Operating System Lab

(BTCCSPCC602)

An Assignment Work Submitted for the 6th Semester
of Bachelor of Technology

in

Computer Science and Engineering (Data Science)

by

Suman Mondal

Registration Number: 100227240046

DEPARTMENT OF COMPUTER SCIENCE
KAZI NAZRUL UNIVERSITY
ASANSOL - 713340, WEST BENGAL

Contents

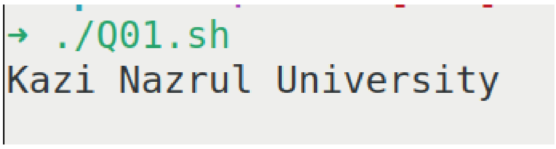
1 Shell Scripting

1.1 Write a shell script that prints ”Kazi Nazrul University” to the terminal

Source Code :

```
# Write a shell script that prints "Kazi Nazrul University" to the terminal

#!/bin/zsh
echo "Kazi Nazrul University"
```

Program Output :



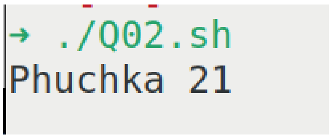
```
→ ./Q01.sh
Kazi Nazrul University
```

1.2 Assign a value to a variable in a shell script

Source Code :

```
# Assign a value to a variable in a shell script

#!/bin/zsh

name="Phuchka"
age=21

echo $name $age
```

Program Output :



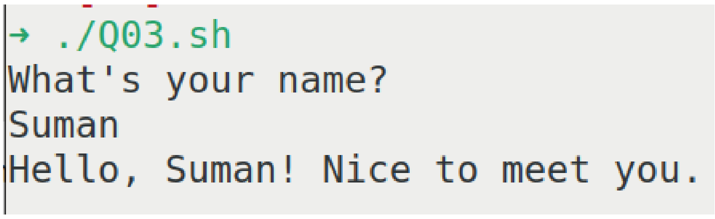
```
→ ./Q02.sh
Phuchka 21
```

1.3 Write a shell script that takes a user’s name as input and greets them

Source Code :

```
# Write a shell script that takes a user's name as input and greets them

#!/bin/zsh

echo "What's your name?"
read name
echo "Hello, $name! Nice to meet you."
```

Program Output :



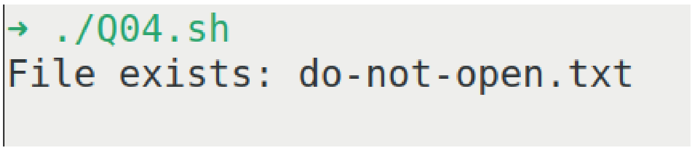
```
➜ ./Q03.sh
What's your name?
Suman
Hello, Suman! Nice to meet you.
```

1.4 Create a shell script that checks if a file exists in the current directory

Source Code :

```
# Create a shell script that checks if a file exists in the current directory

#!/bin/zsh

file="do-not-open.txt"

if [ -e "$file" ]; then
        echo "File exists: $file"
else
        echo "File not exists: $file"
fi

```

Program Output :



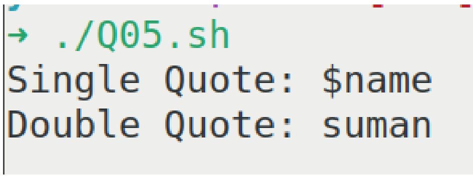
```
➜ ./Q04.sh
File exists: do-not-open.txt
```

1.5 What is the difference between single quotes (”) and double quotes (””) in shell scripting?

Source Code :

```
# What is the difference between single quotes ('') and double quotes ("") in shell scripting?

#!/bin/zsh

name="suman"

echo 'Single Quote:' '$name'
echo 'Double Quote:' "$name"
```

Program Output :



