

```

erfan@erfan-virtual-machine:~/Desktop/ECS/CA2/P5$ ./test.sh
make -C /lib/modules/5.15.0-139-generic/build M=/home/erfan/Desktop/ECS/CA2/P5 modules
make[1]: Entering directory '/usr/src/linux-headers-5.15.0-139-generic'
warning: the compiler differs from the one used to build the kernel
The kernel was built by: gcc (Ubuntu 9.4.0-1ubuntu1~20.04.2) 9.4.0
You are using:          gcc (Ubuntu 10.5.0-1ubuntu1~20.04) 10.5.0
CC [M] /home/erfan/Desktop/ECS/CA2/P5/LIFO_Driver.o
MODPOST /home/erfan/Desktop/ECS/CA2/P5/Module.symvers
CC [M] /home/erfan/Desktop/ECS/CA2/P5/LIFO_Driver.mod.o
LD [M] /home/erfan/Desktop/ECS/CA2/P5/LIFO_Driver.ko
BTF [M] /home/erfan/Desktop/ECS/CA2/P5/LIFO_Driver.ko
Skipping BTF generation for /home/erfan/Desktop/ECS/CA2/P5/LIFO_Driver.ko due to unavailability of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-5.15.0-139-generic'
-----
Correct write and read:
Wrote: Hello
Wrote: World
Read: dlroWolleH
-----
Read from empty lifo:
Read:
-----
Read from write-only:
read: Invalid argument
-----
Write to read-only:
write: Invalid argument
-----
make -C /lib/modules/5.15.0-139-generic/build M=/home/erfan/Desktop/ECS/CA2/P5 clean
make[1]: Entering directory '/usr/src/linux-headers-5.15.0-139-generic'
CLEAN /home/erfan/Desktop/ECS/CA2/P5/Module.symvers
make[1]: Leaving directory '/usr/src/linux-headers-5.15.0-139-generic'
rm -f
erfan@erfan-virtual-machine:~/Desktop/ECS/CA2/P5$

```

(۱) در این تمرین با Character device driver آشنا شدیم. این نوع از درایورها داده‌ها را به‌صورت کاراکتر به کاراکتر بین سخت‌افزار و فضای کاربر منتقل می‌کنند. اما به‌جز این‌ها، انواع دیگری از درایورها نیز وجود دارند. از جمله Block device driver ها که داده‌ها را به‌صورت بلوک‌هایی از اطلاعات (مانند دیسک سخت) مدیریت می‌کنند، و Network device driver ها که برای مدیریت دستگاه‌های شبکه مثل کارت شبکه کاربرد دارند. دلیل این تنوع درایورها، تفاوت در نحوه ارتباط و مدیریت داده‌ها بین انواع مختلف سخت‌افزارهاست. هر سخت‌افزار نیازهای خاصی در تعامل با سیستم‌عامل دارد، و درایورها نقش واسطه بین آن‌ها و کرنل را ایفا می‌کنند.

(۲) درایورهای دستگاه برای شناسایی و آدرس‌دهی به دستگاه‌های مختلف از دو عدد به نام‌های Major number و Minor number استفاده می‌کنند. Major number نشان می‌دهد که کدام درایور باید

به درخواست‌ها رسیدگی کند، یعنی مشخص‌کننده نوع یا کلاس درایور است. در مقابل، Minor number به صورت دقیق‌تری مشخص می‌کند که کدام یک از دستگاه‌های خاص آن کلاس مورد نظر است (مثلاً کدام یک از چند پورت سریال). این دو شماره باعث می‌شوند سیستم عامل بتواند به طور دقیق، دستگاه‌های مختلف را شناسایی کرده و مدیریت کند، به خصوص زمانی که چندین دستگاه مشابه در سیستم وجود دارند.