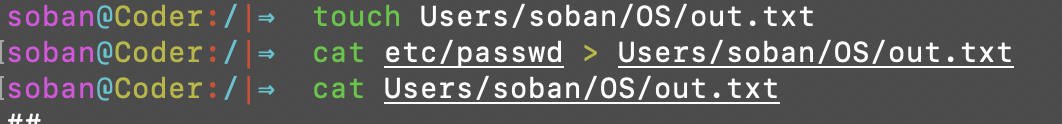
 **Operating System Lab#04**

# I/O Redirection

**Task 01:** Perform the following tasks:

1. Write a single command to copy the contents of **/etc/passwd** into **out.txt** without using **cp** command. (Hint: I/O Redirection)



cat etc/passwd > Users/soban/OS/out.txt

cat /etc/passwd > out.txt: This uses the cat command to read the contents of /etc/passwd and then redirects (>) the output to out.txt.

1. Find all the files named \*libc.so\* in your root directory (using find command) and redirect the output to libc\_locations.txt, and errors to /dev/null.

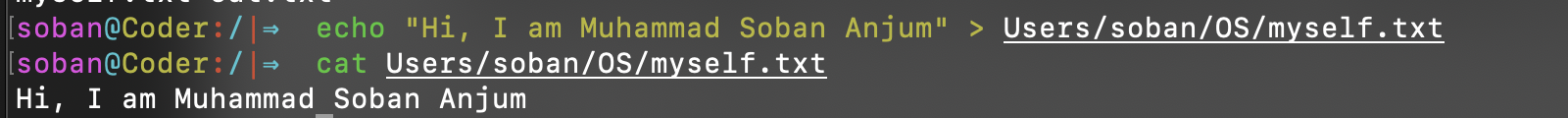


find / -name '\*libc.so\*' > libc\_locations.txt 2>/dev/null

* find / -name '\*libc.so\*': This searches for files named \*libc.so\* starting from the root directory (/).
* > libc\_locations.txt: This redirects the standard output (the list of found files) to libc\_locations.txt.
* 2>/dev/null: This redirects the standard error (any errors encountered during the find operation, such as permission denied errors) to /dev/null, effectively discarding them.

**Task 02:** Perform the following tasks:

1. Write a single command to add “**Hi, I am <Your Name>**” into myself.txt. (Hint: Use echo command)



echo "Hi, I am Muhammad Soban Anjum" > Users/soban/OS/myself.txt

* echo "Hi, I am Muhammad Soban Anjum": This prints the text to the standard output.
* > myself.txt: This redirects the standard output to myself.txt, creating the file if it doesn’t exist or overwriting it if it does.

1. Append “**My Roll No is : <Your Roll No.>**” using IO redirection.



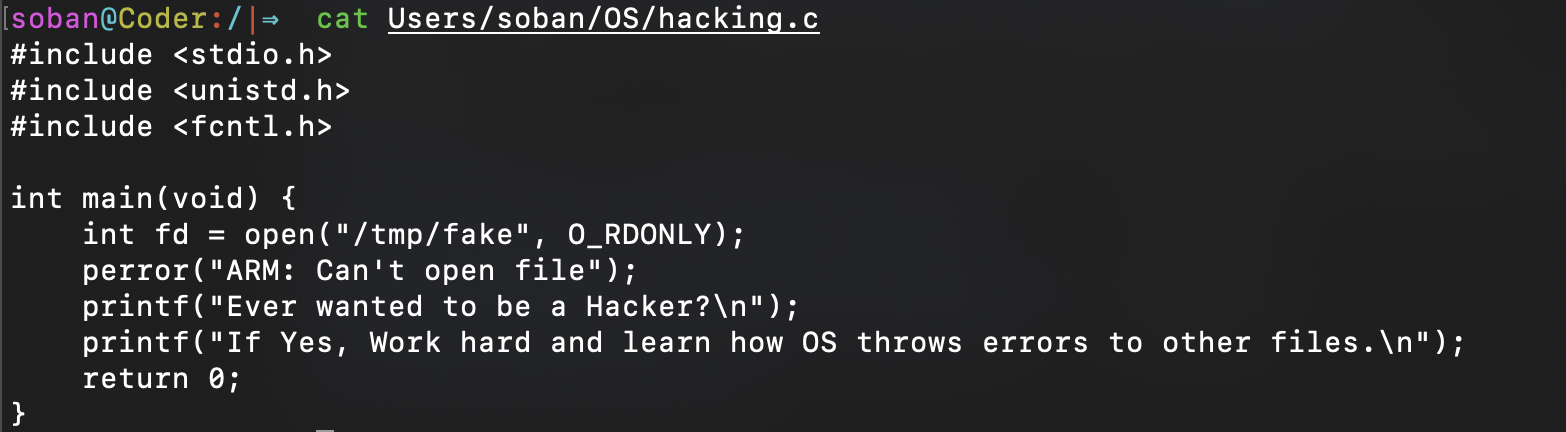
echo "My Roll No is : BSDSF21A007" >> Users/soban/OS/myself.txt

