



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

BAB	: INPUT OUTPUT
NAMA	: DANI ADRIAN
NIM	: 225150201111009
TANGGAL	: 16/05/2023
ASISTEN	: ZHAFRAN RAMA AZMI
	GIBRAN HAKIM

9.3. Dasar Teori

9.3.3 Pengelolaan I/O pada Level Sistem Operasi dengan Kernel Module

Kernel module Linux dapat dibangun dengan beberapa langkah berikut :

1. Pastikan bahwa paket untuk pengembangan kernel sudah terpasang

```
sudo apt-get install build-essential linux-headers-  
$(uname -r)
```

Jawab:

```
userlinux@Linux:~$ sudo apt-get install build-essential linux-headers-$(uname -r)
[sudo] password for userlinux:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
linux-headers-5.19.0-41-generic is already the newest version (5.19.0-41.42-22.04.1).
linux-headers-5.19.0-41-generic set to manually installed.
The following additional packages will be installed:
  cpp-11 dpkg-dev fakeroot g++ g++-11 gcc-11 gcc-11-base libalgorithm-diff-perl libalgorithm-diff-x
  libstdc++-11-dev libtsan0 lto-disabled-list make
Suggested packages:
  gcc-11-locales debian-keyring g++-multilib g++-11-multilib gcc-11-doc gcc-11-multilib git bzr lib
The following NEW packages will be installed:
  build-essential dpkg-dev fakeroot g++ g++-11 libalgorithm-diff-perl libalgorithm-diff-xs-perl lib
  lto-disabled-list make
The following packages will be upgraded:
  cpp-11 gcc-11 gcc-11-base libasan6 libgcc-11-dev libtsan0
6 upgraded, 14 newly installed, 0 to remove and 42 not upgraded.
Need to get 52.2 MB of archives.
After this operation, 54.9 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Open ▾



nam...

~/

1 Nama : Dani Adrian

2 NIM : 225150201111009



```
Preparing to unpack .../12-dpkg-dev_1.21.1ubuntu2.2_all.deb ...
Unpacking dpkg-dev (1.21.1ubuntu2.2) ...
Selecting previously unselected package build-essential.
Preparing to unpack .../13-build-essential_12.9ubuntu3_amd64.deb ...
Unpacking build-essential (12.9ubuntu3) ...
Selecting previously unselected package libfakeroot:amd64.
Preparing to unpack .../14-libfakeroot_1.28-1ubuntu1_amd64.deb ...
Unpacking libfakeroot:amd64 (1.28-1ubuntu1) ...
Selecting previously unselected package fakeroot.
Preparing to unpack .../15-fakeroot_1.28-1ubuntu1_amd64.deb ...
Unpacking fakeroot (1.28-1ubuntu1) ...
Selecting previously unselected package libalgorithm-diff-perl.
Preparing to unpack .../16-libalgorithm-diff-perl_1.201-1_all.deb ...
Unpacking libalgorithm-diff-perl (1.201-1) ...
Selecting previously unselected package libalgorithm-diff-xs-perl.
Preparing to unpack .../17-libalgorithm-diff-xs-perl_0.04-6build3_amd64.deb ...
Unpacking libalgorithm-diff-xs-perl (0.04-6build3) ...
Selecting previously unselected package libalgorithm-merge-perl.
Preparing to unpack .../18-libalgorithm-merge-perl_0.08-3_all.deb ...
Unpacking libalgorithm-merge-perl (0.08-3) ...
Selecting previously unselected package libfile-fcntllock-perl.
Preparing to unpack .../19-libfile-fcntllock-perl_0.22-3build7_amd64.deb ...
Unpacking libfile-fcntllock-perl (0.22-3build7) ...
Setting up gcc-11-base:amd64 (11.3.0-1ubuntu1~22.04.1) ...
Setting up lto-disabled-list (24) ...
Setting up libfile-fcntllock-perl (0.22-3build7) ...
Setting up libalgorithm-diff-perl (1.201-1) ...
Setting up libfakeroot:amd64 (1.28-1ubuntu1) ...
Setting up libasan6:amd64 (11.3.0-1ubuntu1~22.04.1) ...
Setting up fakeroot (1.28-1ubuntu1) ...
update-alternatives: using /usr/bin/fakeroot-sysv to provide /usr/bin/fakeroot (fakeroot) in auto mode
Setting up make (4.3-4.1build1) ...
Setting up libdpkg-perl (1.21.1ubuntu2.2) ...
Setting up libalgorithm-diff-xs-perl (0.04-6build3) ...
Setting up libalgorithm-merge-perl (0.08-3) ...
Setting up libtsan0:amd64 (11.3.0-1ubuntu1~22.04.1) ...
Setting up cpp-11 (11.3.0-1ubuntu1~22.04.1) ...
Setting up dpkg-dev (1.21.1ubuntu2.2) ...
Setting up libgcc-11-dev:amd64 (11.3.0-1ubuntu1~22.04.1) ...
Setting up gcc-11 (11.3.0-1ubuntu1~22.04.1) ...
Setting up libstdc++-11-dev:amd64 (11.3.0-1ubuntu1~22.04.1) ...
Setting up g++-11 (11.3.0-1ubuntu1~22.04.1) ...
Setting up g++ (4:11.2.0-1ubuntu1) ...
update-alternatives: using /usr/bin/g++ to provide /usr/bin/c++ (c++) in auto mode
Setting up build-essential (12.9ubuntu3) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
user@linux@Linux:~$
```

`sudo apt-get install build-essential linux-headers-$(uname -r)` berfungsi untuk menginstal paket yang diperlukan untuk pengembangan perangkat lunak.

2. Buat kode kernel module dengan menggunakan bahasa C. Import library untuk pengembangan kernel module seperti "linux/module.h" dan "linux/kernel.h". Sebagai contoh, berikut adalah kernel module sederhana untuk mencetak informasi pada log file Linux. (**kernelmodule.c**).

