Rajshahi University of Engineering & Technology

Course No.: CSE 3202
Course Title: Sessional Based on CSE 3201

Submitted To:

Mohiuddin Ahmed
Lecturer
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology

Submitted By:

Name: Mondol Mridul Provakar
Roll: 1803062
Section: B
Department: Computer Science & Engineering

Date of submission: 21-11-2022
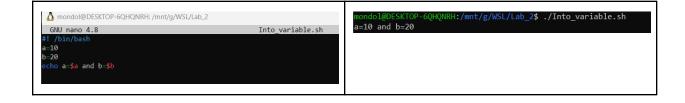
Experiment no.: 02

Experiment Name: Basic shell coding

Theory: A shell is a special user program which provides an interface to users to use operating system services. Shell accept human readable commands from the user and convert them into something which the kernel can understand. It is a command language interpreter that executes commands read from input devices such as keyboards or from files. The shell gets started when the user logs in or starts the terminal.

Problem-1: Printing hello world.

⚠ mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2 GNU nano 4.8 hello.sh #! /bin/bash echo "Hello World!"	mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./hello.sh Hello World!

Problem-2: Using variables:

There are two ways of variable declaration and they are
$CAPITAL_CASE_VARIABLE_NAME = System Variable
$lower_case_variable_name = User Variable

⚠ mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2 GNU nano 4.8 Into_variable.sh #! /bin/bash a=10 b=20 echo a=$a and b=$b	mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Into_variable.sh a=10 and b=20

Problem-3: Arithmetic operation

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                              Arithmatic.sh
#! /bin/bash
a=10
b=20
echo $((a+b))
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Arithmatic.sh
30
```

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                              Arithmatic_2.sh
#! /bin/bash
a=10
b=20
c=$((a+b))
echo $c
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Arithmatic_2.sh
30
```

Problem-4: Calculating $(a+b)^2$

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                              Arithmatic_try.sh
#! /bin/bash
a=10
b=20
echo "($a+$b)^2" |bc -l
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Arithmatic_try.sh
900
```

Problem-5: Summation of two floating numbers.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                              Arithmatic_3.sh
#! /bin/bash
a=10.11
b=10.11
c=$a+$b
echo $c|bc
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Arithmatic_3.sh
20.22
```

Problem-6: Precision of numbers.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                              Precision.sh
#! /bin/bash
echo "scale=5;11.211/3" |bc
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Precision.sh
3.73700
```

Problem-7: Calculation of powers.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
GNU nano 4.8                                    Power.sh
#! /bin/bash
echo "2^8" | bc -l
#-l is used to invoke math library
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Power.sh
256
```

Problem-8: Calculating square root.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
GNU nano 4.8                                    Square_Root.sh
#! /bin/bash
echo "scale=4;sqrt(13)" | bc -l
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Square_Root.sh
3.6055
```

Problem-9: Taking input from user

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
GNU nano 4.8                                    Input.sh
#! /bin/bash
echo "Enter a:"
read a
echo "Enter b:"
read b
echo a = $a and b= $b
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Input.sh
Enter a:
10
Enter b:
20
a = 10 and b= 20
```

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
GNU nano 4.8                                    Input_2.sh
#! /bin/bash
echo "Enter a & b:"
read a b
echo a = $a and b= $b
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Input_2.sh
Enter a & b:
10 20
a = 10 and b= 20
```

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
GNU nano 4.8                                    Input_3.sh
#! /bin/bash
read -p "Enter a:" a
read -p "Enter b:" b
echo a = $a and b= $b
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Input_3.sh
Enter a:10
Enter b:20
a = 10 and b= 20
```

Problem-10: Taking input from user as id and password

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
GNU nano 4.8                                    Input_4.sh
#! /bin/bash
read -p "Enter id:" id
read -sp "Enter password:" pass
echo id = $id and pass= $pass
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Input_4.sh
Enter id:1
Enter password:id = 1 and pass= 12345
```

Problem-11: Pass Argument during execution -1

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2

  GNU nano 4.8                                    Argument.sh
#! /bin/bash
echo $0 $1 $2 $3
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Argument.sh
./Argument.sh
```

Here,

Special variable	Description
$0	The name of the bash script.
$1, $2...$n	The bash script arguments.