* echo "My Roll No is : BSDSF21A007": This prints the text to the standard output.
* >> myself.txt: This appends the standard output to myself.txt, creating the file if it doesn’t exist or appending to it if it does.

**Task 03:**

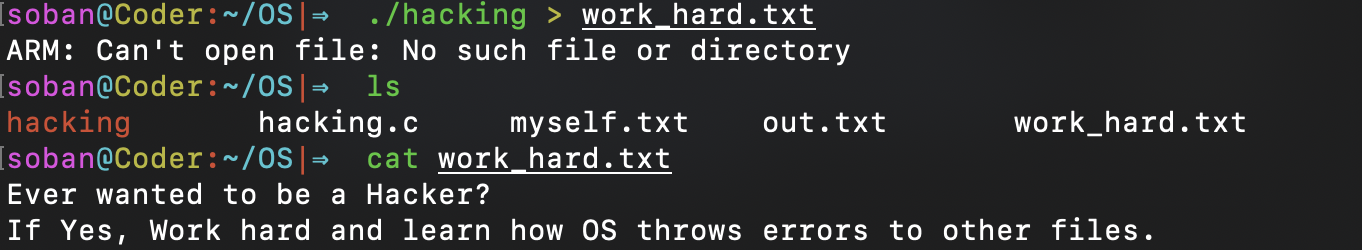
|  |
| --- |
| #include<stdio.h>  #include<unistd.h> #include<fcntl.h>  **int main**(**void**) { **int** fd = open("/tmp/fake", O\_RDONLY); perror("ARM: Can't open file"); printf("Ever wanted to be a Hacker?\n"); printf("If Yes, Work hard and learn how OS throws errors to other files.\n"); **return** 0;  } |

Save the above given source code as **hacking.c**. Compile and make executable of the **hacking.c** and perform I/O redirection operations as described below:



