

1 SYSTEMS PROGRAMMING LABORATORY VII - IPC II & Devices

Examples&Exercises:

- Complete the following codes if necessary, then compile and run the code.
- Analyze the code and output.

1. Local Sockets; [socket-server.c](#), [socket-client](#)

- Study and analyze the code.
- During/after execution, give the following commands

```
$ ./socket-server /yourdirectory/socketname
$ ./socket-client /yourdirectory/socketname "Hello, world."
$ ./socket-client /yourdirectory/socketname "This is a test."
$ ./socket-client /yourdirectory/socketname "quit"
```
- Modify the code to correct as return non-zero if the client sent a "quit" message.

2. Internet-Domain Sockets; [socket-inet.c](#)

- Study and analyze the code.
- During/after execution, give the following command

```
$ ./socket-inet siber.cankaya.edu.tr
```
- Change the content of the "buffer" (not GET),
- Change the port number (not 80),
- Try some other addresses (e.g., ari.cankaya.edu.tr, www.cankaya.edu.tr, www.metu.edu.tr); what could be the reason if it does not succeed?

3. *About Devices*; study the following commands:

- Device Entries

```
$ mknod /yourdirectory/yourdevice ? ? ?
$ ls -l yourdevice
$ rm /yourdirectory/yourdevice
```
- The /dev Directory

```
$ ls -l /dev/hda /dev/hda1
```

```
$ ls -l /dev/lp0
```

- What are the *major* and *minor* numbers of these devices?
- How to understand a device as if block or character device?
- What about special devices (which are not hardware devices).

```
$ ls -l /dev/
```

- Accessing Devices by Opening Files

```
$ cat /usr/share/sndconfig/sample.au > /dev/audio
```

```
$ cat yourdocument.txt > /dev/lp0
```

- This is the general way to access to a device.
- You should have privileged to access to printer.
- Be Super User and try again.

```
$ cat yourdocument.txt >> logfile 2>&1
```

What about *2* and *1*, which devices are these?

- /dev/null

```
$ verbose_command > /dev/null
```

Where for the *verbose_command* use any command (e.g., *ls*).

```
$ cp /dev/null empty-file
```

```
$ ls -l empty-file
```

- /dev/zero

```
$ hexdump -v /dev/zero
```

```
$ hexdump anyfile
```

- /dev/full

```
$ cp /etc/fstab /dev/full
```

- Random Number Devices

```
$ od -t a (d2,x1) /dev/random
```

```
$ od -t x1 /dev/urandom
```

```
$ man od
```

- Loopback Devices

```
$ dd if=/dev/zero of=/yourdirectory/disk-image count=20480
$ ls -l /yourdirectory/disk-image
$ mke2fs -q /yourdirectory/disk-image
$ mkdir /yourdirectory/virtual-fs
$ mount -o loop=/dev/loop0 /yourdirectory/disk-image
  /yourdirectory/virtual-fs
$ df -h /yourdirectory/virtual-fs
$ cd /yourdirectory/virtual-fs
$ echo "Hello, world!" > test.txt
$ ls -l
$ cat test.txt
$ cd /yourdirectory
$ umount /yourdirectory/virtual-fs
```

4. **TO BE GRADED;** Write a C program to access a device by opening file.
 - Say, you are sending a file to the printer,
 - Open a device and return to a file descriptor,
 - Write some buffer to this file descriptor,
 - Test your code as privileged user.
5. **TO BE GRADED;** Write a C program to use random numbers;
 - Complete the [code](#).