```
#include <linux/module.h>
#include <linux/kernel.h>
```



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
static int __init hello_init(void)
{
    pr_info("Hello, Kernel!\n");
    return 0;
}

static void __exit hello_exit(void)
{
    pr_info("Goodbye, Kernel!\n");
}

module_init(hello_init);
module_exit(hello_exit);
MODULE_LICENSE("GPL");
MODULE_AUTHOR("Your Name");
MODULE_DESCRIPTION("Hello Kernel Module");
```

Jawab:



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
GNU nano 6.2

#include <linux/module.h>
#include <linux/kernel.h>

static int __init hello_init(void)
{
    pr_info("Hello, Kernel!\n");
    return 0;
}

static void __exit hello_exit(void)
{
    pr_info("Goodbye, Kernel!\n");
}

module_init(hello_init);
module_exit(hello_exit);
MODULE_LICENSE("GPL");
MODULE_AUTHOR("Your Name");
MODULE_DESCRIPTION("Hello Kernel Module");
```

Open ▾

nam...
~/

Save

≡

—

1 Nama : Dani Adrian
2 NIM : 225150201111009|

Plain Text ▾ Tab Width: 8 ▾ Ln 2, Col 22

File Name to Write: hello.c|

^G Help
^C Cancel

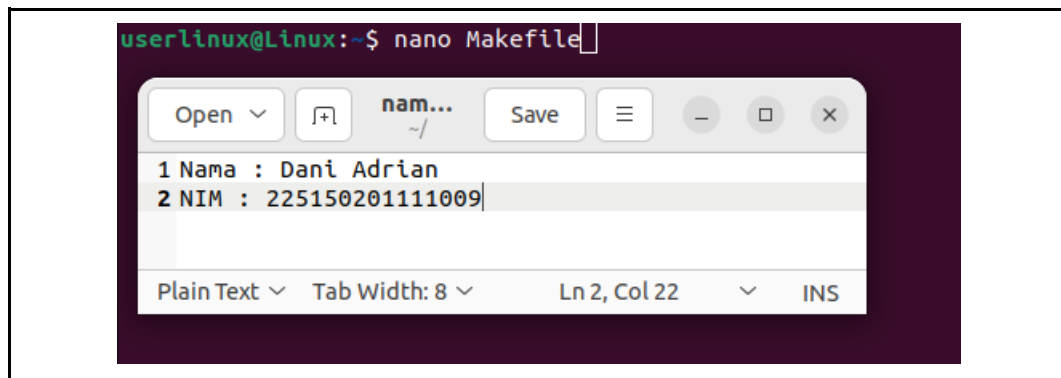


LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

3. Buat file bernama Makefile untuk proses kompilasi kode.

```
obj-m += hello.o
all:
make -C /lib/modules/$(shell uname -r)/build M=$(PWD)
modules
clean:
make -C /lib/modules/$(shell uname -r)/build M=$(PWD)
clean
```

Jawab:





LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
GNU nano 6.2
obj-m += hello.o

all:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules

clean:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

Open [icon] nam... [icon] Save [icon] [icon] [icon] [icon]

1 Nama : Dani Adrian
2 NIM : 225150201111009

Plain Text [icon] Tab Width: 8 [icon] Ln 2, Col 22 [icon] INS

File Name to Write: Makefile
^G Help M-D DOS Format
^C Cancel M-M Mac Format

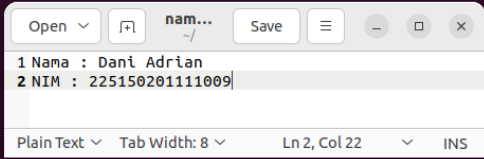
4. Build kernel module dengan menjalankan perintah **make**. Kompilasi akan menghasilkan file bernama **hello.ko**.

Jawab:



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
userlinux@Linux:~$ ls
alloc-best-fit  alloc-first-fit.c  baca      deadlock     dir      Documents  join.c
alloc-best-fit.c  alloc-worst-fit     baca.c    deadlock.c   dir.c    Downloads  main
alloc-first-fit  alloc-worst-fit.c  conditionalVariable.c  Desktop    direktori  hello.c    main1
userlinux@Linux:~$ make
make -C /lib/modules/5.19.0-41-generic/build M=/home/userlinux modules
make[1]: Entering directory '/usr/src/linux-headers-5.19.0-41-generic'
warning: the compiler differs from the one used to build the kernel
The kernel was built by: x86_64-linux-gnu-gcc (Ubuntu 11.3.0-1ubuntu1~22.04.1) 11.3.0
You are using:          gcc (Ubuntu 11.3.0-1ubuntu1~22.04.1) 11.3.0
CC [M] /home/userlinux/hello.o
MODPOST /home/userlinux/Module.symvers
CC [M] /home/userlinux/hello.mod.o
LD [M] /home/userlinux/hello.ko
BTF [M] /home/userlinux/hello.ko
Skipping BTF generation for /home/userlinux/hello.ko due to unavailability of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-5.19.0-41-generic'
userlinux@Linux:~$
```

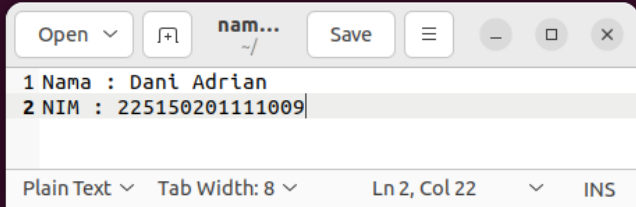


5. Lakukan instalasi kernel module dengan perintah

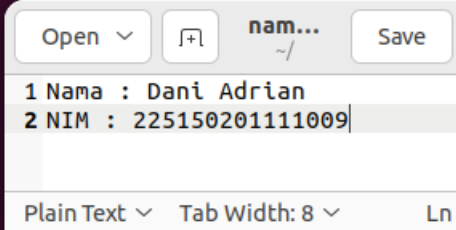
`sudo insmod hello.ko`

Jawab:

```
userlinux@Linux:~$ sudo insmod hello.ko
```



```
userlinux@Linux:~$ sudo insmod hello.ko
[sudo] password for userlinux:
userlinux@Linux:~$
```





LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
userlinux@Linux:~$ ls
alloc-best-fit      alloc-worst-fit      conditionalVariable.c  dir                Downloads          hello.mod.c
alloc-best-fit.c    alloc-worst-fit.c    deadlock              dir.c              hello.c            hello.mod.o
alloc-first-fit     baca                deadlock.c            direktori           hello.ko            hello.o
alloc-first-fit.c  baca.c              Desktop               Documents           hello.mod           join.c
```

Open ▾ [icon] nama prak ~/ Save [icon] [icon] [icon]

1 Nama : Dani Adrian
2 NIM : 225150201111009

6. Cek konten file /var/log/syslog untuk melihat output dari kernel module yang telah dipasang.

Jawab:

