Conceptual Questions

1. Summary of linux, why it's famous and examples of it's use.

Linux is an open source kernel based on the Unix OS, written by Linus Torvalds and is licensed under the GPL 2.0. The operating system that uses this Kernal is called GNU, which was developed by Richard Stallman and the GNU Project. Most Linux users today make use of custom Linux builds called disributions or distros for short. A Linux distro will have it's own custom version of the Linux kernal, some kind of GUI manager and a host of software that will vary between each distro. Each distribution must also be licensed under the GPL.

The main reason why Linux has become so popular is because it is open source, meaning any company or organization interested in creating software using Linux can do so freely. Linux is also very stable because it does not require you to execute uncessesary processes in the background and it does not require you to have a GUI. This makes it ideal for servers that need stability. Linux is also incredibly customizable, allowing users to build their own operating system from the bottom up with all sub components being interchangeable and modifiable.

Some examples of where Linux is used:
The Android OS and Blackberry OS are built on Linux. The ISS uses Linux. The US military, federal aviation and many other companies will use Linux to run their servers and computers.

2. What is free or open source software, why is it significant?

Open source software means that users should be able to view and edit the source code for the software and should be able to contribute to it's development.

Free software means that users of that software will have the ability to run, copy, distribute, study, change and improve the software. It does not necessarily entail that the software must be free of cost, instead the 'free' means 'libre' as in having the liberty to do whatever one wishes with the software. The only restriction on free software is that it must be licensed under the GPL. This means that any one who receives a copy of that software must also follow the GPL and can only distribute it under that same license.

The two terms essentially refer to the same type of software and they follow the same ideologies. However, some freedoms such as the ability to redistribute and resell code is specific to free software. Also, open source software doesn't necessarily need to be licensed under the GPL, however free software by definition must be licensed under the GPL.

3. What are the benefits of Open Source software, what are the drawbacks?

Open source software is usually crowd sourced and allows users to test and propose changes for the software. This allows for lower cost, flexibility and increased security when compared to proprietary software. Another benefit is that bugs and security flaws in the software can be found and dealt with much faster when the source code is open to the public.

One drawback of open source software is that if a hacker were to find a bug and exploit it rather than report it to the developers or attempt to create a fix for it, user data and other sensitive information can be leaked. Another issue is that proprietary software usually makes more money than open source software because it makes it harder for competing developers to develop similar software. This is especially true for embedded software and driver software.

4. Name an open source license.

The MIT License is an example of an open source license.

5. Name the three standard streams, their numeric value, and the purpose.

STDIN 0 - This is the standard input stream for a program

STDOUT 1 - This is the standard output stream for a program

STDERR 3 - This is the standard error stream for a program, it is used for error messages and and diagnostics

Application Questions

1.

```
> mkdir tutorials
> touch tutorial1
> cp tutorial1 tutorial_1
> rm tutorial1
```

2.

```
> cd ~
> ln -s tutorial_1 /work/tutorials/tutorial_1
```

3.

```
> grep -rl "stdin"
> ~/work/tutorials/
```

4.

```
> grep -rl "tutorial*" 2>errors.log
```

5.

```bash
#!/bin/bash

if [ "$1" == "loop" ];
then
 for i in {1..5}
 do
  echo $1
  echo $i >> "count.log"
 done
fi
```

Question 5 output.

```
> ./question5.sh loop
loop
loop
loop
loop
loop

> cat processes.log:
   PID TTY          TIME CMD
     2 tty1     00:00:03 bash
   594 tty1     00:00:00 question5.sh
   595 tty1     00:00:00 ps

> cat count.log:
1
2
3
4
5
```