

Name: Rishika Sinha

Email: rishikasinha2016@gmail.com

Day6: Repetition Practice Problems with for loop

1. Write a program that takes a command-line argument n and prints a table of the powers of 2 that are less than or equal to 2^n .

Shell Script:

```
#!/bin/bash -x
echo "Enter the value:"
read n
sum=1
for((i=0;i<=n;i++))
do
sum=$((sum * 2))
echo $sum
done
```

Output:

```
$ ./w2p1.sh
+ echo 'Enter the value:'
Enter the value:
+ read n
7
+ sum=1
+ (( i=0 ))
+ (( i<=n ))
+ sum=2
+ echo 2
2
+ (( i++ ))
+ (( i<=n ))
+ sum=4
+ echo 4
4
+ (( i++ ))
+ (( i<=n ))
+ sum=8
+ echo 8
8
+ (( i++ ))
+ (( i<=n ))
```

```

+ sum=16
+ echo 16
16
+ (( i++ ))
+ (( i<=n ))
+ sum=32
+ echo 32
32
+ (( i++ ))
+ (( i<=n ))
+ sum=64
+ echo 64
64
+ (( i++ ))
+ (( i<=n ))
+ sum=128
+ echo 128
128
+ (( i++ ))
+ (( i<=n ))
+ sum=256
+ echo 256
256
+ (( i++ ))
+ (( i<=n ))

```

2. Write a program that takes a command-line argument n and prints the nth harmonic number. Harmonic Number is of the form

$$H_n = 1/1 + 1/2 + 1/3 + 1/4 + \dots + 1/n.$$

Shell Script:

```

#!/bin/bash -x
echo "Enter the number:"
read n
sum=0
for((i=1;i<=n;i++))
do
    div=$(awk 'BEGIN {print "1/"$i}')
    sum=$(awk 'BEGIN {print "$sum"+"$div}')
done
echo "Harmonic value of $n is $sum:"

```

Output:

```

$ ./w2p2.sh
+ echo 'Enter the number:'
Enter the number:
+ read n

```

```

3
+ sum=0
+ (( i=1 ))
+ (( i<=n ))
++ awk 'BEGIN {print 1/1}'
+ div=1
++ awk 'BEGIN {print 0+1}'
+ sum=1
+ (( i++ ))
+ (( i<=n ))
++ awk 'BEGIN {print 1/2}'
+ div=0.5
++ awk 'BEGIN {print 1+0.5}'
+ sum=1.5
+ (( i++ ))
+ (( i<=n ))
++ awk 'BEGIN {print 1/3}'
+ div=0.333333
++ awk 'BEGIN {print 1.5+0.333333}'
+ sum=1.83333
+ (( i++ ))
+ (( i<=n ))
+ echo 'Harmonic value of 3 is 1.83333:'
Harmonic value of 3 is 1.83333:

```

3. Write a program that takes an input and determines if the number is a prime.

Shell Script:

```

#!/bin/bash
echo "enter number"
read num
function prime
{
for((i=2; i<=num/2; i++))
do
if [  $((num\%i))$  -eq 0 ]
then
echo "$num is not a prime number."
exit
fi
done
echo "$num is a prime number."
}
r=`prime $number`
echo "$r"

```

Output:

```
$ ./w2p3.sh
enter number
31
31 is a prime number.
```

```
$ ./w2p3.sh
enter number
16
16 is not a prime number.
```

4. Extend the program to take a range of number as input and output as prime numbers in that range.

Shell Script:

```
#!/bin/bash
echo "Enter the number"
read N
echo "Enter the second number"
read M
for (( i=$N+1; i <= $M-1; i++ ))
do
p=0
for (( j=2; j <= $i-1; j++ ))
do
if [ `expr $i % $j` = 0 ]
then
p=1
break
fi
done
if [ `expr $p` = 0 ]
then
echo $i
fii
```

Output:

```
$ ./w2p4.sh
Enter the number
1
Enter the second number
20
2
3
5
7
11
13
```

17
19

5. Write a program that computes a factorial of a number taken as input.

5 Factorial – $5! = 1 * 2 * 3 * 4 * 5$

Shell Script:

```
#!/bin/bash -x
echo "Enter a number"
read num
fact=1
for((i=2;i<=num;i++))
{
    fact=$((fact * i))
}
echo $fact
```

Output:

```
$ ./w2p5.sh
+ echo 'Enter a number'
Enter a number
+ read num
5
+ fact=1
+ (( i=2 ))
+ (( i<=num ))
+ fact=2
+ (( i++ ))
+ (( i<=num ))
+ fact=6
+ (( i++ ))
+ (( i<=num ))
+ fact=24
+ (( i++ ))
+ (( i<=num ))
+ fact=120
+ (( i++ ))
+ (( i<=num ))
+ echo 120
12
```

6. Extend the program to take a range of number as input and output as prime numbers in that range.

Shell Script:

```
#!/bin/bash
echo "Enter the number"
read N
echo "Enter the second number"
read M
for (( i=$N+1; i <= $M-1; i++ ))
do
p=0
for (( j=2; j <= $i-1; j++ ))
do
if [ `expr $i % $j` = 0 ]
then
p=1
break
fi
done
if [ `expr $p` = 0 ]
then
echo $i
fi
fii
```

Output:

```
$ ./w2p6.sh
Enter the number
1
Enter the second number
20
2
3
5
7
11
13
17
19
```

Repetition Practice Problems with while loop

1. Write a program that takes a command-line argument n and prints a table of the powers of 2 that are less than or equal to 2^n till 256 is Reached..