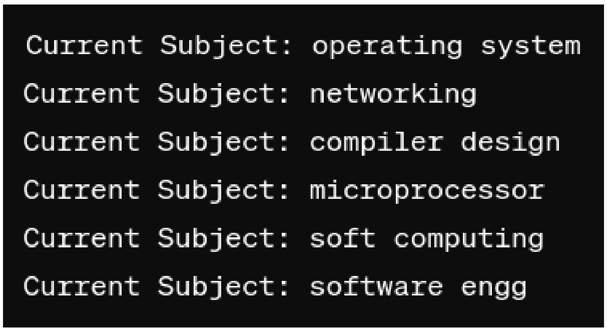
```
→ ./Q05.sh
Single Quote: $name
Double Quote: suman
```

1.6 How do you use the for loop to iterate through a list of values?

Source Code :

```
# How do you use the for loop to iterate through a list of values?

#!/bin/zsh

subjects=("operating system" "networking" "compiler design" "microprocessor" "soft computing" "software engg")

for subject in "${subjects[@]}"; do
    echo "Current Subject: $subject"
done

```

Program Output :



```
Current Subject: operating system
Current Subject: networking
Current Subject: compiler design
Current Subject: microprocessor
Current Subject: soft computing
Current Subject: software engg
```

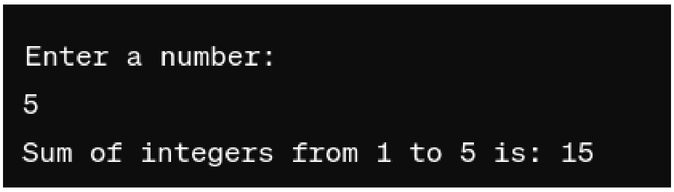
1.7 Write a shell script that calculates the sum of integers from 1 to N using a loop

Source Code :

```
# Write a shell script that calculates the sum of integers from 1 to N using a loop

#!/bin/zsh

echo "Enter a number: "
read N
sum=0

for ((i=1; i<=N; i++)); do
```

```
    sum=$((sum + i))
done

echo "Sum of integers from 1 to $N is: $sum"

```

Program Output :



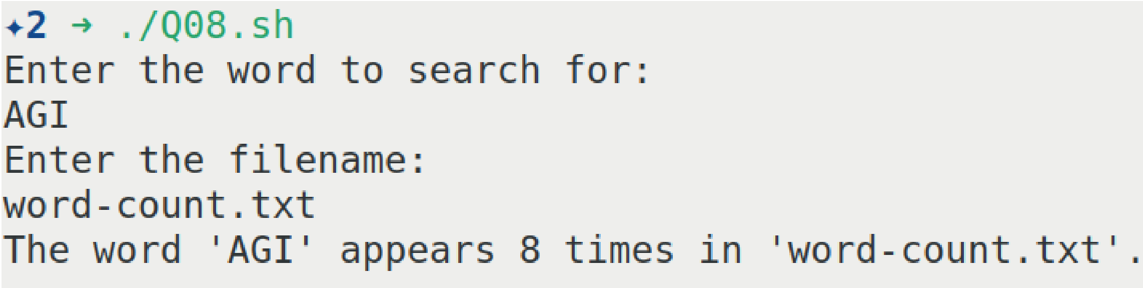
```
Enter a number:
5
Sum of integers from 1 to 5 is: 15
```

1.8 Create a script that searches for a specific word in a file and counts its occurrences

Source Code :

```
# Create a script that searches for a specific word in a file and counts its occurrences

#!/bin/zsh

echo "Enter the word to search for:"
read target_word
echo "Enter the filename:"
read filename

count=$(grep -o -w "$target_word" "$filename" | wc -l)
echo "The word '$target_word' appears $count times in '$filename'."
```

Program Output :



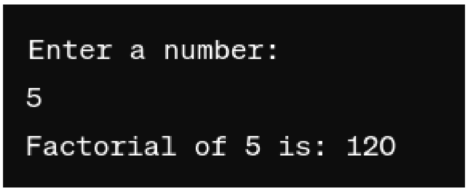
```
✦2 ➜ ./Q08.sh
Enter the word to search for:
AGI
Enter the filename:
word-count.txt
The word 'AGI' appears 8 times in 'word-count.txt'.
```

