

## Lab 2 prep

### 1. What do the commands `ps` and `top` do?

Displays information about the active processes. Running `ps` from a terminal shows the list of processes started from the shell. Normally, it would only show `bash` and `ps`, but running a program in the background previously followed by `ps` would also show the given program.

To view all running processes (not just started from the shell), use `ps -A`.

The `top` command simply displays a table of all the processes, along with some information about the CPU and memory.

### 2. How do you run a program in the background?

To run a program in the background, we type the program name followed by a `&`. For example, if you wanted to start Firefox in the background, you would type:

```
firefox &
```

### 3. How do you terminate a task using its process ID?

In order to terminate a process by its name, we use the `killall` command. To terminate a process given its **PID**, we use `pkill`.

For example, to terminate process with PID of 14722, you would type `kill 14722`.

Notice that the above is equivalent to `kill -15 14722`, where the 15 indicates the process should be terminated, which allows the program to do some last minute cleanup before closing.

In order to force close the program, use `kill -9`.

### 4. How do you pause a running program, and then send it to the background / bring a background back to the foreground?

- If a process is already running, you can stop it with the **CTRL+Z** shortcut
- Use the `bg` command to bring the stopped program to the background
- Use the `fg` command to bring the stopped program to the foreground

## 5. What does the `taskset` command do?

The processor **affinity** allows binding processes to specific CPU cores.

The `taskset` command can be used to **retrieve the affinity** of a specific process or to launch a given command with a specified affinity.

### Retrieve affinity of a process

Where [PID] is the process id:

```
taskset -p [PID]
```

This would normally output something like this:

```
pid [PID]'s current affinity mask: f
```

The `f` is the hexadecimal value which describes that the process can run on any 4 of the CPU cores. `f` corresponds to hex `0x15` indicating it would be able to run on cores 0-15.

### Changing process affinity

To change the affinity of process with id [PID] to affinity with hex value of `0x1`, we write:

```
taskset -p 0x1 [PID]
```

output:

```
pid [PID]'s current affinity mask: f
```

```
pid [PID]'s new affinity mask: 1
```

## 6. What would passing the parameter `-c 0` to `taskset` do?

The `-c` allows to specify a range of CPUs, such as 0, 1, 0-3, etc... So `taskset -c 0 ls` would set the affinity mask to 0, only allowing the command to run on the one CPU core.

## 7. What is an oscilloscope?

An oscilloscope is an electronic test instrument that allows real time graphing of input voltages over time. This results in a 2D representation of one or more voltage signals over time. It can be used to measure waveforms, display things like capacitor discharging and more. Most oscilloscopes allow you to calibrate/adjust the time scale, voltage range and many other properties.

8. What is shown on each axis of the oscilloscope?

- Time on the X-Axis.
- Voltage on the Y-Axis.

9. How long is one cycle in the waveform above?

- 5ms (0.005sec)

10. What frequency is the square wave?

- 200Hz

11. Why are the green traces thick at the peaks of the cycle? What causes this '*roughness*' in the data?

This is called **noise**. It is caused by internal component issues such as loose connections, power supplies, or even faulty components.

12. Draw the waveform you would expect for one byte sent over this serial link