```
userlinux@Linux:~$ cd /
```

Open ▾ [icon] nama p ~/

1 Nama : Dani Adrian
2 NIM : 225150201111009


```
userlinux@Linux:/$ sudo less var/log/syslog
```

Open ▾ [icon] nama prak ~/ Sa

1 Nama : Dani Adrian
2 NIM : 225150201111009|



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

```
May 24 08:30:31 Linux systemd[1]: Starting Cleanup of Temporary Directories...
May 24 08:30:32 Linux systemd[1]: systemd-tmpfiles-clean.service: Deactivated successfully.
May 24 08:30:32 Linux systemd[1]: Finished Cleanup of Temporary Directories.
May 24 08:31:17 Linux systemd[1]: Condition check resulted in Run anacron jobs being skipped.
May 24 08:38:01 Linux systemd[1]: Starting Ubuntu Advantage Timer for running repeated jobs...
May 24 08:38:03 Linux systemd[1]: ua-timer.service: Deactivated successfully.
May 24 08:38:03 Linux systemd[1]: Finished Ubuntu Advantage Timer for running repeated jobs.
May 24 08:38:03 Linux systemd[1]: ua-timer.service: Consumed 2.060s CPU time.
May 24 08:45:09 Linux update-notifier[3390]: gtk_widget_get_scale_factor: assertion 'GTK_IS_WIDGET (widget)' failed
May 24 08:45:09 Linux update-notifier[3390]: gtk_widget_get_scale_factor: assertion 'GTK_IS_WIDGET (widget)' failed
May 24 08:48:05 Linux dbus-daemon[643]: [system] Activating via systemd: service name='net.reactivated.Fprint' unit='fprintd.service' requested by '1:80' (uid=1000 label='unconfined')
May 24 08:48:05 Linux systemd[1]: Starting Fingerprint Authentication Daemon...
May 24 08:48:05 Linux dbus-daemon[643]: [system] Successfully activated service 'net.reactivated.Fprint'
May 24 08:48:05 Linux systemd[1]: Started Fingerprint Authentication Daemon.
May 24 08:48:05 Linux gnome-shell[1675]: JS ERROR: Failed to initialize fprintd service: Gio.IOErrorEnum: GDBus.Error:net.reactivated.Fprint.Error.NoSuchDevice: No device found
May 24 08:48:15 Linux NetworkManager[645]: <info> [1684892895.5317] agent-manager: agent[c6b9bd66b1a4fd92,1:80/org.gnome.Shell.NetworkAgent/1000]: agent registered
May 24 08:48:15 Linux ubuntu-appindicators@ubuntu.com[1675]: unable to update icon for software-update-available
May 24 08:48:16 Linux ubuntu-appindicators@ubuntu.com[1675]: unable to update icon for livepatch
May 24 08:48:16 Linux dbus-daemon[1538]: [session uid=1000 pid=1538] Activating service name='org.gnome.ArchiveManager1' requested by '1:120' (uid=1000 pid=4133 cor
ns/ding@rasters" label="unconfined")
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 31 with keysym 31 (keycode a).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 37 with keysym 37 (keycode 10).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 33 with keysym 33 (keycode c).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 35 with keysym 35 (keycode e).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 36 with keysym 36 (keycode f).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 38 with keysym 38 (keycode 11).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 32 with keysym 32 (keycode b).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 39 with keysym 39 (keycode 12).
May 24 08:48:16 Linux gnome-shell[1675]: Window manager warning: Overwriting existing binding of keysym 34 with keysym 34 (keycode d).
May 24 08:48:17 Linux dbus-daemon[1538]: [session uid=1000 pid=1538] Successfully activated service 'org.gnome.ArchiveManager1'
May 24 08:48:17 Linux gnome-shell[1675]: DING: Detected async api for thumbnails
May 24 08:48:17 Linux gnome-shell[1675]: DING: GNOME nautilus 42.2
May 24 08:48:36 Linux systemd[1]: fprintd.service: Deactivated successfully.
May 24 09:00:12 Linux kernel: [ 2688.915015] Goodbye, Kernel!
May 24 09:00:34 Linux kernel: [ 2710.724474] Hello, Kernel!
May 24 09:00:59 Linux gnome-shell[1675]: Window manager warning: last_user_time (2734901) is greater than comparison timestamp (2734860). This most likely represents a window that was active before the system was restarted. Working around...
May 24 09:00:59 Linux gnome-shell[1675]: Window manager warning: W2 appears to be one of the offending windows with a timestamp of 2734901. Working around...
May 24 09:00:59 Linux systemd[1]: Condition check resulted in Daily apt upgrade and clean activities being skipped.
May 24 09:09:43 Linux gnome-shell[1675]: Window manager warning: last_user_time (3258870) is greater than comparison timestamp (3258867). This most likely represents a window that was active before the system was restarted. Working around...
May 24 09:09:43 Linux gnome-shell[1675]: Window manager warning: W2 appears to be one of the offending windows with a timestamp of 3258870. Working around...
May 24 09:09:43 Linux gnome-shell[1675]: Window manager warning: last_user_time (3258870) is greater than comparison timestamp (3258870). This most likely represents a window that was active before the system was restarted. Working around...
May 24 09:09:43 Linux gnome-shell[1675]: Window manager warning: W2 appears to be one of the offending windows with a timestamp of 3258870. Working around...
May 24 09:09:43 Linux gnome-shell[1675]: Window manager warning: W2 appears to be one of the offending windows with a timestamp of 3258870. Working around...
[END]
```

7. Kernel module dapat dihapus dengan perintah

```
sudo rmmod hello
```

Jawab:



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
userlinux@Linux:/$ sudo rmmod hello
userlinux@Linux:/$ lsmod | grep hello
```

Open ▾



nama prak

~/

1 Nama : Dani Adrian

2 NIM : 225150201111009|

Penerapan System Call.

