

1. What is a user-defined function in shell scripting? Explain with an example. (CO4)

Ans. A user-defined function in shell scripting is a block of code that performs a specific task and that can be reused multiple times in the script.

```
1 #!/bin/bash
2 greet(){
3     echo "Hello, $1!"
4 }
5
6 greet Priyanshu
```

```
priyanshu@priyanshu:~$ gedit greet.sh
priyanshu@priyanshu:~$ chmod +x greet.sh
priyanshu@priyanshu:~$ ./greet.sh
Hello, Priyanshu!
priyanshu@priyanshu:~$
```

2. Write a bash script with a function that multiply two integer numbers. (CO4)

Ans.

```
1 #!/bin/bash
2 multiply(){
3     result=$(( $1 * $2 ))
4     echo "The product is: $result"
5 }
6
7 multiply 5 7
```

```
priyanshu@priyanshu:~$ gedit multiply.sh
priyanshu@priyanshu:~$ chmod +x multiply.sh
priyanshu@priyanshu:~$ ./multiply.sh
The product is: 35
priyanshu@priyanshu:~$
```

3. Explain how arrays (1D, 2D, and 3D) are declared in bash scripting. (CO4)

Ans. 1D Array

```
1 fruits=("apple" "banana" "cherry")
```

2D Array

```
1 matrix[0,0]=1
2 matrix[0,1]=2
3 matrix[1,0]=3
4 matrix[1,1]=4
```

3D Array

```
1 cube[0,0,0]=10
2 cube[0,0,1]=20
3 cube[0,1,0]=30
```

4. Write a shell script to display elements of an array. (CO4)

Ans.

```
1 #!/bin/bash
2
3
4 fruits=("apple" "banana" "cherry" "mango")
5 echo "The elements of the array are:"
6
7 for item in "${fruits[@]}"
8 do
9     echo "$item"
10 done
```

```
priyanshu@priyanshu:~$ gedit display.sh
priyanshu@priyanshu:~$ chmod +x display.sh
priyanshu@priyanshu:~$ ./display.sh
The elements of the array are:
apple
banana
cherry
mango
priyanshu@priyanshu:~$
```

5. What is the purpose of cron in Linux? (CO4)

Ans. The purpose of cron in Linux is to automate tasks by scheduling commands or scripts to run at specific time or intervals. It helps us to run tasks automatically - like backups, updates or sending reports without doing them manually

6. Write a cron job to run a backup script every day at midnight. (CO4)

Ans. A cron job to run a backup script every day at midnight is

0 0 * * * /home/user/backup.sh

0 0 = Run at midnight

* * * = Every day, every month, every weekday

7. How do you schedule a one-time job using at command? (CO4)

Ans. To schedule a one-time job using at command we use the following step

```
priyanshu@priyanshu:~$ at 14:30
warning: commands will be executed using /bin/sh
at Wed Oct 15 14:30:00 2025
at> 
```

8. Write a script to display disk usage using df and du. (CO4)

Ans.

```
1 #!/bin/bash
2
3 echo "===Disk usage of file systems (df) ==="
4 df -h
5
6 echo ""
7 echo "===Disk usage of current directory (du)=== "
8 du -sh*
```

```
===Disk usage of file systems (df) ===
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           297M  1.6M  296M   1% /run
/dev/sda2       20G   12G   6.9G  64% /
tmpfs           1.5G    0   1.5G   0% /dev/shm
tmpfs           5.0M   8.0K  5.0M   1% /run/lock
tmpfs           297M  124K  297M   1% /run/user/1000
```

9. How can you log the output of a script using the tee command? (CO4)

Ans. To log the output of a script using the tee command, we can use the following command

```
priyanshu@priyanshu:~$ ./disk_usage.sh | tee output.log
./disk_usage.sh: line 7: echo===Disk usage of current directory (du)===: command
not found
===Disk usage of file systems (df) ===
Filesystem      Size  Used Avail Use% Mounted on
tmpfs            297M  1.6M  296M   1% /run
/dev/sda2        20G   12G   6.9G  64% /
tmpfs            1.5G   0    1.5G   0% /dev/shm
tmpfs            5.0M   8.0K  5.0M   1% /run/lock
tmpfs            297M  124K  297M   1% /run/user/1000
```

10. Explain with an example how shell scripting can automate system administration tasks.
(CO4)

Ans. Shell scripting can automate system administration tasks by running repetitive jobs like backups, disk monitoring, or user management automatically.

A backup script can copy files from **/home/user/documents** to **/home/user/backup** and log the time. By scheduling it with cron, the script runs daily at midnight without manual intervention, saving time and reducing errors.