Problem-12: Pass Argument during execution-2

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2

  GNU nano 4.8                                    Argument_2.sh
#! /bin/bash
args=("$@")
echo $@
echo $#
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Argument_2.sh

0
```

Here,

Special variable	Description
$#	The total number of arguments passed to the script.
$@	The value of all the arguments passed to the script.

Problem-13: Pass Argument during execution-3.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2

  GNU nano 4.8                                    Argument_3.sh
#! /bin/bash
args=("$@")
echo ${args[0]} ${args[1]} ${args[2]}
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Argument_3.sh
```

Problem-14: Checking if a number is equal to 10 or not.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                              If.sh
#! /bin/bash
a=10
if [ $a -eq 10 ]
    then
    echo $a is equal to 10
    else
    echo $a is not equal to 10
fi
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./If.sh
10 is equal to 10
```

Problem-15: Checking a number is greater than or equal to 10.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                              If_2.sh
#! /bin/bash
a=13
if [ $a -ge 10 ]
    then
    echo $a is greater than or equal to 10
fi
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./If_2.sh
13 is greater than or equal to 10
```

Problem-16: Checking valid password.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                              if_3.sh
#! /bin/bash
pass=abc123
read -sp "Enter your password:" inp
echo
    if [  $pass == $inp ]
        then
            echo welcome
        else
            echo incorrect password
    fi
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./if_3.sh
Enter your password:
incorrect password
```

Problem-17: Printing 1 to 9 by using a while loop.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                              loop.sh
#! /bin/bash
i=1
while [ $i -lt 10 ]
do
echo $i
((i++))
done
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./loop.sh
1
2
3
4
5
6
7
8
9
```

Problem-18: Printing 1 to 10 by using while loop and relational operator.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                    loop_2.sh
#! /bin/bash
i=1
while (($i <= 10 ))    #we can use relational sign inside (( ))
do
echo $i
((i++))
done
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./loop_2.sh
1
2
3
4
5
6
7
8
9
10
```

Problem-19: Printing 1 to 10 by using a for loop.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                    loop_3.sh
#! /bin/bash
for i in {1..10}
do
echo $i
done
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./loop_3.sh
1
2
3
4
5
6
7
8
9
10
```

Problem-20: Printing 1 to 10 by using for loop and relational operator.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                    loop_4.sh
#! /bin/bash
for ((i=1;i<=10;i++))
do
echo $i
done
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./loop_4.sh
1
2
3
4
5
6
7
8
9
10
```

Problem-21: Printing all elements of an array.

```
Select mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                    Array.sh
#! /bin/bash

# To declare static Array
arr=(Soykot Shuvra Barik Dibbo)

# To print all elements of array
echo ${arr[@]}
echo ${arr[*]}
echo ${arr[@]:0}
echo ${arr[*]:0}
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Array.sh
Soykot Shuvra Barik Dibbo
Soykot Shuvra Barik Dibbo
Soykot Shuvra Barik Dibbo
Soykot Shuvra Barik Dibbo
```

Problem-22: Printing elements from a particular index.

```
mondol@DESKTOP-6QHQNRH: /mnt/g/WSL/Lab_2
  GNU nano 4.8                                    Array_2.sh
# To print elements from a particular index
arr=(Soykoy shuvro barik dibbo)
echo ${arr[@]:0}
echo ${arr[@]:1}
echo ${arr[@]:2}
echo ${arr[0]:1}
```

```
mondol@DESKTOP-6QHQNRH:/mnt/g/WSL/Lab_2$ ./Array_2.sh
Soykoy shuvro barik dibbo
shuvro barik dibbo
barik dibbo
oykoy
```

Discussion: In this experiment, some basic shell codes were written. Variables, if-else, loop, array all of this were introduced in this experiment. All codes were run successfully. After doing all the above codes, the basic knowledge of coding linux commands became clear.