a. System Call **open()**

Jawab:

```
userlinux@Linux:/$ cd ~
userlinux@Linux:~$ nano
```

Open ▾



nama prak

~/

1 Nama : Dani Adrian

2 NIM : 225150201111009|



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
GNU nano 6.2
#include <errno.h>
#include <fcntl.h>
#include <stdio.h>

extern int errno;

int main()
{
    // if file does not have in directory
    // then file foo.txt is created.
    int fd = open("foo.txt", O_RDONLY | O_CREAT);

    printf("fd = %d\n", fd);

    if (fd == -1)
    {
        // print which type of error have in a code
        printf("Error Number %d\n", errno);
        // print program detail "Success or failure"
        perror("Program");
    }
    return 0;
}
```

File Name to Write: open.c

^G Help M-D DOS Format M-
^C Cancel M-M Mac Format M-

```
userlinux@Linux:~$ gcc -o open open.c
userlinux@Linux:~$ ./open
fd = 3/userlinux@Linux:~$ ls | grep foo
foo.txt
userlinux@Linux:~$
```

b. System Call **read()**

Jawab:



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
userlinux@Linux:~$ nano
```

Open ▾

+

*nama prak
~/

```
1 Nama : Dani Adrian  
2 NIM : 225150201111009
```



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
GNU nano 6.2
#include <fcntl.h>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

int main()
{
    char c[42];

    int fd1 = open("foo.txt", O_RDONLY, 0);
    // int fd2 = open("sample2.txt", O_RDONLY, 0);

    read(fd1, &c, 42);
    // read(fd2, &c, 3);

    for (int i = 0; i < sizeof(c); i++)
    {
        printf("%c ", c[i]);
    }
    printf("\n");

    exit(0);
}
```

Open *nama prak
~/

1 Nama : Dani Adrian
2 NIM : 225150201111009|

Plain Text Tab Width: 8

File Name to Write: read.c

^G Help M-
^C Cancel M-

```
userlinux@Linux:~$ gcc -o read read.c
userlinux@Linux:~$ ./read
h e l l o   g e e k s
userlinux@Linux:~$
```

Open

1 Nama : Dani Adrian
2 NIM : 225150201111009



c. System Call **write()**

Jawab:

```
userlinux@Linux:~$ nano

GNU nano 6.2
#include <fcntl.h>
#include <stdio.h>
main()
{
    int sz;
    int fd = open("foo.txt", O_WRONLY | O_CREAT | O_TRUNC, 0644);
    if (fd < 0)
    {
        perror("r1");
        exit(1);
    }
    sz = write(fd, "hello geeks\n", strlen("hello geeks\n"));
    printf("called write(%d, \"hello geeks\\n\\n\", %d).",
           " It returned %d\n",
           fd, strlen("hello geeks\n"), sz);
    close(fd);
}
```

File Name to Write: write.c

^G Help M-D DOS Format
^C Cancel M-M Mac Format



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
userlinux@Linux:~$ nano
userlinux@Linux:~$ gcc -o write write.c
write.c:3:1: warning: return type defaults to 'int' [-Wimplicit-int]
  3 | main()
    | ^~~~~
write.c: In function 'main':
write.c:10:9: warning: implicit declaration of function 'exit' [-Wimplicit-function-declaration]
   10 |     exit(1);
       |     ^~~~~
write.c:3:1: note: include '<stdlib.h>' or provide a declaration of 'exit'
   3 | main()
     | ^~~~~
write.c:10:9: warning: incompatible implicit declaration of built-in function 'exit' [-Wbuiltin-declaration-mismatch]
   10 |     exit(1);
       |     ^~~~~
write.c:10:9: note: include '<stdlib.h>' or provide a declaration of 'exit'
   10 |     exit(1);
       |     ^~~~~
write.c:12:10: warning: implicit declaration of function 'write'; did you mean 'fwrite'? [-Wimplicit-function-declaration]
   12 |     sz = write(fd, "hello geeks\n", strlen("hello geeks\n"));
       |           ^~~~~
write.c:12:37: warning: incompatible implicit declaration of built-in function 'strlen' [-Wbuiltin-declaration-mismatch]
   12 |     sz = write(fd, "hello geeks\n", strlen("hello geeks\n"));
       |                                     ^~~~~
write.c:12:37: note: include '<string.h>' or provide a declaration of 'strlen'
   12 |     sz = write(fd, "hello geeks\n", strlen("hello geeks\n"));
       |                                     ^~~~~
write.c:12:37: note: include '<string.h>' or provide a declaration of 'strlen'
   12 |     sz = write(fd, "hello geeks\n", strlen("hello geeks\n"));
       |                                     ^~~~~
write.c:13:52: warning: format '%d' expects argument of type 'int', but argument 3 has type 'long unsigned int' [-Wformat=]
   13 |     printf("called write(%d, \"hello geeks\n\", %d).",
       |                                           ^~
       |                                           int
       |                                           %d
   14 |         fd, strlen("hello geeks\n"), sz);
       |         ~~~~~
       |         long unsigned int
write.c:16:5: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]
   16 |     close(fd);
       |     ^~~~~
   17 |     pclose
userlinux@Linux:~$
```

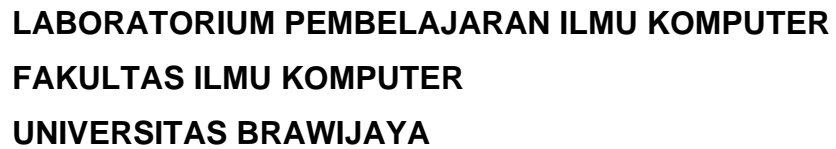
```
userlinux@Linux:~$ ./write
r1: Permission denied
userlinux@Linux:~$ sudo ./write
[sudo] password for userlinux:
called write( 3, "hello geeks\n", 12). It returned 12
userlinux@Linux:~$
```

d. System Call **close()**

Jawab:

```
userlinux@Linux:~$ sudo gcc open.c -o close
userlinux@Linux:~$ ./close
fd = 3/userlinux@Linux:~$
```

