

National University of Computer and Emerging Sciences



**Laboratory Manual**  
*for*  
**Operating Systems Lab**  
**(CL-220)**

Course Instructor	Ms. Namra Absar
Lab Instructor(s)	Ms. Saba Tariq
Section	BCS-4A
Semester	Spring 2024

Department of Computer Science  
FAST-NU, Lahore, Pakistan

## Objectives:

- Input and output redirection using Dup,Dup2 system calls.
- Use of pipe and dup together

## 1 “dup” System Call Manual

### NAME

**dup**- Duplicate an open file descriptor

### SYNOPSIS

```
#include<unistd.h>
int dup(int oldfd);
```

### DESCRIPTION

The **dup** system call creates a new file descriptor that refers to the same open file description as the **oldfd** File descriptor. The new file descriptor is the lowest-numbered available descriptor.

### PARAMETERS

- **oldfd**: The file descriptor to be duplicated.

### RETURN VALUE

- On success, **dup** returns a new file descriptor that refers to the same file as **oldfd**. If an error occurs, it returns -1, and **errno** is set to indicate the error.

### ERRORS

- **EBADF**: **oldfd** is not a valid file descriptor.
- **EMFILE**: The process has too many open file descriptors.

## 2 “dup2” System Call Manual

### NAME

**dup2**- Duplicate an open file descriptor to a specified file descriptor number

### SYNOPSIS

```
#include<unistd.h>
```

```
int dup2(int oldfd, int newfd) ;
```

## DESCRIPTION

The `dup2` system call duplicates the file descriptor `oldfd` to `newfd`, allowing you to specify a particular file descriptor number for the duplication. If `newfd` is already in use, it is closed before the duplication occurs.

## PARAMETERS

- **oldfd**: The file descriptor to be duplicated.
- **newfd**: The desired file descriptor number for the duplication.

## RETURNVALUE

On success, **dup2** returns **newfd**, which is the duplicated file descriptor. If an error occurs, it returns -1, and **errno** is set to indicate the error.

## ERRORS

- **EBADF**: **oldfd** is not a valid file descriptor, or **newfd** is negative or exceeds the maximum allowed file descriptor value.
- **EMFILE**: The process has too many open file descriptors.

# Lab Tasks

## Question1: Using the dup Function

Write a C/C++ program that reads text from a file named "original.txt" and converts all the characters to lowercase. Then, update the content of the file with the modified text, ensuring that all letters are in lowercase. Utilize the **dup** function to duplicate the file descriptor for performing the file operations. Ensure proper error handling throughout the program.

## Question2: Using the dup2 Function

Develop a C/C++ program to find prime numbers within a given range specified in a file named "input.txt". The file contains two numbers separated by a space, indicating the start and end of the range, respectively. Instead of using read or write system calls for file operations, employ the **dup2** function to provide these two numbers as input via standard input (stdin). Similarly, redirect the output generated on the terminal to a file named "output.txt" without using read or write system calls. Ensure error handling for file operations and follow proper coding standards.

## Question3:

Write a cpp program that can perform the task of the below shell command.

```
man ls | grep ls > file.txt
```

(Hint: Pipes, execlp and dup2 system calls)