**DOCUMENTATION**

The system call takes the PID of the required process as well as name of the destination file.

The task\_struct structure defined in sched.h stores all the details of every process that exists in the system, and all the processes in turn are stored in a circular double linked list which are traversed by for\_each\_process.

The code matches the given pid with pid of every other process in the linked list. If no such process exists, then errno ESRCH which signifies “No such process” is returned. Else corresponding PID, Process Name and State is returned.

Next the code performs file-handling to write the output in the given file. If no such file exists, either a new file is created with the same name or errno ENOENT which signifies “No such file or directory” is returned.

To handle the address space mismatch, get\_fs() and set\_fs() are used. The writing functionality is implemented by using fget, fput and vfs\_write functions.

The tester functions first ask for the process PID and then the filename. Process info is returned if correct PID is given.