9.4. Langkah Praktikum



Jawab:

```
userlinux@Linux:~$ nano keyboard_io.c
GNU nano 6.2 keyboard_io.c
#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/init.h>
#include <linux/keyb...
static struct notifier_block keyboard_notifier;
static int keyboard_event(struct notifier_block *nb, unsigned long event, void *data)
{
    struct keyboard_notifier_param *param = data;
    if (event == KBD_KEYCODE)
    {
        pr_info("Keyboard Event: keycode = %d, value = %d, down = %d\n", param->value, param->down, param->down ? 1 : 0);
        return NOTIFY_OK;
    }
}
static int __init keyboard_io_init(void)
{
    pr_info("Keyboard I/O Module: Initializing\n");
    keyboard_notifier.notifier_call = keyboard_event;
    register_keyboard_notifier(&keyboard_notifier);
    return 0;
}
static void __exit keyboard_io_exit(void)
{
    pr_info("Keyboard I/O Module: Exiting\n");
    unregister_keyboard_notifier(&keyboard_notifier);
}
module_init(keyboard_io_init);
module_exit(keyboard_io_exit);
MODULE_LICENSE("GPL");
MODULE_AUTHOR("Your Name");
MODULE_DESCRIPTION("Keyboard I/O Module");

File Name to Write: keyboard_io.c
^G Help      M-D DOS Format  M-A Append
^C Cancel    M-M Mac Format  M-P Prepend

userlinux@Linux:~$ nano Makefile
Makefile
1 Nama : Dani Adrian
2 NIM : 225150201111009
```




LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

GNU nano 6.2

```
obj-m += keyboard_io.o
```

all:

```
make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules
```

clean:

```
make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

File Name to Write: Makefile

^G Help M-D DOS Format
^C Cancel M-M Mac Format

userlinux@Linux:~\$ make

```
make -C /lib/modules/5.19.0-41-generic/build M=/home/userlinux modules
make[1]: Entering directory '/usr/src/linux-headers-5.19.0-41-generic'
warning: the compiler differs from the one used to build the kernel
The kernel was built by: x86_64-linux-gnu-gcc (Ubuntu 11.3.0-1ubuntu1-22.04.1) 11.3.0
You are using: gcc (Ubuntu 11.3.0-1ubuntu1-22.04.1) 11.3.0
CC [M] /home/userlinux/keyboard_io.o
MODPOST /home/userlinux/Module.symvers
CC [M] /home/userlinux/keyboard_io.mod.o
LD [M] /home/userlinux/keyboard_io.ko
BTF [M] /home/userlinux/keyboard_io.ko
Skipping BTF generation for /home/userlinux/keyboard_io.ko due to unavailability of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-5.19.0-41-generic'
userlinux@Linux:~$
```

userlinux@Linux:~\$ cd /var/log

userlinux@Linux:~\$ sudo insmod keyboard_io.ko

userlinux@Linux:~\$ cd /var/log

userlinux@Linux:~\$ sudo less syslog



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

2. Tugas apa saja yang dilakukan oleh kernel module tersebut dalam kaitannya dengan pengelolaan perangkat I/O? Buktikan dengan menganalisis output file log sistem.

Jawab:

The screenshot shows a terminal window where the user runs the following commands:

```
userlinux@Linux:~$ sudo insmod keyboard_io.ko
userlinux@Linux:~$ cd /var/log
userlinux@Linux:/var/log$ sudo less syslog
```

Below the terminal, a file editor shows the user's name and NIM:

```
1 Nama : Dani Adrian
2 NIM : 225150201111009
```

The terminal output shows the initialization of the keyboard I/O module and a series of keyboard events:

```
May 24 10:42:11 Linux kernel: [ 8807.667863] Keyboard I/O Module: Initializing
May 24 10:42:11 Linux kernel: [ 8807.714939] Keyboard Event: keycode = 28, value = 0, down = 0
May 24 10:42:15 Linux kernel: [ 8811.287555] Keyboard Event: keycode = 46, value = 1, down = 1
May 24 10:42:15 Linux kernel: [ 8811.360993] Keyboard Event: keycode = 46, value = 0, down = 0
May 24 10:42:15 Linux kernel: [ 8811.460095] Keyboard Event: keycode = 32, value = 1, down = 1
May 24 10:42:15 Linux kernel: [ 8811.541316] Keyboard Event: keycode = 32, value = 0, down = 0
May 24 10:42:15 Linux kernel: [ 8811.660974] Keyboard Event: keycode = 57, value = 1, down = 1
May 24 10:42:15 Linux kernel: [ 8811.819639] Keyboard Event: keycode = 57, value = 0, down = 0
May 24 10:42:16 Linux kernel: [ 8812.315667] Keyboard Event: keycode = 14, value = 1, down = 1
May 24 10:42:16 Linux kernel: [ 8812.331499] Keyboard Event: keycode = 14, value = 0, down = 0
May 24 10:42:16 Linux kernel: [ 8812.438511] Keyboard Event: keycode = 14, value = 1, down = 1
May 24 10:42:16 Linux kernel: [ 8812.512909] Keyboard Event: keycode = 14, value = 0, down = 0
May 24 10:42:16 Linux kernel: [ 8812.597916] Keyboard Event: keycode = 14, value = 1, down = 1
May 24 10:42:16 Linux kernel: [ 8812.673043] Keyboard Event: keycode = 14, value = 0, down = 0
May 24 10:42:17 Linux kernel: [ 8812.893298] Keyboard Event: keycode = 54, value = 1, down = 1
May 24 10:42:17 Linux kernel: [ 8812.977784] Keyboard Event: keycode = 54, value = 0, down = 0
May 24 10:42:17 Linux kernel: [ 8813.361814] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:17 Linux kernel: [ 8813.425331] Keyboard Event: keycode = 103, value = 0, down = 0
May 24 10:42:17 Linux kernel: [ 8813.569798] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:17 Linux kernel: [ 8813.641145] Keyboard Event: keycode = 103, value = 0, down = 0
May 24 10:42:18 Linux kernel: [ 8814.147110] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:18 Linux kernel: [ 8814.273342] Keyboard Event: keycode = 103, value = 0, down = 0
May 24 10:42:18 Linux kernel: [ 8814.664971] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:18 Linux kernel: [ 8814.748246] Keyboard Event: keycode = 103, value = 0, down = 0
May 24 10:42:20 Linux kernel: [ 8816.296341] Keyboard Event: keycode = 28, value = 1, down = 1
May 24 10:42:20 Linux kernel: [ 8816.392446] Keyboard Event: keycode = 28, value = 0, down = 0
May 24 10:42:24 Linux kernel: [ 8820.349173] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:24 Linux kernel: [ 8820.440163] Keyboard Event: keycode = 103, value = 0, down = 0
May 24 10:42:25 Linux kernel: [ 8820.904026] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:25 Linux kernel: [ 8821.015210] Keyboard Event: keycode = 103, value = 0, down = 0
May 24 10:42:25 Linux kernel: [ 8821.367844] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:25 Linux kernel: [ 8821.432988] Keyboard Event: keycode = 103, value = 0, down = 0
May 24 10:42:25 Linux kernel: [ 8821.606079] Keyboard Event: keycode = 103, value = 1, down = 1
May 24 10:42:25 Linux kernel: [ 8821.676165] Keyboard Event: keycode = 103, value = 0, down = 0
```

Below the terminal output, a text box contains the following text:

keyboard_io.c berperan dalam membaca setiap input pada keyboard (keyboard event).

3. Jelaskan apa saja yang dilakukan oleh kode kernel module tersebut!

Jawab:

Program diatas menjabarkan seluruh input pada keyboard (keyboard event).
Seluruh log input-an tersebut dapat ditemui di syslog.



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

4. (Soal Bonus) Dengan memanfaatkan system call di atas, modifikasi kernel module sehingga bisa menyimpan kata-kata tertentu (bisa didefinisikan sendiri) ke dalam sebuah file.

Jawab:

