**INTER PROCESS COMMUNICATION**

#include <stdio.h>

#include <unistd.h>

#include <string.h>

int main() {

int fd[2]; // fd[0]: read end, fd[1]: write end

pid\_t pid;

char write\_msg[] = "Hello from parent!";

char read\_msg[100];

// Create a pipe

if (pipe(fd) == -1) {

perror("pipe");

return 1;

}

pid = fork();

if (pid < 0) {

perror("fork");

return 1;

}

if (pid > 0) {

// Parent process

close(fd[0]); // Close reading end

write(fd[1], write\_msg, strlen(write\_msg) + 1);

close(fd[1]);

} else {

// Child process

close(fd[1]); // Close writing end

read(fd[0], read\_msg, sizeof(read\_msg));

printf("Child received: %s\n", read\_msg);

close(fd[0]);

}

return 0;

}