

P2 Pre-Fork Server

This exercise develops a pre-forking model for a web server which maintains a process pool to handle client requests. Server binds to a port and creates child processes. Each child process of the server acts as a server and accepts connection request and sleeps for one second then sends a dummy reply. Server maintains a minimum number of idle child processes `minSpareServers` and checks that at any point its not more than `maxSpareServers`. Each child handles a fixed number of accept requests before it is killed and a new child child is created.

The communication between the child processes and server happens using UNIX domain sockets. Whenever the number of idle child processes is less than `minSpareServers` server adds 1 process to the process pool and waits for one second and then if its still less it will continue spawning child procesees exponentially to the server pool till its 32 processes per second and the spawning rate becomes constant after that.

User can press `Ctrl + C` to print number of children currently active, and for each child how many clients it has handled.

How to Run:

The following command is used to run on the machine. It compiles and runs the executable.

```
make run <min_spare_server> <max_spare_server> <max_requests_per_child>
```

To exit the process you can press `Ctrl + Z`, and kill the process using `kill` command.