```
userlinux@Linux:~$ nano
1 Nama : Dani Adrian
2 NIM : 225150201111009
```



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
GNU nano 6.2
obj-m += modifkernel.o

all:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules
clean:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

Open ▾

nam...
~/

Save

≡

—

□

×

1 Nama : Dani Adrian

2 NIM : 225150201111009|

Plain Text ▾ Tab Width: 8 ▾ Ln 2, Col 22 ▾ INS

File Name to Write: Makefile

^G Help

^C Cancel

M-D DOS Format

M-M Mac Format



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
GNU nano 6.2
#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/fs.h>
#include <linux/uaccess.h>

static int __init modifkernel_init(void)
{
    struct file *file;
    ssize_t ret;

    file = filp_open("SELAMAT HARI LAHIR PANCASILA.txt", O_WRONLY | O_CREAT, 0644);
    if (IS_ERR(file)) {
        pr_err("Failed to create the file\n");
        return PTR_ERR(file);
    }

    ret = kernel_write(file, "modifkernel\n", strlen("modifkernel\n"), &file->f_pos);
    if (ret < 0) {
        pr_err("Failed to write to the file\n");
        goto out;
    }

    filp_close(file, NULL);

    pr_info("modifkernel");
    return 0;

out:
    filp_close(file, NULL);
    return ret;
}

static void __exit modifkernel_exit(void)
{
    pr_info("TANGGAL 1 JUNI 2023");
}

module_init(modifkernel_init);
module_exit(modifkernel_exit);
MODULE_LICENSE("GPL");
MODULE_AUTHOR("Your Name");
MODULE_DESCRIPTION("modifkernel");

File Name to Write: modifkernel.c
^G Help
^C Cancel
```

Open ▾

nam...
~/

Save

≡

1 Nama : Dani Adrian

2 NIM : 225150201111009

Plain Text ▾

Tab Width: 8 ▾

Ln 2, Col 2:

M-D DOS Format

M-M Mac Format



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

```
GNU nano 6.2
obj-m += modifkernel.o

all:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules
clean:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

Open ▾

nam...
~/

Save

≡

—

□

×

1 Nama : Dani Adrian

2 NIM : 225150201111009|

Plain Text ▾ Tab Width: 8 ▾ Ln 2, Col 22 ▾ INS

File Name to Write: Makefile

^G Help

^C Cancel

M-D DOS Format

M-M Mac Format



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

```
userlinux@Linux:~$ ls
alloc-best-fit.c  baca.c  dir.c  hello.mod  kernelmodule.mod  keyboard_io.mod.c  main2.c  mutex.c  public  tolle
alloc-best-fit.c  close  documents  hello.mod.c  kernelmodule.mod.c  keyboard_io.o  Makefile  'nama prak'  read.c  tulis.c
alloc-first-fit.c  conditionalVariable.c  Foo.txt  hello.o  kernelmodule.mod.o  keyboard_io.o  modules.order  nano.35005.save  read.c  Video
alloc-first-fit.c  deadlock.c  join.c  kernelmodule.c  keyboard_io.c  main.c  open  open.c  snap  write.c
alloc-worst-fit.c  deadlock.c  kernelmodule.ko  keyboard_io.ko  main1.c  Module.symvers  Pictures  tugas2.c  write.c
alloc-worst-fit.c  baca  kernelmodule.ko  keyboard_io.mod  main2  Rustic  Praktikum_IX.txt  tugas3.c
userlinux@Linux:~$ make
make -C /lib/modules/5.19.0-41-generic/build M=/home/userlinux/modules
make[1]: Entering directory '/usr/src/linux-headers-5.19.0-41-generic'
warning: the compiler differs from the one used to build the kernel
The kernel was built by: x86_64-linux-gnu-gcc (Ubuntu 11.3.0-1ubuntu1-22.04.1) 11.3.0
You are using: gcc (Ubuntu 11.3.0-1ubuntu1-22.04.1) 11.3.0
CC [M] /home/userlinux/modifkernel.o
MODPOST /home/userlinux/Module.symvers
CC [M] /home/userlinux/modifkernel.mod.o
LD [M] /home/userlinux/modifkernel.ko
BTF [M] /home/userlinux/modifkernel.ko
Skipping BTF generation for /home/userlinux/modifkernel.ko due to unavailability of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-5.19.0-41-generic'
userlinux@Linux:~$
```

```
userlinux@Linux:~$ sudo insmod modifkernel.ko
userlinux@Linux:~$ sudo dmesg
[ 0.000000] Linux version 5.19.0-41-generic (build[lcy02-and64-045] (x86_64-linux-gnu-gcc (Ubuntu 11.3.0-1ubuntu1-22.04.1) 11.3.0))
T_DYNA Tue Apr 18 17:40:00 UTC 2 (Ubuntu 5.19.0-41.42-22.04.1-generic 5.19.17)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-5.19.0-41-generic root=UUID=3e8ce683-ba98-4a27-976e-9976c768a444 ro quiet spl
[ 0.000000] KERNEL supported cpus:
[ 0.000000] Intel GenuineIntel
[ 0.000000] AMD AuthenticAMD
[ 0.000000] Hygon HygonGenuine
[ 0.000000] Centaur CentaurHauls
[ 0.000000] Zhaoxin Shanghai
[ 0.000000] x86/fpu: x87 FPU will use FXSAVE
[ 0.000000] signal: max sigframe size: 1440
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000a0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000007fffff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000007ff000-0x00000000007fffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee00fff] reserved
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] kvm-clock: Using msrs 4b564d01 and 4b564d00
[ 0.000000] kvm-clock: using sched offset of 5114261268 cycles
[ 0.000000] clocksource: kvm-clock: mask: 0xffffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881590591483 ns
[ 0.000000] tsc: Detected 2688.012 MHz processor
[ 0.001342] e820: update [mem 0x00000000-0x00000fff] usable ==> reserved
[ 0.001345] e820: remove [mem 0x000a0000-0x000fffff] usable
[ 0.001348] last_pfn = 0x7ffff0 max_arch_pfn = 0x400000000
[ 0.001382] Disabled
[ 0.001383] x86/PAT: MTRRs disabled, skipping PAT initialization too.
```

