

This In Class Activity is due at the end of class on today.

Student 1 name **and** ODIN id: Tyler Taormina taormina

Student 2 name **and** ODIN id: James Truong jatruong

Student 3 name **and** ODIN id: Luke Babcock lbabcock

1. What are 2 file descriptors that are open when your process begins?

- a) 0 (stdin)
- b) 1 (stdout)
- c) 2 (stderr)

2. In order to ensure that a file that does not yet exist is created when opening a file for write, you must include what in the flags for the `open()` function?

The `O_CREAT` flag must be included.

3. What do you include in the flags to the `open()` function to have to file automatically set to zero bytes upon opening?

The `O_TRUNC` flag will truncate it to zero bytes.

4. For the purposes of access and protection, Unix divides the world into three categories:

- a) User mode
- b) Group
- c) Other

5. If the octal value for the mode on a file is 755, what is the mode in human form? Consider trying the `chmod` calculator: <https://chmod-calculator.com/>  
rwxr-xr-x

6. What are the 3 valid values for the `whence` argument in the `lseek()` function? You might consider looking at the `manpage`.

- a) `SEEK_SET`
- b) `SEEK_CUR`
- c) `SEEK_END`

7. What does a successful call to the `write()` function return?

The number of bytes actually written.

8. What does a return value of 0 (zero) for the `read()` function indicate?

End-of-file

9. Describe the effect of the following call on an open file descriptor, `in_fd`.

`lseek(in_fd, -100, SEEK_CUR);`

- Sets the offset -100 bytes (before) from the current offset
- `SEEK_CURR` is not an option. There Should be one "R".

10. Are parallelism and concurrency the same thing? Why/Why not?

No they are not the same. Parallelism uses multiple hardware units in order to complete computation more quickly by dividing tasks to execute at the same time (in parallel) whereas concurrency uses multiple streams of execution that execute independently in an interleaved manner. Concurrency can execute on a single processor or multiple.

11. For our class, what is a race condition?

A situation where the result produced by two (or more) processes or threads (streams of execution) operating on a shared resource depends in an unexpected way on the relative order in which the processes are scheduled on the CPU(s).