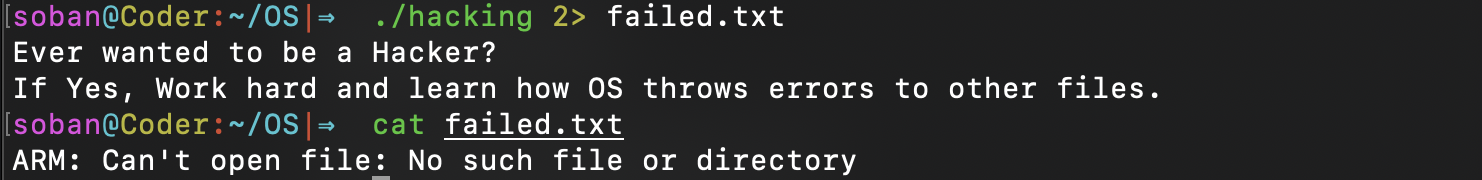


1. Redirect the output to a file named work\_hard.txt.



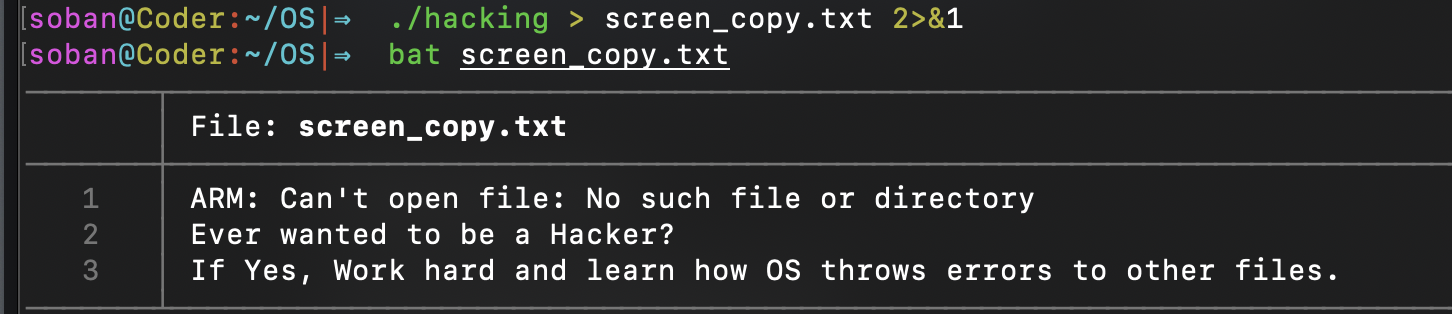
redirects the standard output (stdout) of the program to work\_hard.txt.

1. Redirect the error to a file named failed.txt**.**



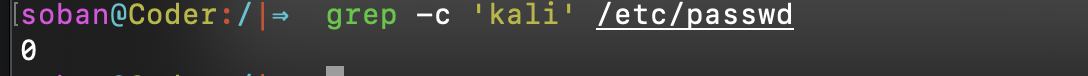
2> redirects the standard error (stderr) of the program to failed.txt.

1. Redirect the stdout and stderr to a file called screen\_copy.txt using **copy descriptor**.



* > redirects the standard output (stdout) of the program to screen\_copy.txt.
* 2>&1 redirects the standard error (stderr) to the same location as the standard output (stdout), which is screen\_copy.txt.

# Pipes/Fifos

1. Write a single command to display all the lines containing **kali** in **/etc/passwd** counts the number of lines in the output. 



grep 'kali' /etc/passwd finds all lines in /etc/passwd containing the string kali. “|” is a pipe that takes the output of the first command and feeds it as input to the next command. “wc -l” counts the number of lines in the input it receives.

1. Write a single command to count the occurrences of word **root** in **/etc/passwd.**



grep -o 'root' /etc/passwd finds all occurrences of the word root in /etc/passwd, each on a new line. “|” is a pipe that takes the output of the first command and feeds it as input to the next command. “wc -l” counts the number of lines (occurrences) in the input it receives.

1. Draw PPFDT of each process that was created in the above questions.

//task1

/etc/passwd 🡪 grep 🡪 (output lines containing “kali”)

Grep 🡪 wc -l 🡪 (count of line)

