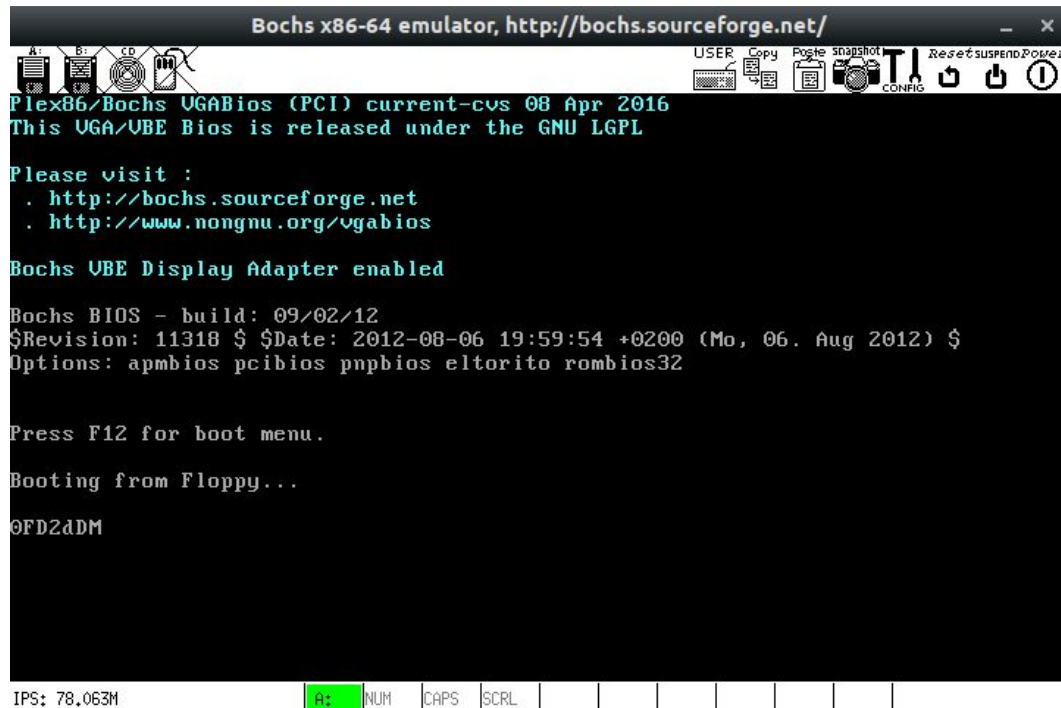
Bochs BIOS - build: 09/02/12
$Revision: 11318 $ $Date: 2012-08-06 19:59:54 +0200 (Mo, 06. Aug 2012) $
Options: apmbios pcibios pnpbios eltorito rombios32

Press F12 for boot menu.

Booting from Floppy...

