11. What is a user-defined function in shell scripting? Explain with an example.

A function is like a mini-script inside your script that you can reuse whenever you want. It helps keep your code neat and avoids repeating yourself. Example:

```
#!/bin/bash
greet() {
echo "Hello, $1!"
}
greet "Pranav"
```

This function prints a greeting using the name you pass.

12.Write a bash script with a function that multiplies two integer numbers.

Here's a script with a function to multiply two numbers:

```
#!/bin/bash
multiply() {
result=$(( $1 * $2 ))
echo "The result is $result"
}
multiply 5 3
```

13. How are arrays (1D, 2D, 3D) declared in bash?

•A simple one-dimensional (1D) array is just a list of values: arr=(1 2 3 4) .

•Bash doesn't directly support 2D or 3D arrays, but you can simulate them by combining indices or creating arrays inside arrays using special tricks.

14.Write a shell script to display elements of an array.
This script prints all the items in an array:

```
#!/bin/bash
arr=(10 20 30 40)
for element in "${arr[@]}"
do
echo $element
done
```

15. What is the purpose of cron in Linux?

Cron is a tool that schedules tasks to run automatically at specific times. It's useful for running backups or cleaning files regularly without needing you to do it manually.

16.Write a cron job to run a backup script every day at midnight.

You add this line to your cron schedule:

0 0 * * * /path/to/backup_script.sh

17. How do you schedule a one-time job using the `at` command?

You can plan a task for one specific time like this:

echo "/path/to/script.sh" | at 10:00 PM

18. Write a script to display disk usage using `df` and `du` .

This script shows free disk space and the size of the current folder:

```
#!/bin/bash
echo "Disk free space:"
df -h
echo "Disk usage of current directory:"
du -sh .
```

19. How can you log the output of a script using the `tee` command?

You can see the output on your screen and save it at the same time using:

./script.sh | tee output.log

20. Explain with an example how shell scripting can automate system administration tasks.

Scripts automate tasks by running commands like checking disk usage or backing up files automatically. For example, a script can check if your disk space is running out and send an alert so you fix it on time.