Worksheet: Named Pipes (FIFOs) in Linux

# Section A: Fill in the blanks

1. A named pipe in Linux is also known as \_\_\_\_\_\_\_\_\_\_.

2. The system call used to create a named pipe is \_\_\_\_\_\_\_\_\_\_.

3. Named pipes are represented as special files of type \_\_\_\_\_\_\_\_\_\_ in the file system.

4. The command used in the terminal to create a named pipe is \_\_\_\_\_\_\_\_\_\_.

5. A named pipe allows communication between processes that are \_\_\_\_\_\_\_\_\_\_ related.

# Section B: Multiple Choice Questions

6. What is the file system type of a named pipe?

* a) Regular File
* b) Directory
* c) FIFO
* d) Character Device

7. Which of the following commands creates a named pipe?

* a) touch pipe
* b) mkfifo pipe
* c) mknod pipe p
* d) Both b and c

8. Which system call is used to open a named pipe?

* a) fopen
* b) creat
* c) open
* d) pipe

9. Named pipes can be used for communication between:

* a) Threads
* b) Unrelated processes
* c) Only parent-child processes
* d) Only in same process

10. Which header file is required to use mkfifo() system call?

* a) <stdio.h>
* b) <stdlib.h>
* c) <sys/stat.h>
* d) <unistd.h>

# Section C: Short Answer Questions

11. What is the difference between a named pipe and an unnamed pipe?

12. Write the command to create a named pipe named `myfifo`.

13. How do you write to a named pipe from one terminal?

14. How do you read from a named pipe from another terminal?

15. What are some limitations of named pipes in inter-process communication?

# Bonus: Coding Task

16. Write a simple C program where one process writes to a named pipe and another reads from it.  
Hint: Use `mkfifo()`, `open()`, `write()`, `read()`, and `close()`.