DAYANANDA SAGAR UNIVERSITY

Subject: Linux Programming

Name: Amanpatel Biradar

USN: ENG24CY0081

LINUX PROGRAMMING ASSIGNMENT

1. Write a shell script using if…else to check if a number is even or odd.

```
#!/bin/bash
echo "Enter a number:"
read num
if [ $((num % 2)) -eq 0 ]
then
  echo "$num is Even"
else
  echo "$num is Odd"
fi
```

2. Explain the difference between if and case statements in bash.

The if statement is used for conditional branching based on logical conditions.
Example:

```
if [ $a -gt 10 ]; then echo "Greater"; fi
```

The case statement is used for multiple pattern matching.
Example:

```
case $var in
  1) echo "One";;
  2) echo "Two";;
  *) echo "Other";;
esac
```

3.Write a script to find the largest of three numbers entered by the user.

```
#!/bin/bash
echo "Enter three numbers:"
read a b c
if [ $a -ge $b ] && [ $a -ge $c ]
then
  echo "$a is largest"
elif [ $b -ge $a ] && [ $b -ge $c ]
then
  echo "$b is largest"
else
  echo "$c is largest"
fi
```

4. How do you use a for loop to traverse an array in bash? Give an example.

```
arr=(123 "Abs" -2.3 'A' 23.56 0)
for i in "${arr[@]}"
do
  echo $i
done
```

5. Write a shell script to loop through all files in the current directory and display their names.

```
#!/bin/bash
for file in *
do
  echo "$file"
done
```

6. What is the difference between while and until loops in bash?

while loop executes as long as the condition is TRUE.
until loop executes as long as the condition is FALSE.

Example:

```
while [ $a -lt 5 ]; do echo $a; ((a++)); done
until [ $a -ge 5 ]; do echo $a; ((a++)); done
```

7. Write a countdown timer script using a while loop.

```
#!/bin/bash
echo "Enter countdown time:"
read t
while [ $t -gt 0 ]
do
  echo "$t"
  sleep 1
  ((t--))
done
echo "Time's up!"
```

8. How do you use break and continue statements in loops? Give examples.

break → exits the loop completely.
continue → skips to the next iteration.

Example:

```
for i in {1..5}
do
  if [ $i -eq 3 ]; then continue; fi
  if [ $i -eq 5 ]; then break; fi
  echo $i
done
```

9. Write a script to check if a file exists or not using the if and else loop.

```
#!/bin/bash
echo "Enter file name:"
read fname
if [ -e $fname ]
```

```
then
  echo "File exists."
else
  echo "File does not exist."
fi
```

10. Write a script to calculate factorial of a number using for loop.

```
#!/bin/bash
echo "Enter a number:"
read n
fact=1
for (( i=1; i<=n; i++ ))
do
  fact=$((fact*i))
done
echo "Factorial of $n is $fact"
```