

# Assignment 1 CS 4328 Lab 1

---

## Requirements

- 1). The first program "pre.c" should read in a list of student names and their GPAs. You can make up a list. To be simple, you can make student name one word in lower cases. We assume there are at most 10 students in the list. The input ends when an EOF is encountered. The output of the program should display the students whose GPA is above 3.0.
- 2). The second program "sort.c" reads in a list of names from stdin and outputs them in an alphabetical order. Assume there are no more than 10 names and the sequence is read until an EOF is encountered.
3. Write a c program to set up a child-TO-parent pipe; the child should 'exec' to perform a "pre" process and its output should be connected to the pipe connected to the parent, which should 'exec' to perform a "sort" process.
4. Write a program to take a UNIX command from the command line and fork() a child to execute it.

## How to run the files!

- 1.) First unzip the folder and navigate into the Assignment - 1 folder.
- 2.) Once you are inside the Assignment -1 folder you can begin compiling the assignment files.
- 3.) Now you will need to open a terminal in this directory. You can do this by following these instructions:
  - Windows 10 **hold shift + right click** you will see "open powershell window here" click that.
  - Mac **right click the Assignment 1 folder and navigate to Services. Here you should see "New Terminal at Folder"**
  - Linux Distros **simply right click the Assignment -1 folder and click "Open in Terminal"**
- 4.) Once the terminal is open you can start compiling the files. To do so following the instructions below.

To compile the files you can simply copy and paste the commands below:

- Assignment pre.c

```
gcc pre.c
```

To run the executable file that pre.c created input ./a.out in the terminal:

```
./a.out
```

This will execute pre.c and allow you to names and their associated gpa. The input should be formatted to look like the following, CTRL-D notifies pre.c when to stop taking in names:

```
Zebra 3.0  
Monkey 1.1
```

```
Timbo 3.3  
Alex 1.1  
CTRL-D
```

The output should produce the names associated with gpa's higher than 3.0. Output from the example:

```
Timbo  
Zebra
```

- Assignment sort.c

Just like before copy and paste the following command to execute the sort.c file:

```
gcc sort.c
```

To run the executable file that sort.c created input ./a.out in the terminal:

```
./a.out
```

This will execute the sort.c and allow you to enter no more than 10 names to be sorted. Input should follow this format:

```
Zebra  
Monkey  
.  
.  
.  
Timbo
```

Sort.c will then sort the names alphabetically and produce the following output:

```
Timbo  
.  
.  
.  
Monkey  
Zebra
```

- Assignment pipe.c To compile pipe.c use the following commands:

```
gcc pre.c -o pre
gcc sort.c -o sort
```

Followed by

```
gcc pipe.c
```

To run the executable file that pipe.c created input ./a.out in the terminal:

```
./a.out
```

Just like before pre.c will prompt you to enter names and associated gpas

The input should follow the same structure as before:

```
Zebra 3.0
Monkey 1.1
Timbo 3.3
Alex 1.1
CTRL-D
```

Now sort.c will sort the output from pre.c. The output should follow as before:

```
Timbo
Zebra
```

- Assignment prog3.c To compile prog3.c copy and paste the following command in the terminal:

```
gcc prog3.c
```

To run the executable file that prog3 created copy and paste the following command in the terminal:

```
./a.out ls
** or **
./a.out ls -a
** or **
./a.out ls -t
** or **
./a.out <any unix command>
```

Hope everything runs correctly! 👍