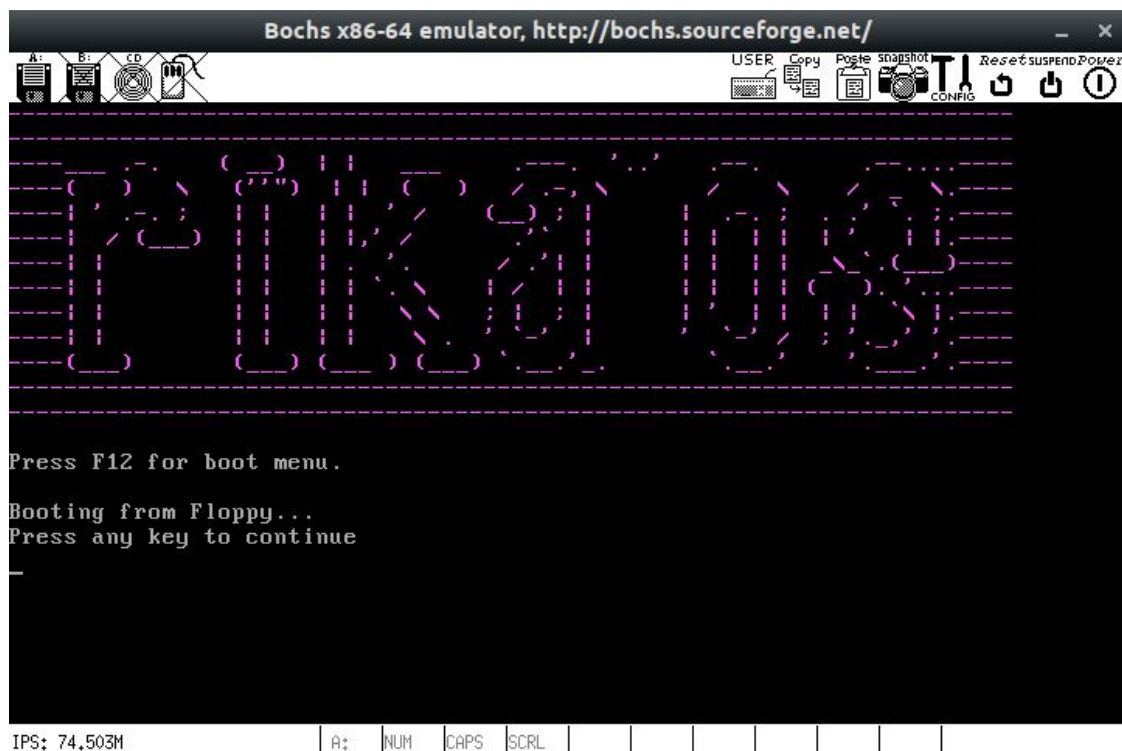
Starting key generation procedure
Input access code:
Key written to key.txt
```

At the bottom of the window, there is a status bar showing "IPS: 78.705M" and a keyboard layout indicator with "A:" highlighted in green, followed by "NUM", "CAPS", "SCRL", and several empty slots.

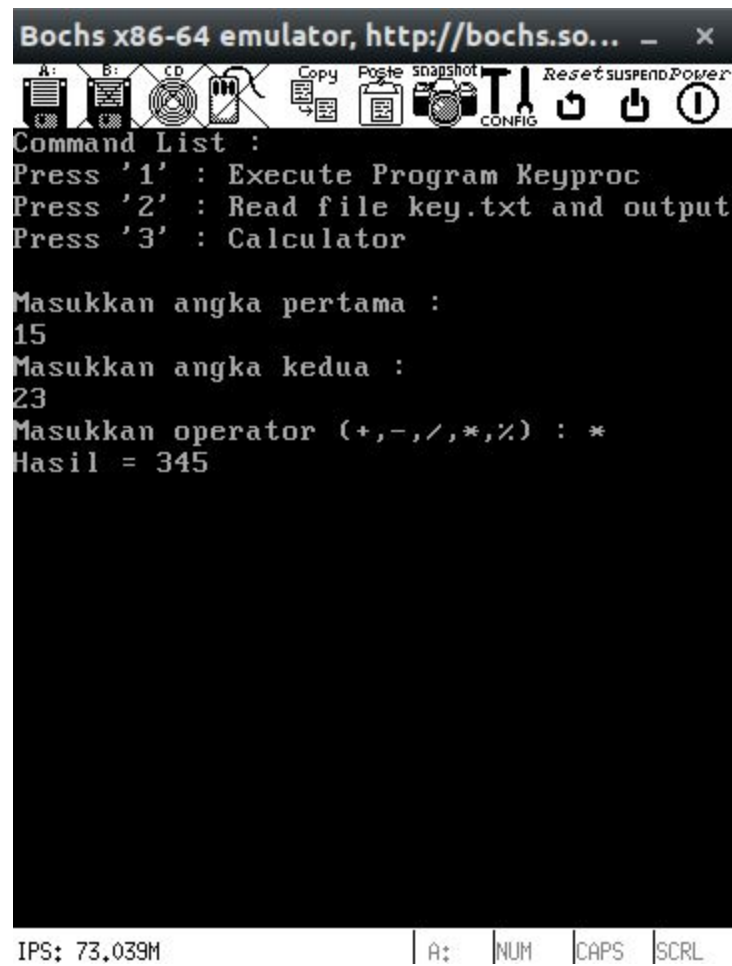


2.3 Bonus

2.3.1 Tampilan Logo OS



2.3.2 Kalukator Sederhana



The screenshot shows a Bochs x86-64 emulator window. The title bar reads "Bochs x86-64 emulator, http://bochs.so...". The menu bar includes "A:", "B:", "CD", "Copy", "Paste", "snapshot", "Reset", "suspend", and "Power". The toolbar contains icons for these functions. The main window displays the following text:

```
Command List :  
Press '1' : Execute Program Keyproc  
Press '2' : Read file key.txt and output  
Press '3' : Calculator  
  
Masukkan angka pertama :  
15  
Masukkan angka kedua :  
23  
Masukkan operator (+,-,/,*,%) : *  
Hasil = 345
```

At the bottom of the window, the status bar shows "IPS: 73,039M" and a keyboard layout indicator with "A:", "NUM", "CAPS", and "SCRL" buttons.

PEMBAGIAN TUGAS

NIM	NAMA	BAGIAN KERJA	PERSENTASE KONTRIBUSI
13517006	Lukas Kurnia Jonathan	Implementasi kernel.c & laporan	33.3%
13517066	Willy Santoso	Implementasi kernel.c & laporan	33.3%
13517144	Rika Dewi	Implementasi kernel.c & laporan	33.3%

KESULITAN DAN FEEDBACK

- Referensi belajar yang didapat terbatas karena hanya mengandalkan spesifikasi yang diberikan asisten. Sehingga ketika ada kendala menjadi kesulitan.
- Saat membuat soal bonus, pembuatan kalkulator sederhana, terdapat nilai variabel yang berubah karena fungsi ReadString yang digunakan konflik dengan ReadInt yang dibuat.
- Beberapa fungsi interrupt yang harus dicoba-coba terlebih dahulu.
- Penyelesaian kasus backspace pada proses input.