Assignment #6: UNIX I/O

- The purpose of this assignment is to practice making UNIX I/O system calls in C.
- In a multiplexed manner, your main process will
  - read from multiple files, and
  - read from the standard input (the terminal).
- Create five pipes and spawn five child processes.
  - Connect a pipe to each child process.
  - Each child process should write to its pipe.
  - The parent process should read from all the pipes.

Assignment #6, *cont'd*

- Each of the first four child processes should generate time stamped messages to the nearest 1000th of a second (time starts at 0):

```
0:00.123: Child 1 message 1
0:02.456: Child 1 message 2
```

etc.

and write the messages one at a time to its pipe.

  - Sleep for a random time of 0, 1, or 2 seconds between messages.
  - Terminate the process after 30 seconds.

Assignment #6, *cont'd*

- Meanwhile, the fifth child process should repeatedly prompt at the terminal (standard out) and read one line of input (standard in) typed by the user.
  - Write the message (with time stamp) to its pipe.
  - Immediately prompt for the next message.
  - Terminate the process after 30 seconds.

Assignment #6, *cont'd*

- After spawning the child processes, the parent process should repeatedly read lines from the pipes.
- Use the **`select()`** system call to determine whether any of the pipes has any input.

Read the "man" pages about **`select()`** .

Assignment #6, *cont'd*

- Write the lines to an output file **`output.txt`** in the order that they were read.
- Prepend each line with a time stamp to the nearest 1000th of a second.
  - Therefore, each line with have two time stamps.
  - The first time stamp from the parent process.
  - Followed by the child process's time stamp.

Assignment #6, *cont'd*

- File **`output.txt`** will contain a mixture of lines from the child processes.
- Terminate the main process after all the child processes have terminated.

Assignment #6, *cont'd*

- Email a zip file to ron.mak@sjsu.edu
  - Your C source files
  - Your **`output.txt`** file
  - A brief report describing any issues you encountered.
- Name the zip file after your team.
  - Example: **`SuperCoders.zip`**
- Subject line:
  `CS 149-`*section* **`number Assignment #6`** *team name*
- **Due:** Monday, April 27 at 11:59 PM