Lesson 1

Introduction:

Process = Program + Resources (= program in execution)

Foreground: Although the computer supports multi processes, you can only run one process.
Background: In order to run several processes we need to run them in the background. Running programs in the background is done using ***&*** at the end of the program name. You can basically run numerous programs in the background (limited by system resources).

Process in foreground can print output but cannot receive input. In order to receive input the process is ***stopped*** - the process no longer receives CPU time. A process can stop getting CPU time, do I/O operation and then continue getting CPU from the state it stopped - ***context switching*** (more in the lecture). Example:

```
% cat input.c
main()
{
        char c;
        scanf("%c", &c);
}
% gcc -o need_input input.c
% need_input &
[1] 731
[1]     + Suspended (tty input)     need_input
```

Kill:
One can cause a process to stop:
Every process has a unique PID (process id) number. In order to *stop* a process:
% kill -STOP 1005 (sending process #1005 a STOP signal → causing the process to stop)

% kill -KILL 1005 (killing process #1005 which runs in the background).
% Ctrl-C. (killing process which runs in the foreground).

Job = a collection of processes. For example:
% who | sort | more.
Each process has its own unique pid, but the collection has one jid. This number controls all the processes (for example, move all to background/foreground).

Note: The pid is unique across the system. The jid is unique per terminal.

↓ next/current →	stop	foreground	background
stop		ctrl-Z	kill -STOP pid kill -STOP %jid
foreground	fg %jid		fg %jid
background	kill -CONT pid kill -CONT %jid bg %jid		
terminated	kill -KILL pid kill -KILL %jid	ctrl-C	kill -KILL pid kill -KILL %jid

% jobs (show the jobs which are currently alive).
% ps (show status of active processes).

Example:

```
1 > cat ensof.c
main()
{
        while (1);
}
2 > gcc -o ensof ensof.c
3 > ensof
^Z
[1]     + Suspended     ensof
4 > jobs
[1]     + Suspended     ensof
5 > jobs -l
[1]     + 11068 Suspended               ensof
6 > ps
PID     TT      S       TIME    COMMAND
6822    pts/24  S       0:00    -tcsh
11068   pts/24  T       0:01    ensof
7 > fg %1
ensof
^Z
[1]     + Suspended     ensof
8 > bg %1
[1]     ensof &
9 > jobs
[1]     Running         ensof
10 > kill -STOP 11068
[1]     + Suspended (signal)    ensof
11 > jobs
[1]     + Suspended (signal)    ensof
12 > bg %1
[1]     ensof &
13 > fg %1
ensof
^C
14 > jobs
15 > ensof &
[1] 11173
16 > kill -KILL %1
[1]     Killed  ensof
17 > jobs
18 > ps
PID     TT      STAT    TIME    COMMAND
7348    pts/24  S       0:02    - tcsh
```