Laboratory 1

Due Date : April 20, 1999 **Points** : 100

1. (*20 points*) Write a program to determine the maximum number of *simultaneous* processes that MINIX will support for a single user. Once this number is determined, any processes created to compute this number are to be terminated. Don't forget that the user's shell is itself a process!
   Hint: Spawn processes until the *fork* fails. To terminate these processes, keep track of their PIDs as they are created, and use signals.
2. (*20 points*) The FreeBSD command *stat* prints information about a file in the following format:

```
File: "/"
  Size: 1024    Allocated Blocks: 2     Filetype: Directory
  Mode: (0755/drwxr-xr-x)       Uid: (0/root) Gid: (0/wheel)
Device: 132096  Inode: 2        Links: 28
Access: Thu Apr  1 22:56:24 1999
Modify: Thu Apr  1 12:08:59 1999
Change: Thu Apr  1 12:08:59 1999
```

   Please implement this command for MINIX.
3. (*20 points*) Write a program to print the numeric and symbolic names of any signals it receives. For example, if I send the process a signal 1, it should print:

```
Thu Apr 1 23:34:19 1999 Received signal 1 (SIGHUP)
```

   The program should not attempt to catch signal 9 (SIGKILL).
4. (*40 points*) Write a program *script* that spawns a shell, and records all input to the shell and output from the shell into a file named "typescript". If your program is given an argument, use that to name the file rather than "typescript"..