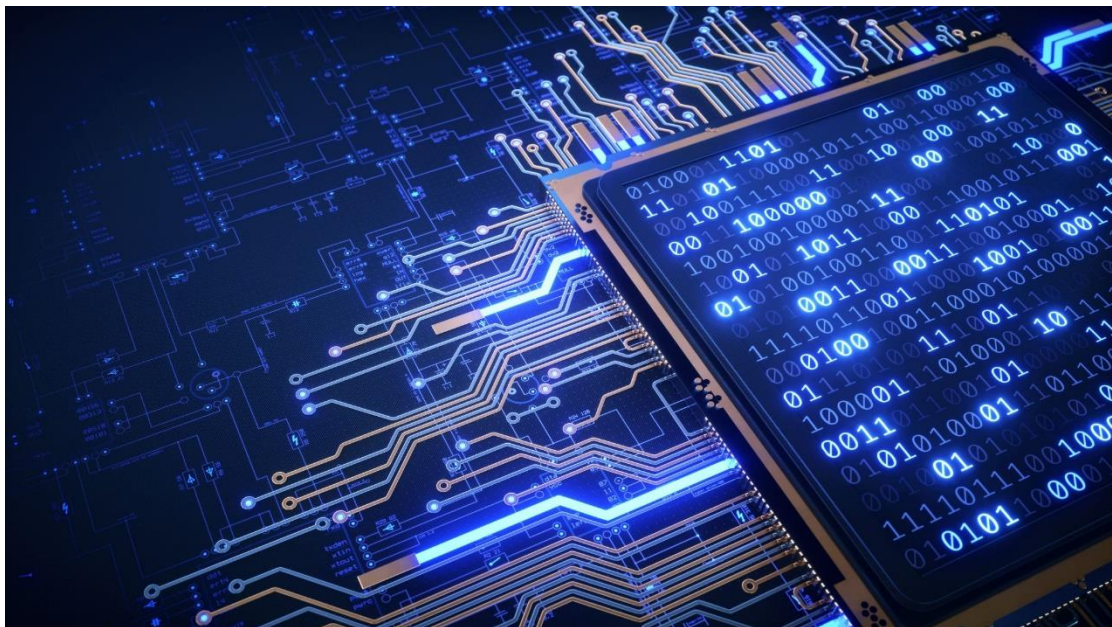# ADVANCED UNIX PROGRAMMING ASSIGNMENT REPORT

ASSIGNMENT 2

TEAM 9 — 林禾堃、馬毓昇、陳曦

Sep 2023

1. Code implementation

1.1 Check the arguments

```c
    // check arguments
    // which should include "filename", "source_file" and "dest_file"
    if(argc == 1){
        fprintf(stderr, "cp: missing file operand\n");
        return 1;
    }
    else if(argc == 2){
        fprintf(stderr, "cp: missing file operand after '%s'\n",
argv[1]);
        return 1;
    }
```

First, we check whether the argument format is valid, and print out the error if there's one.

1.2 Open the files

```c
    char *source_file = argv[1];
    char *dest_file = argv[2];

    // open source_file and check if source_file exists
    int source_fd = open(source_file, O_RDONLY);
    if(source_fd == -1){
        fprintf(stderr, "cp: cannot stat '%s': No such file or
directory\n", source_file);
        close(source_fd);
        return 2;
    }
    // open dest_file and check if there's an error
    int dest_fd = open(dest_file, O_WRONLY | O_CREAT | O_TRUNC, 0666);
    if(dest_fd == -1){
        perror("Error occurred: ");
        return errno;
    }
```

Second, we open the files by open and print out the error if there's one.

1.3 Read from source and write to destination

```c
// read source and write destination
char buffer[4096];
int read_sz;

// read and write from the beginning of the files
lseek(source_fd, 0, SEEK_SET);
lseek(dest_fd, 0, SEEK_SET);

while((read_sz = read(source_fd, buffer, sizeof(buffer))) > 0){
    // write the content
    if (write(dest_fd, buffer, read_sz) == -1)
        fprintf(stderr, "write error");
}
```

Afterward, we read and write the files by lseek-ing both files to the beginning, and run a while loop, which read from the source file chunk by chunk (size of buffer) until the file ends, then overwrite the chunk to the destination file.

1.4 Close the files

```c
// close the files
close(source_fd);
close(dest_fd);
```

Finally, close the files and the function terminates.