```
[ 252.147684] SELAMAT HARI LAHIR PANCASILA
[ 1585.788250] modifkernelinfo
userlinux@Linux:~$
```

1 Nama : Dani Adrian
2 NIM : 225150201111009

```
userlinux@Linux:~$ nano modifkernel.txt
```

Open



nam...

Save

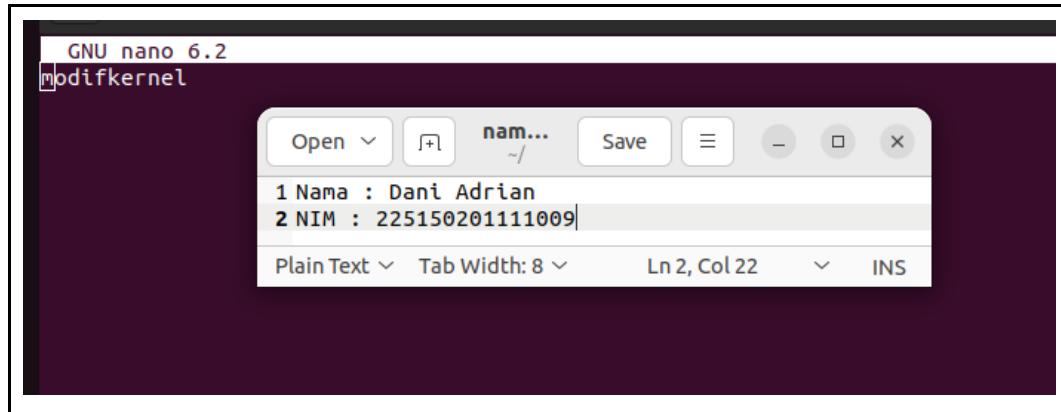


1 Nama : Dani Adrian

2 NIM : 225150201111009



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA



```
1. #include <linux/module.h>
2. #include <linux/kernel.h>
3. #include <linux/fs.h>
4. #include <linux/uaccess.h>
5.
6. static int __init modifkernel_init(void)
7. {
8.     struct file *file;
9.     ssize_t ret;
10.
11.     file = filp_open("modifkernel.txt",
12. O_WRONLY | O_CREAT, 0644);
13.     if (IS_ERR(file)) {
14.         pr_err("Failed to create the file\n");
15.         return PTR_ERR(file);
16.     }
17.     ret = kernel_write(file, "modifkernel\n",
18. strlen("modifkernel\n"), &file->f_pos);
19.     if (ret < 0) {
20.         pr_err("Failed to write to the
21. file\n");
22.         goto out;
23.     }
24.     filp_close(file, NULL);
25.     pr_info("modifkernel");
26.     return 0;
27.
28. out:
```