1.9 Write a function in a shell script that calculates the factorial of a given number

Source Code :

```
# Write a function in a shell script that calculates the factorial of a given number

#!/bin/zsh

calculate_factorial() {
    num=$1
    fact=1
    for ((i=1; i<=num; i++)); do
        fact=$((fact * i))
    done
    echo $fact
}

echo "Enter a number:"
read input_number
factorial_result=$(calculate_factorial $input_number)
echo "Factorial of $input_number is: $factorial_result"

```

Program Output :



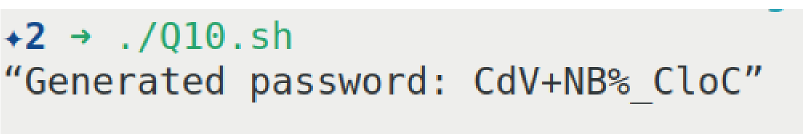
```
Enter a number:
5
Factorial of 5 is: 120
```

1.10 Write a script that generates a secure random password

Source Code :

```
# Write a script that generates a secure random password

#!/bin/zsh

generate_password() {
        tr -dc 'A-Za-z0-9!@#$%^&*()_+{}[]' < /dev/urandom | fold -w 12 | head -n 1
}

password=$(generate_password)
echo “Generated password: $password”
```

Program Output :



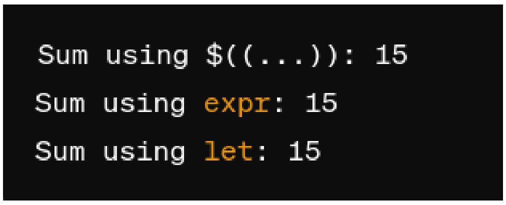
```
✦2 ➜ ./Q10.sh
“Generated password: CdV+NB%_CloC”
```

1.11 How can you use arithmetic operations within a shell script?

Source Code :

```
# How can you use arithmetic operations within a shell script?

#!/bin/zsh

num1=10
num2=5

# Using $((...)) for arithmetic
result=$((num1 + num2))
echo "Sum using \$((...)): $result"

# Using expr for arithmetic
sum=$(expr $num1 + $num2)
echo "Sum using expr: $sum"

# Using let for arithmetic
let "sum = num1 + num2"
echo "Sum using let: $sum"

```

Program Output :



```
Sum using $((...)): 15
Sum using expr: 15
Sum using let: 15
```

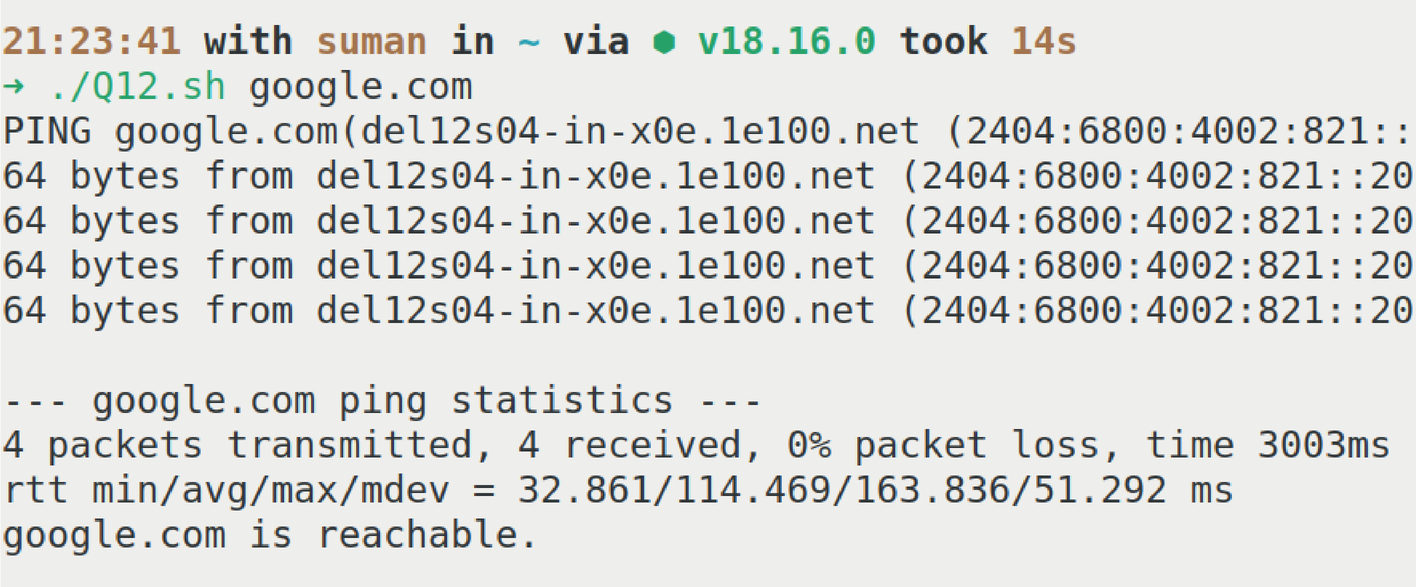
1.12 Create a script that checks if a network host is reachable

