

Feature	Linux Distribution 1	Linux Distribution 2	Linux Distribution 3
User interface	GNOME(easy to use)	GNOME(modern)	No default its the user choice
Packagemntagem ent	APT (.deb packages)	DNF (.rpm packages)	Pacman(.pkg, zst packages)
Pre-Installed apps	Office tools	Minimal but polished apps	User installs everything
Customizability	Medium themes and extensions	Only medium extensions available	Hlgh (the user can build their system how they want)
Community support	Large community	Large community	Niche but passionate, with a medium community
Release model	Fixed releases every 6 months	Rolling (within versions)	Rolling releases
System requirements	Moderate (can run on most PC's)	Moderate (typically needs new hardware)	Light to heavy depending on the configuration
Security focus	Good	Very strong(lots of security patches)	Dependant on user configuration
Ease of installation	Very easy	easy	Advanced
Ideal for	Beginners and general users	Developers	Advanced users

## 2. (p. 864) question#7

**What does the following Linux shell pipeline do?**

```
grep rt xyz | wc -l
```

This command counts how many lines in the file xyz contain the letter "rt"

### 3. (p. 865) question#24

#### **Why are open-file-descriptor tables necessary in Linux?**

The open file descriptor table links a small file descriptor number to the actual open files information (ex. permissions). They essentially make it easy for processes to manage and access their open files effectively without needing to know all the details from the files themselves.

### 4. (p. 866) question#36

#### **Explain under what situations a process may request a shared lock or an exclusive lock. What problem may a process requesting an exclusive lock suffer from?**

For shared lock: they are requested when a process wants to read a file and multiple processes can hold a shared lock at the same time.

For exclusive lock: these are requested when a process wants to write to a file. Only one process can hold an exclusive lock; no other reads or writes to the file can happen at the same time.

The problem that arises is known as starvation. If many shared locks are constantly active, a process wanting an exclusive lock may wait forever.

### 5. (p. 867) question#45

#### **. On multi-CPU platforms, Linux maintains a *runqueue* for each CPU. Is this a good idea? Explain your answer?**

Yes, it's a good idea. It improves speed and scalability, but needs smart balancing to avoid one CPU doing all the work. Additionally this is a good approach because it allows for better parallelism and faster scheduling because each CPU can choose a process from its own list. Some potential considerations to think about is one CPU might be overloaded while the other is idle.