```
29.     filp_close(file, NULL);
30.     return ret;
31. }
32.
33. static void __exit modifkernel_exit(void)
34. {
35.     pr_info("modifkernel");
36. }
37.
38. module_init(modifkernel_init);
39. module_exit(modifkernel_exit);
40. MODULE_LICENSE("GPL");
41. MODULE_AUTHOR("Your Name");
42. MODULE_DESCRIPTION("modifkernel");
```

9.5. Kesimpulan

Karakteristik Perangkat I/O meliputi :

- **Character-stream (Aliran Karakter):** Perangkat I/O dengan karakteristik ini mentransfer data byte-per-byte, artinya data dikirim dalam bentuk byte tunggal secara berurutan.

Contoh perangkat I/O karakter-stream termasuk keyboard dan printer.

- **Block:** Perangkat I/O dengan karakteristik block mentransfer data dalam blok atau sejumlah byte tertentu. Dalam mode blok, data dikumpulkan dalam buffer sebelum ditransfer.
- Contoh perangkat I/O block adalah hard disk drive.
- **Sequential (Berurutan):** Perangkat I/O yang beroperasi secara sequential membaca atau menulis data dalam urutan berurutan. Data diakses satu per satu, tidak dapat langsung mengakses data di posisi sembarang.
- Contoh perangkat I/O sequential adalah tape drive.
- **Random (Sembarang):** Perangkat I/O yang beroperasi secara random memungkinkan akses data ke lokasi tertentu tanpa harus melewati data di lokasi sebelumnya. Dengan karakteristik ini, kita dapat langsung mengakses data pada posisi yang diinginkan.



Contoh perangkat I/O random termasuk hard disk drive.

- **Synchronous (Sinkron):** Transfer data pada perangkat I/O synchronous terjadi dengan waktu respon yang dapat diprediksi. Proses transfer data menunggu respons dari perangkat I/O sebelum melanjutkan eksekusi.

Contoh perangkat I/O synchronous termasuk pengiriman data melalui jaringan dengan protokol yang menetapkan waktu respons.

- **Asynchronous (Asinkron):** Transfer data pada perangkat I/O asynchronous tidak bergantung pada waktu respon yang dapat diprediksi. Proses transfer data dilakukan secara independen tanpa harus menunggu respons langsung dari perangkat I/O.

Contoh perangkat I/O asynchronous adalah transfer data melalui jaringan menggunakan protokol yang tidak menetapkan waktu respons yang pasti.

- **Sharable (Dapat Dibagikan):** Perangkat I/O yang dapat dibagikan (sharable) dapat digunakan secara bersamaan oleh beberapa entitas atau proses. Lebih dari satu entitas dapat mengakses dan menggunakan perangkat I/O tersebut pada saat yang sama.
- **Read-write (Baca-Tulis):** Perangkat I/O yang mendukung operasi baca dan tulis memungkinkan penggunaan perangkat untuk membaca dan menulis data. Data dapat dibaca dari perangkat dan juga ditulis ke dalam perangkat tersebut.
- **Readonly (Hanya Baca):** Perangkat I/O dengan karakteristik readonly hanya mendukung operasi baca. Data hanya dapat dibaca dari perangkat tersebut dan tidak dapat ditulis ke dalamnya.

Contohnya adalah CD-ROM atau file sistem yang dipasang sebagai readonly.

- **Writeonce (Tulis Sekali):** Perangkat I/O dengan karakteristik writeonce hanya memungkinkan data ditulis sekali ke dalamnya. Setelah data ditulis, tidak dapat diubah atau ditimpa.



Contohnya adalah CD-R (Compact Disc-Recordable) yang memungkinkan data ditulis sekali dan tidak dapat diubah lagi setelahnya.

Karakteristik-karakteristik ini memberikan pemahaman tentang cara perangkat I/O beroperasi dan fitur-fitur yang mereka miliki

- **Driver Perangkat:** Kernel module digunakan untuk mengimplementasikan driver perangkat yang berkomunikasi dengan perangkat keras, seperti keyboard, mouse, kartu jaringan, dan perangkat penyimpanan. Modul kernel ini menyediakan antarmuka dan logika yang diperlukan untuk mengenali dan berinteraksi dengan perangkat keras yang terhubung ke sistem. Mereka memfasilitasi komunikasi antara perangkat keras dan perangkat lunak di tingkat kernel, sehingga memungkinkan sistem operasi dan aplikasi menggunakan dan mengontrol perangkat keras tersebut.
- **Sistem File:** Kernel module memungkinkan penggunaan berbagai sistem file, seperti ext4, XFS, NTFS, yang memungkinkan akses dan pengelolaan file dan direktori. Modul kernel ini berisi implementasi sistem file spesifik yang mendefinisikan cara penyimpanan dan pengambilan data pada media penyimpanan. Mereka menyediakan antarmuka dan fungsi-fungsi yang digunakan oleh sistem operasi dan aplikasi untuk membaca, menulis, membuat, menghapus, dan mengelola file serta direktori di sistem.
- **Pengelolaan Jaringan:** Modul jaringan di kernel digunakan untuk mengimplementasikan protokol jaringan dan menangani komunikasi jaringan antara sistem. Modul jaringan ini berisi implementasi protokol seperti TCP/IP, UDP, ICMP, dan lainnya. Mereka memungkinkan sistem untuk berkomunikasi dengan perangkat jaringan seperti kartu jaringan, router, dan switch. Modul jaringan ini mengatur pengiriman dan penerimaan paket data, mengelola koneksi jaringan, serta menangani aspek keamanan dan pengaturan jaringan lainnya.
- **Fungsionalitas Tambahan:** Kernel module juga dapat digunakan untuk menyediakan fungsionalitas tambahan, seperti pengoptimalan kinerja, pengaturan keamanan, pengelolaan daya, dan banyak lagi. Modul tambahan ini dapat diperkenalkan ke dalam kernel untuk meningkatkan kinerja sistem, mengimplementasikan fitur-fitur baru, atau menyediakan



abstraksi dan layanan tambahan kepada aplikasi yang berjalan di atasnya. Misalnya, modul kernel dapat ditambahkan untuk mengimplementasikan mekanisme caching, enkripsi data, manajemen daya untuk perangkat seluler, atau fungsionalitas lainnya yang tidak termasuk dalam kernel dasar.

Kernel module memberikan fleksibilitas dan kemampuan untuk memperluas dan memodifikasi fungsionalitas inti dari kernel Linux. Mereka memungkinkan pengguna dan pengembang untuk menyesuaikan sistem operasi sesuai kebutuhan mereka dan memperluas kemampuan kernel tanpa harus mengubah kode sumber kernel yang sudah ada.

Selain itu, ada juga perintah-perintah lain yang digunakan dalam pengelolaan I/O pada sistem operasi Linux. Beberapa perintah yang umum digunakan adalah:

- System call `open()`: System call `open()` digunakan untuk membuka file atau perangkat dalam mode tertentu. Dengan menggunakan system call ini, aplikasi dapat mengakses file untuk membaca, menulis, atau melakukan operasi lainnya. System call `open()` akan mengembalikan file descriptor yang digunakan untuk mengidentifikasi file yang telah dibuka.
- System call `write()`: System call `write()` digunakan untuk menulis data ke dalam file atau perangkat. Dengan menggunakan system call ini, aplikasi dapat mengirim data dari buffer ke file atau perangkat yang ditentukan. Data yang ditulis akan ditransfer dalam bentuk byte atau blok tergantung pada karakteristik perangkat I/O yang sedang digunakan.
- System call `close()`: System call `close()` digunakan untuk menutup file atau perangkat yang telah dibuka sebelumnya. Dengan menggunakan system call ini, aplikasi dapat mengakhiri akses terhadap file atau perangkat, dan melepaskan sumber daya yang terkait dengan file descriptor yang digunakan.
- System call `read()`: System call `read()` digunakan untuk membaca data dari file atau perangkat. Dengan menggunakan system call ini, aplikasi dapat membaca data dari file atau perangkat yang ditentukan ke dalam buffer yang telah disediakan. Data yang dibaca akan ditransfer dalam



LABORATORIUM PEMBELAJARAN ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

bentuk byte atau blok tergantung pada karakteristik perangkat I/O yang sedang digunakan.

- Selain system call yang terkait dengan I/O, juga terdapat keyboard I/O. keyboard I/O mencakup interaksi antara pengguna dan sistem melalui keyboard. Ketika pengguna menekan tombol-tombol pada keyboard, data karakter akan dikirim ke sistem, dan sistem akan merespons sesuai dengan perintah atau aksi yang sesuai.

Untuk memantau log terkait dengan perintah-perintah I/O dan keyboard I/O, dapat digunakan file `syslog` di sistem operasi Linux. `syslog` adalah suatu sistem logging yang menyimpan dan mengelola pesan log dari berbagai sumber, termasuk sistem operasi dan aplikasi. Dengan memeriksa file `syslog`, pengguna dapat melihat informasi terkait dengan perintah-perintah I/O yang dieksekusi dan kejadian-kejadian yang terjadi melalui keyboard.