Source Code :

```
# Create a script that checks if a network host is reachable

#!/bin/zsh

host="$1"

if [ -z "$host" ]; then
    echo "Usage: $0 <hostname or IP>"
    exit 1
fi

ping -c 4 "$host"

if [ $? -eq 0 ]; then
    echo "$host is reachable."
```

```
else
    echo "$host is not reachable."
fi

```

Program Output :



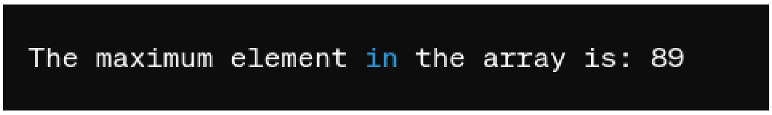
```
21:23:41 with suman in ~ via ⬢ v18.16.0 took 14s
➜ ./Q12.sh google.com
PING google.com(del12s04-in-x0e.1e100.net (2404:6800:4002:821::
64 bytes from del12s04-in-x0e.1e100.net (2404:6800:4002:821::20
64 bytes from del12s04-in-x0e.1e100.net (2404:6800:4002:821::20
64 bytes from del12s04-in-x0e.1e100.net (2404:6800:4002:821::20
64 bytes from del12s04-in-x0e.1e100.net (2404:6800:4002:821::20

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 32.861/114.469/163.836/51.292 ms
google.com is reachable.
```

1.13 Write a Shell Script to Find the Greatest Element in an Array

Source Code :

```
# Write a Shell Script to Find the Greatest Element in an Array

#!/bin/zsh

array=(3 56 24 89 67)

max=${array[0]}

for num in "${array[@]}"; do
    if (( num > max )); then
        max=$num
    fi
done

echo "The maximum element in the array is: $max"

```

Program Output :



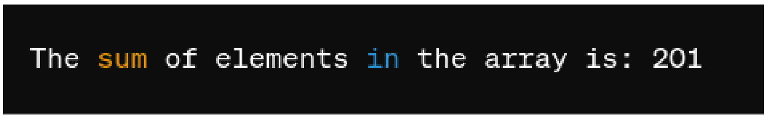
```
The maximum element in the array is: 89
```

1.14 Write a scripte to calculate the sum of Elements in an Array

Source Code :

```
# Write a scripte to calculate the sum of Elements in an Array

#!/bin/zsh

array=(1 65 22 19 94)

sum=0

for num in "${array[@]}"; do
        sum=$((sum + num))
done

echo "The sum of elements in the array is: $sum"

```

Program Output :



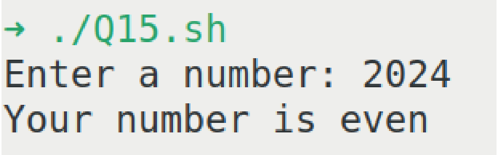
```
The sum of elements in the array is: 201
```

1.15 Check if a number is Even or Odd

Source Code :

```
# Check if a number is Even or Odd

#!/bin/zsh

read -p "Enter a number: " mynumber
if [ $((mynumber%2)) -eq 0 ]; then
echo "Your number is even"
else
echo "Your number is odd."
fi
```

Program Output :



```
→ ./Q15.sh
Enter a number: 2024
Your number is even
```