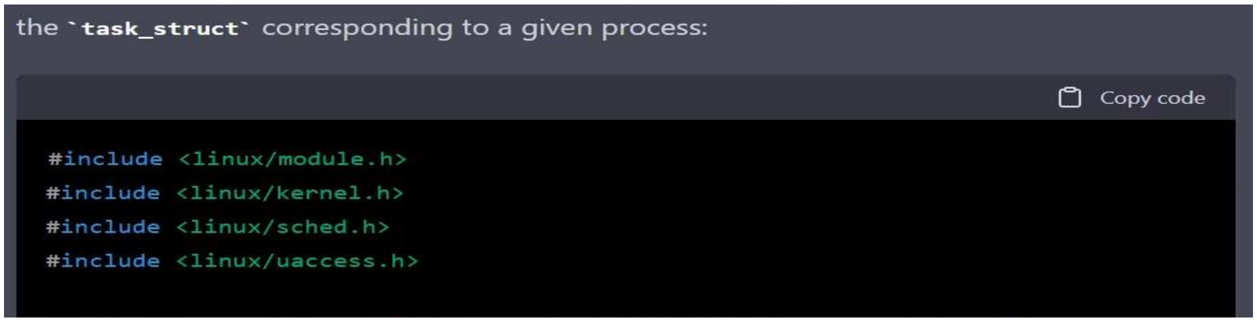
This exercise requires you to write your own small kernel module. You require to implement a kernel system call as a module. The task of the system call would be to read the entries of the process task_struct corresponding to any given process (supplied as input via command line argument) and prints the values of the following field: pid, user id, process group id (pgid) and command Path.

Firstly , we have to write module kernel system call using header file like this :



the `task_struct` corresponding to a given process:

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

And write code to print pid , uid , pgid and command line of any process in it.

Secondly make MAkefile to module install in your kernel :

```
obj-m += task.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
```

Task.o is file name of file or code of module kernel

After that many different file is made

 Write two pieces of command to install and set parameter to the code

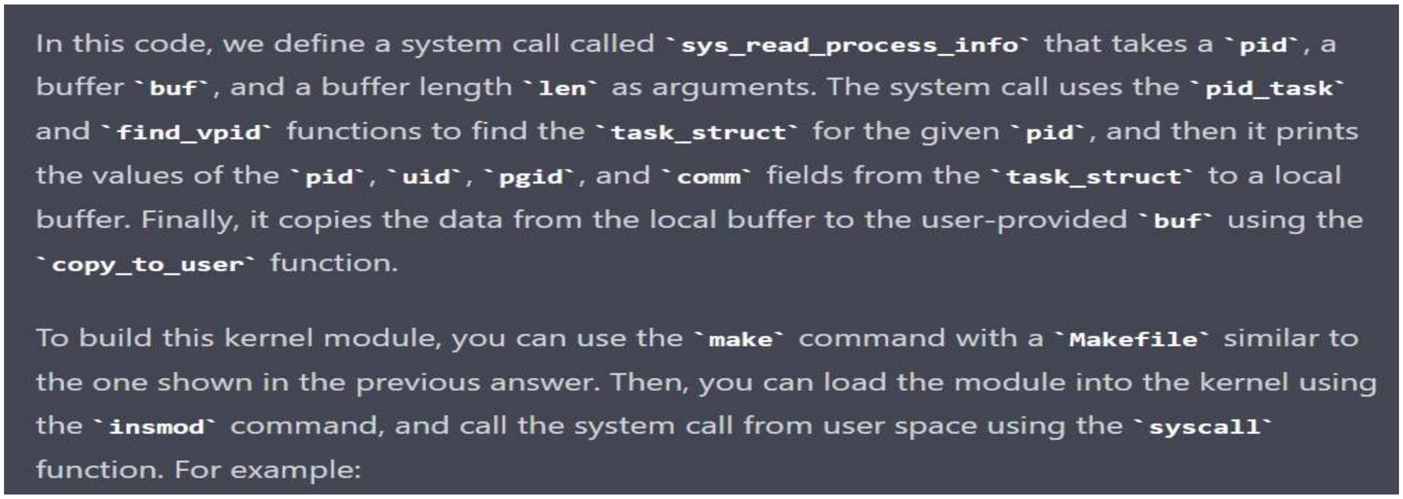
Insmod task.ko myint =987 or pid =PID

dmesg | tail

Than remove command to remove or un install the kernel module

rmmod task.ko

dmesg | tail



In this code, we define a system call called `sys_read_process_info` that takes a `pid`, a buffer `buf`, and a buffer length `len` as arguments. The system call uses the `pid_task` and `find_vpid` functions to find the `task_struct` for the given `pid`, and then it prints the values of the `pid`, `uid`, `pgid`, and `comm` fields from the `task_struct` to a local buffer. Finally, it copies the data from the local buffer to the user-provided `buf` using the `copy_to_user` function.

To build this kernel module, you can use the `make` command with a `Makefile` similar to the one shown in the previous answer. Then, you can load the module into the kernel using the `insmod` command, and call the system call from user space using the `syscall` function. For example: