# CSE 344 System Programming
# Homework #5 Report

Firstly I created a file structure FileInformation. In this structure sourceFd hides sourceFiles file descriptor, destFd hides destination files file descriptor,filename hides name of coppied file,type hides type of file.This is for consumerThreadFunc.In this function I first look for type of my file.My copy operation is different for FIFO and Regular Files so I need to pass the type of coppied file,destinationPath is for hide files destination path.I use it for create FIFO to destination file with mkfifo,and fileSize is for hiding coppied regular files sizes.I open descriptors here and I get information about files (like type,filename,path) here so I need to pass them in producerThreadFunc so I must hide them in here for usage in consumerThreadFunc. The global variables bufferMutex, bufferEmpty, bufferFull, and outputLock are used for synchronization and mutual exclusion in program.Since it is a multithreaded program I need them for prevent race conditions and data corruption.
`bufferMutex`: that is used to protect access to shared resources or critical sections of code. It ensures that only one thread can enter the critical section at a time for preventing data races.
`bufferEmpty` and `bufferFull`: These are used with the `bufferMutex` to implement a producer-consumer pattern. They allow threads to wait until a certain condition is met before proceeding. `bufferEmpty` is used by consumer threads to wait until there is data in the buffer to consume, and `bufferFull` is used by the producer thread to wait until there is space available in the buffer to produce more data.
`outputLock`: This mutex is used to synchronize access to the standart output, such as printing messages. It ensures that only one thread can access the standart output at a time.

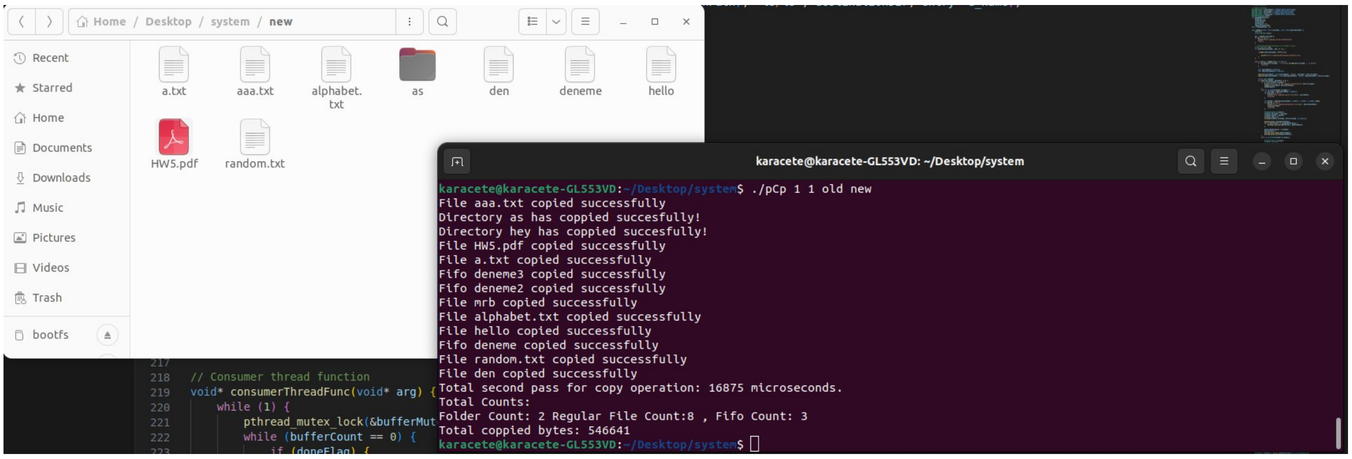
I create copyDirs function for creating directories to destination directory. I need to recursively copy subdirectories. So for this, I created this function.When producerThreadFunc finds a folder for copy,It calls this function.This function is recursive. Every time it recursively founds subdirectories and files.Looks directory for if it is a parent directory or current directory.If one of them it skip it.Creates destination directory if it doesn't exist. This is done by checking the existence of the destination directory using stat. If the directory doesn't exist, mkdir is called to create it with permissions 0777.And fills information about files. If file is directory, calls himself again.If it is regular file,makes it's type 0. Opens sourceFd and destFd for usage in consumerThreadFunc.Save file info iin buffer.Then wait for consumerThread to copy file.If it is a FIFO makes it's type 1.Hides infos like path type in buffer.And wait for consumer thread to copy fifo.

My producerThreadFunc is so similar to copyDirs function.It again controls files for 3 type.Directory,FIFO,Regular File then make sames operation as copyDirs function.

My consumerThreadFunc is reads an item from the buffer,If file type is regular file (type==1) copies the file from the source file descriptor to the destination file descriptor and closes files,If the type is FIFO , it creates a new FIFO with same name and it's same directory.In critical section It writes a message to standard output,increases regular files count or FIFO count and add totalBytesSize to fileSize.

In main I allocate space for global pointer variables. Control if the program called properly. Program should call like ./pCp bufferSize numberOfConsumers oldDirectory newDirectory. I hide this informations in variables and then create threads.I get time of day before creating threads.There must be one producerThread and there should be at most bufferSize consumer thread.Then pthread join, I wait for all thread finish ther jobs. Then get time of day again and substrack them to each other to find time passed for program working time.Then I print required informations.

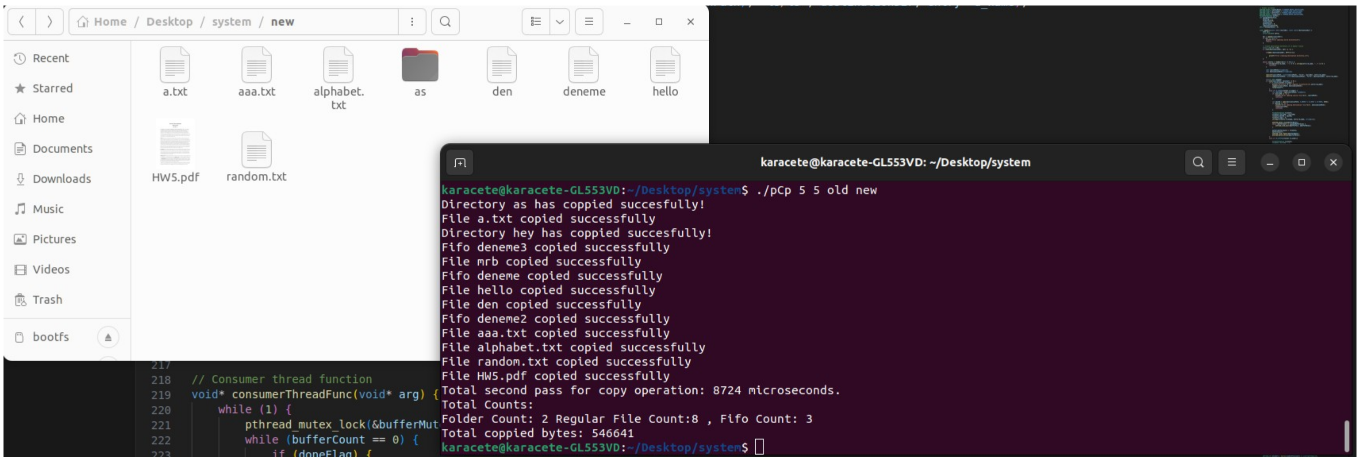
Different runs and results:


Home / Desktop / system / new
a.txt
aaa.txt
alphabet.txt
as
den
deneme
hello
HW5.pdf
random.txt
karacete@karacete-GL553VD: ~/Desktop/system

```
// Consumer thread function
void* consumerThreadFunc(void* arg) {
    while (1) {
        pthread_mutex_lock(&bufferMut
        while (bufferCount == 0) {
            if (doneFlag) {
```

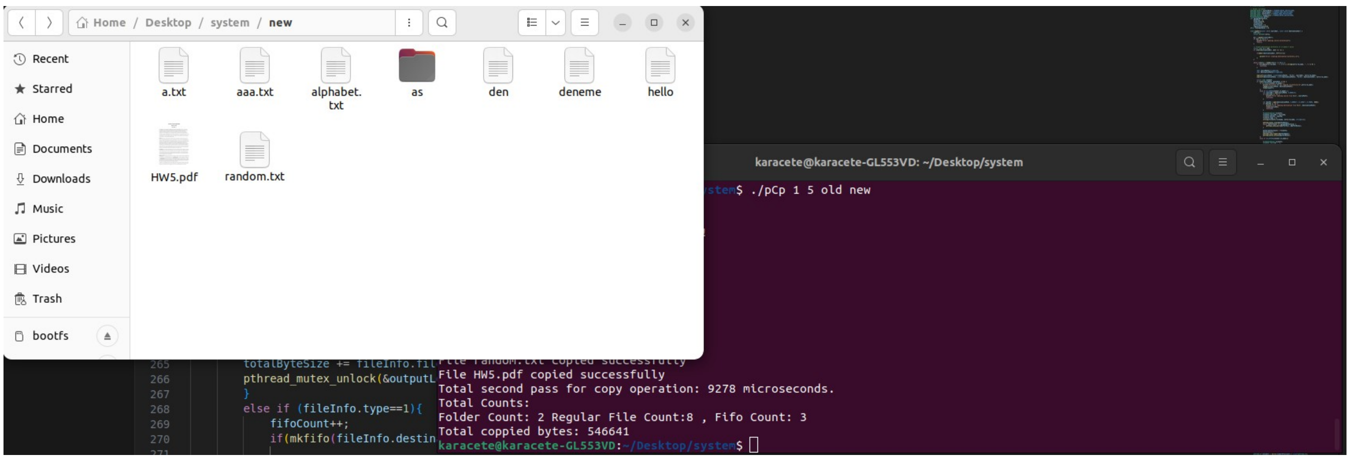
```
karacete@karacete-GL553VD:~/Desktop/system$ ./pCp 1 1 old new
File aaa.txt copied successfully
Directory as has copied succesfully!
Directory hey has copied succesfully!
File HW5.pdf copied successfully
File a.txt copied successfully
Fifo deneme3 copied successfully
Fifo deneme2 copied successfully
File mrb copied successfully
File alphabet.txt copied successfully
File hello copied successfully
Fifo deneme copied successfully
File random.txt copied successfully
File den copied successfully
Total second pass for copy operation: 16875 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:8 , Fifo Count: 3
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$
```

With 1 thread it took very long time then others.


Home / Desktop / system / new
a.txt
aaa.txt
alphabet.txt
as
den
deneme
hello
HW5.pdf
random.txt
karacete@karacete-GL553VD: ~/Desktop/system

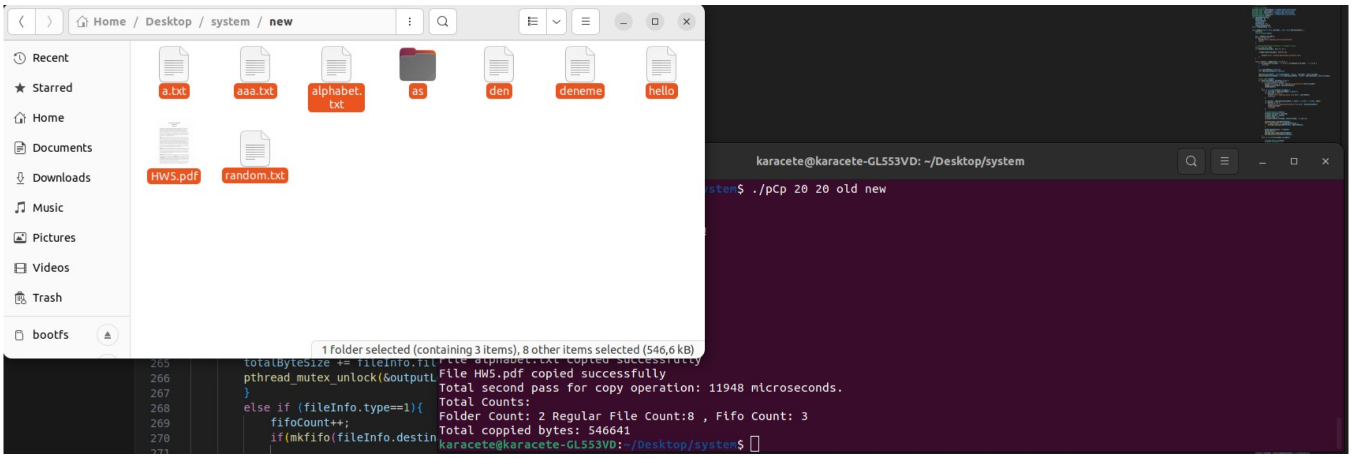
```
// Consumer thread function
void* consumerThreadFunc(void* arg) {
    while (1) {
        pthread_mutex_lock(&bufferMut
        while (bufferCount == 0) {
            if (doneFlag) {
```

```
karacete@karacete-GL553VD:~/Desktop/system$ ./pCp 5 5 old new
Directory as has copied succesfully!
File a.txt copied successfully
Directory hey has copied succesfully!
Fifo deneme3 copied successfully
File mrb copied successfully
Fifo deneme copied successfully
File hello copied successfully
File den copied successfully
Fifo deneme2 copied successfully
File aaa.txt copied successfully
File alphabet.txt copied successfully
File random.txt copied successfully
File HW5.pdf copied successfully
Total second pass for copy operation: 8724 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:8 , Fifo Count: 3
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$
```


Home / Desktop / system / new
a.txt
aaa.txt
alphabet.txt
as
den
deneme
hello
HW5.pdf
random.txt
karacete@karacete-GL553VD: ~/Desktop/system

```
            totalByteSize += fileInfo.fil
            pthread_mutex_unlock(&outputL
            }
            else if (fileInfo.type==1){
                fifoCount++;
                if(mkfifo(fileInfo.destin
```

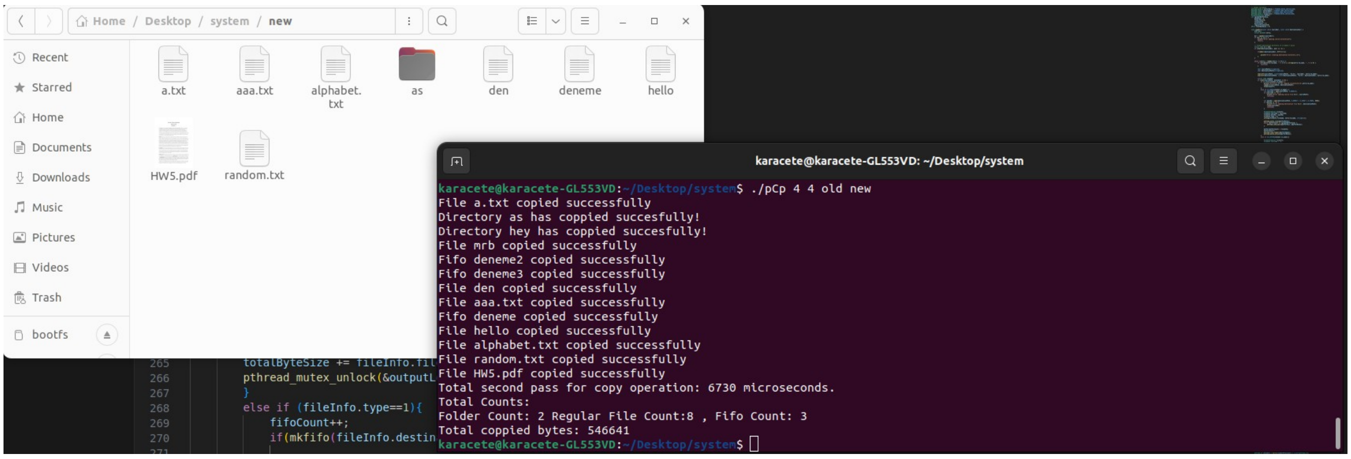
```
stem$ ./pCp 1 5 old new
File random.txt copied successfully
File HW5.pdf copied successfully
Total second pass for copy operation: 9278 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:8 , Fifo Count: 3
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$
```


Home / Desktop / system / new
Recent
Starred
Home
Documents
Downloads
Music
Pictures
Videos
Trash
bootfs
a.txt
aaa.txt
alphabet.txt
as
den
deneme
hello
HW5.pdf
random.txt
1 folder selected (containing 3 items), 8 other items selected (546,6 kB)
karacete@karacete-GL553VD: ~/Desktop/system

```
totalByteSize += fileInfo.fil
pthread_mutex_unlock(&outputL
}
else if (fileInfo.type==1){
    fifoCount++;
    if(mkfifo(fileInfo.destin
```

```
stem$ ./pCp 20 20 old new
File alphabet.txt copied successfully
File HW5.pdf copied successfully
Total second pass for copy operation: 11948 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:8 , Fifo Count: 3
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$ 
```

When thread size increased, programs started to became slow.

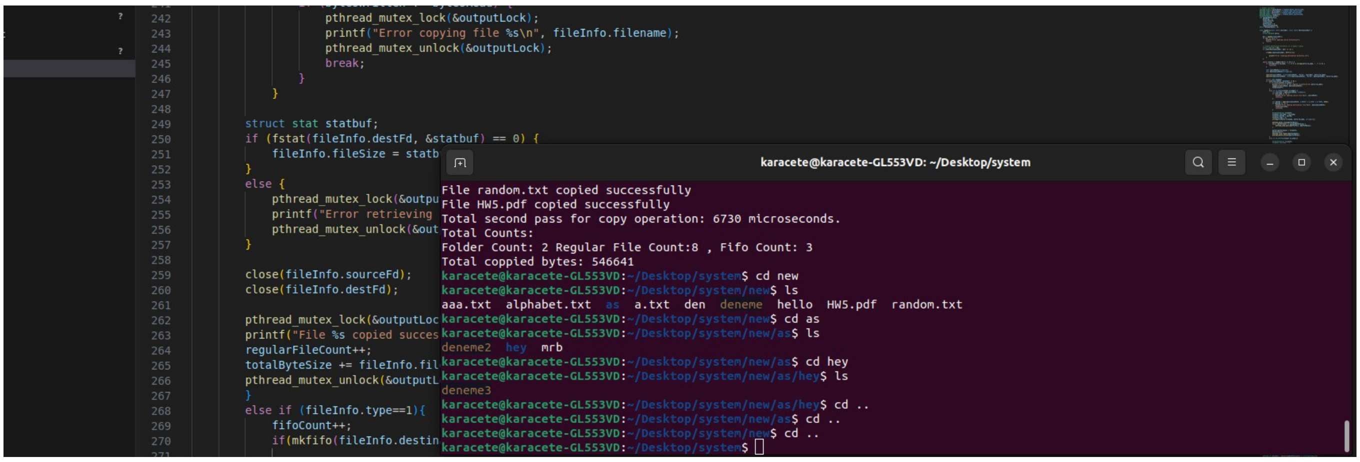


```
totalByteSize += fileInfo.fil
pthread_mutex_unlock(&outputL
}
else if (fileInfo.type==1){
    fifoCount++;
    if(mkfifo(fileInfo.destin
```

```
karacete@karacete-GL553VD:~/Desktop/system$ ./pCp 4 4 old new
File a.txt copied successfully
Directory as has copied succesfully!
Directory hey has coppied succesfully!
File mrb copied successfully
Fifo deneme2 copied successfully
Fifo deneme3 copied successfully
File den copied successfully
File aaa.txt copied successfully
Fifo deneme copied successfully
File hello copied successfully
File alphabet.txt copied successfully
File random.txt copied successfully
File HW5.pdf copied successfully
Total second pass for copy operation: 6730 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:8 , Fifo Count: 3
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$ 
```

I get best result in 4 thread and 4 buffer size.

Look from terminal:

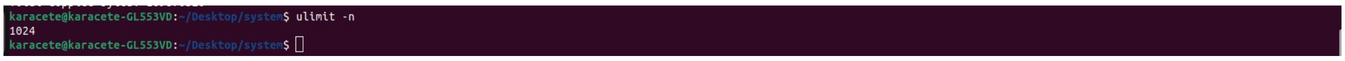


```
            pthread_mutex_lock(&outputLock);
            printf("Error copying file %s\n", fileInfo.filename);
            pthread_mutex_unlock(&outputLock);
            break;
        }
    }

    struct stat statbuf;
    if (fstat(fileInfo.destFd, &statbuf) == 0) {
        fileInfo.fileSize = statb
    }
    else {
        pthread_mutex_lock(&outpu
        printf("Error retrieving
        pthread_mutex_unlock(&out
    }

    close(fileInfo.sourceFd);
    close(fileInfo.destFd);

    pthread_mutex_lock(&outputLoc
    printf("File %s copied succes
    regularFileCount++;
    totalByteSize += fileInfo.fil
    pthread_mutex_unlock(&outputL
    }
    else if (fileInfo.type==1){
        fifoCount++;
        if(mkfifo(fileInfo.destin
```

```
File random.txt copied successfully
File HW5.pdf copied successfully
Total second pass for copy operation: 6730 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:8 , Fifo Count: 3
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$ cd new
karacete@karacete-GL553VD:~/Desktop/system/new$ ls
aaa.txt  alphabet.txt  as  a.txt  den  deneme  hello  HW5.pdf  random.txt
karacete@karacete-GL553VD:~/Desktop/system/new$ cd as
karacete@karacete-GL553VD:~/Desktop/system/new/as$ ls
deneme2  hey  mrb
karacete@karacete-GL553VD:~/Desktop/system/new/as$ cd hey
karacete@karacete-GL553VD:~/Desktop/system/new/as/hey$ ls
deneme3
karacete@karacete-GL553VD:~/Desktop/system/new/as/hey$ cd ..
karacete@karacete-GL553VD:~/Desktop/system/new/as$ cd ..
karacete@karacete-GL553VD:~/Desktop/system/new$ cd ..
karacete@karacete-GL553VD:~/Desktop/system$ 
```

The limit on the number of open file descriptors is determined by the operating system.

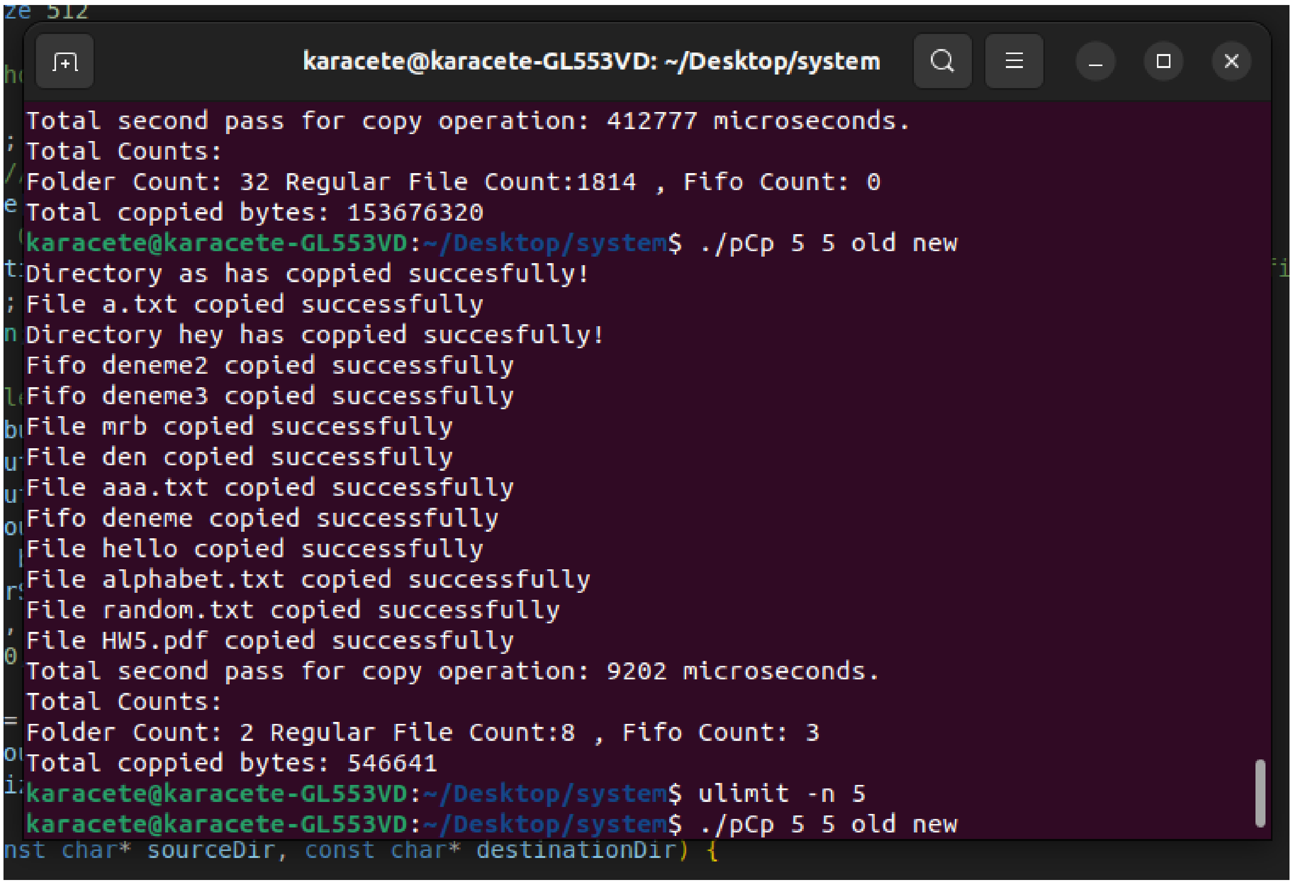
1. **Per-process limit**: This is the maximum number of file descriptors that a single process can have open at the same time. I check my limit with



```
karacete@karacete-GL553VD:~/Desktop/system$ ulimit -n
1024
karacete@karacete-GL553VD:~/Desktop/system$ 
```

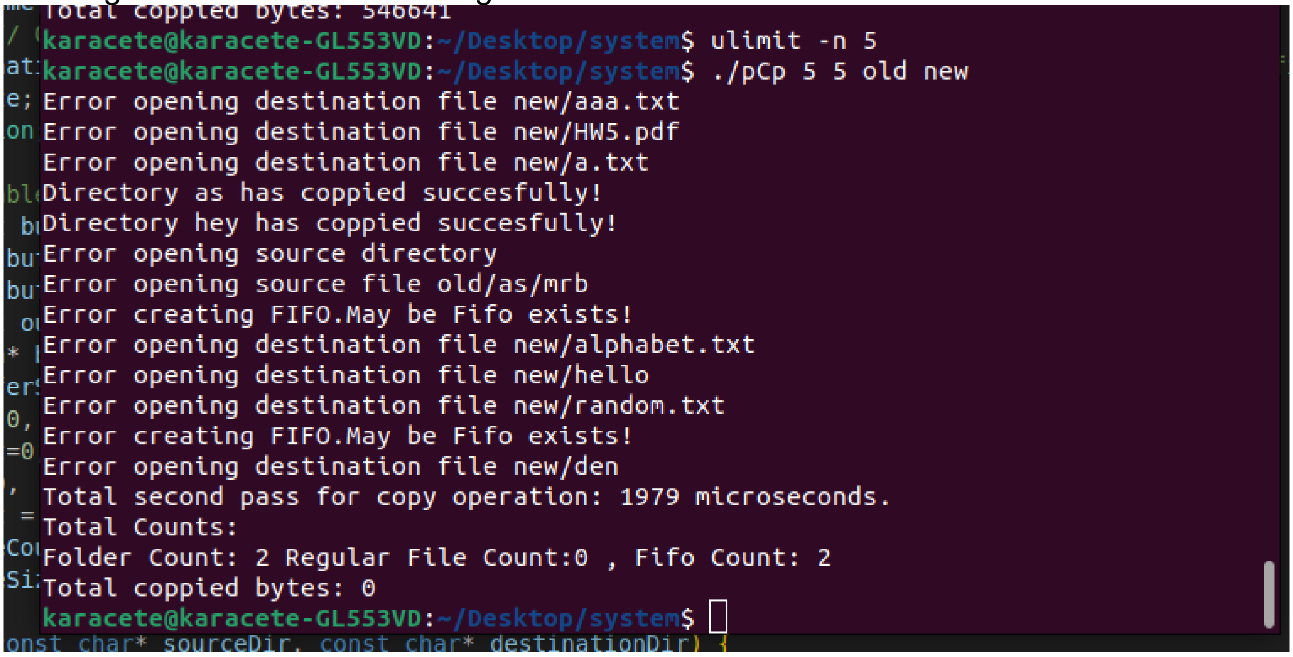
Exceeding the per-process limit on open file descriptors can lead to errors when attempting to open additional files.

ulimit -n it gives me 1024.
I firstly use it and result:



```
Total second pass for copy operation: 412777 microseconds.
Total Counts:
Folder Count: 32 Regular File Count:1814 , Fifo Count: 0
Total coppied bytes: 153676320
karacete@karacete-GL553VD:~/Desktop/system$ ./pCp 5 5 old new
Directory as has coppied succesfully!
File a.txt copied successfully
Directory hey has coppied succesfully!
Fifo deneme2 copied successfully
Fifo deneme3 copied successfully
File mrb copied successfully
File den copied successfully
File aaa.txt copied successfully
Fifo deneme copied successfully
File hello copied successfully
File alphabet.txt copied successfully
File random.txt copied successfully
File HW5.pdf copied successfully
Total second pass for copy operation: 9202 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:8 , Fifo Count: 3
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$ ulimit -n 5
karacete@karacete-GL553VD:~/Desktop/system$ ./pCp 5 5 old new
```

I change it to 5 with ulimit -n 5 .I get error.



```
Total coppied bytes: 546641
karacete@karacete-GL553VD:~/Desktop/system$ ulimit -n 5
karacete@karacete-GL553VD:~/Desktop/system$ ./pCp 5 5 old new
Error opening destination file new/aaa.txt
Error opening destination file new/HW5.pdf
Error opening destination file new/a.txt
Directory as has coppied succesfully!
Directory hey has coppied succesfully!
Error opening source directory
Error opening source file old/as/mrb
Error creating FIFO.May be Fifo exists!
Error opening destination file new/alphabet.txt
Error opening destination file new/hello
Error opening destination file new/random.txt
Error creating FIFO.May be Fifo exists!
Error opening destination file new/den
Total second pass for copy operation: 1979 microseconds.
Total Counts:
Folder Count: 2 Regular File Count:0 , Fifo Count: 2
Total coppied bytes: 0
karacete@karacete-GL553VD:~/Desktop/system$
onst char* sourceDir, const char* destinationDir) {
```

Limit is so high so there is no problem.