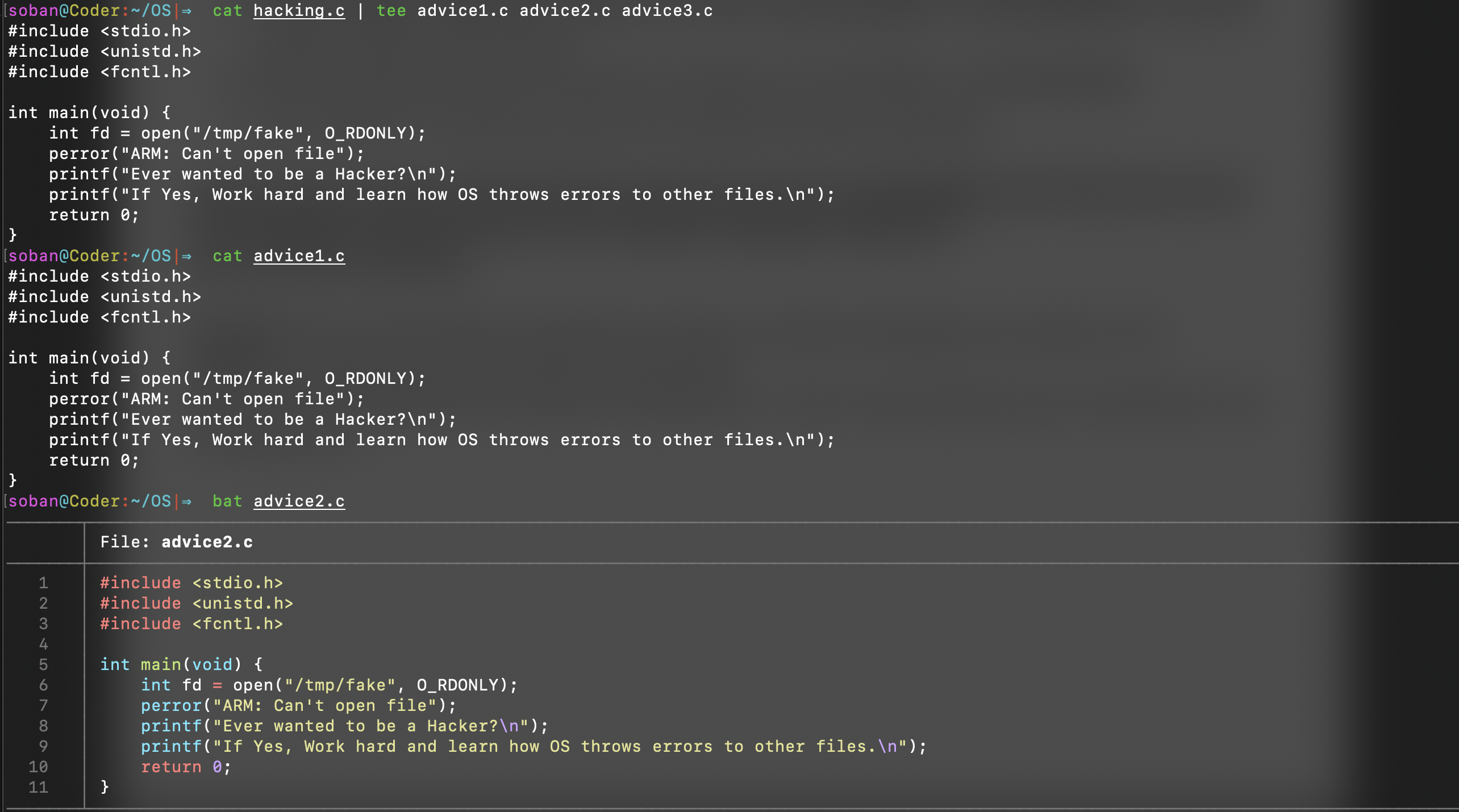
//task2

| /etc/passwd 🡪| grep | 🡪 (each occurrence of "root" on a new line)

Grep 🡪 | wc -l| 🡪 (count of occurrences)

**Task 03:** Write a single command which will copy the source code **hacking.c** (I/O Redirection: Task 03) on **stdout** and in the following files: **advice1.c, advice2.c, advice3.c.**

(Hint: use tee command)



* cat hacking.c: This command reads the contents of hacking.c and outputs it to stdout.
* |: This is the pipe operator, which takes the output of cat hacking.c and feeds it as input to the tee command.
* tee advice1.c advice2.c advice3.c: The tee command reads from stdin and writes to stdout and to the specified files (advice1.c, advice2.c, advice3.c).

**Task 04:** Create a fifo called transporter**.** Open two shells and display the contents of the **advice1.c (**created in the above task) on both shells.

(Hint: Use tee command to save data in transporter and on second shell use any command to read transporter)

In shell 1



In shell 2



* **Shell 1**:
  + cat advice1.c: This reads the contents of advice1.c.
  + |: The pipe operator passes the output of cat advice1.c to tee.
  + tee transporter: This writes the output to transporter and also prints it to stdout, displaying it in Shell 1.
* **Shell 2**:
  + cat transporter: This reads the contents from the FIFO transporter and displays it on Shell 2.