Shell Script:

```
#!/bin/bash
read -p "Enter n=" n
echo "power of n that are less than 256 are:"
c=1
s=$((2 ** 0))
while [ $s -le 256 ]
do
echo $s
s=$((2 ** c))
((c++))
done
```

Output:

```
$ ./w2p7.sh
Enter n=7
power of n that are less than 256 are:
1
2
4
8
16
32
64
128
256
```

2. Find the Magic Number
 - a. Ask the user to think of a number n between 1 to 100
 - b. Then check with the user if the number is less then $n/2$ or greater
 - c. Repeat till the Magic Number is reached.

Shell Script:

```
#!/bin/bash
read -p "Enter the number between 1 to 100 =" number
if [ $number -gt 1 ] || [ $number -le 100 ]
then
n=0;
high=100;
low=1;
while [ $n -ne $number ]
do
div=$((high + low))
mid=$((div / 2))
```

```

if [ $number -eq $mid ]
then
n=$number
echo "magic number $number found"
elif [ $number -le $mid ]
then
echo "$number less than $mid"
high=$mid
elif [ $number -gt $mid ]
then
echo "$number is greater than $mid"
low=$mid
fi
done
else
echo "$number is not between 1-100"
fi

```

Output:

```

$ ./w2p8.sh
Enter the number between 1 to 100 =30
30 less than 50
30 is greater than 25
30 less than 37
30 less than 31
30 is greater than 28
30 is greater than 29
magic number 30 found

```

3. Extend the Flip Coin problem till either Heads or Tails wins 11 times.

Shell Script:

```

#!/bin/bash -x
heads=0
tails=0
while [ $heads<11 && $tails<11 ]
do
win=$(( RANDOM%2 ))
if [ $win -eq 0 ]
then
Heads=$((heads+1))
else
Tails=$((xtails+1))
fi
if [ $heads -eq 11 ]
then

```



```

echo "Heads Win"
elif [ $tails -eq 11 ]
then
echo "Tails win"
else
echo "Display score"
echo "Heads=$heads Tails=$tails"
fi
done

```

:

Functions Practice Problems

1. Help user find degF or degC based on their Conversion Selection. Use Case Statement and ensure that the inputs are within the Freezing Point (0 °C / 32 °F) and the Boiling Point of Water (100 °C / 212 °F)
 - a. $\text{degF} = (\text{degC} * 9/5) + 32$
 - b. $\text{degC} = (\text{degF} - 32) * 5/9$

Shell Script:

```

#!/bin/bash
echo "1. Convert Celsius to Fahrenheit:"
echo "2. Convert Fahrenheit to Celsius:"
read a
if((a==1))
then
echo "Enter value in Celsius:"
read c
if((c<=100&&c>=0))
then
t=$(awk 'BEGIN {print'$c' * '9' / '5'}')
f=$(awk 'BEGIN {print'$t' + '32'}')
echo "$c Celsius = $f Fahrenheit"
else
echo "You should enter value b/w 1 to 100"
fi
else
echo "Enter the value in Fahrenheit"
read c
if((c<=212&&c>=32))
then
f=$(awk 'BEGIN {print'$c' - '32'}')
echo "$f Fahrenheit = $c Celsius"
fi

```

Output:

1. Convert Celsius to Fahrenheit:
2. Convert Fahrenheit to Celsius:

1

Enter the value in Celsius:

98

98C=32F

2. Write a function to check if the two numbers are Palindromes

Shell Script:

```
echo "Enter the number"
read n
number=$n
reverse=0
while [ $n -gt 0 ]
do
a=`expr $n % 10 `
n=`expr $n / 10 `
reverse=`expr $reverse \* 10 + $a`
done
echo $reverse
if [ $number -eq $reverse ]
then
echo "Number is palindrome"
else
echo "Number is not palindrome"
fi
```

Output:

```
$ ./palindrome.sh
+ echo 'Enter the number'
Enter the number
+ read n
123
+ number=123
+ reverse=0
+ '[' 123 -gt 0 ']'
++ expr 123 % 10
+ a=3
++ expr 123 / 10
+ n=12
++ expr 0 '*' 10 + 3
+ reverse=3
```

```

+ '[' 12 -gt 0 ']'
++ expr 12 % 10
+ a=2
++ expr 12 / 10
+ n=1
++ expr 3 '*' 10 + 2
+ reverse=32
+ '[' 1 -gt 0 ']'
++ expr 1 % 10
+ a=1
++ expr 1 / 10
+ n=0
++ expr 32 '*' 10 + 1
+ reverse=321
+ '[' 0 -gt 0 ']'
+ echo 321
321
+ '[' 123 -eq 321 ']'
+ echo 'Number is not palindrome'
Number is not palindrome

```

3. Take a number from user and check if the number is a Prime then show that its palindrome is also prime
 - a. Write function check if number is Prime
 - b. Write function to get the Palindrome.
 - c. Check if the Palindrome number is also prime

Shell Script:

```

#!/bin/bash -x
echo "Enter number:"
read n
function prime
{
for((i=2;i<=n/2;i++))
do
use this
it is working
if [ $((n%i)) -eq 0 ]
then
echo "$n is not a prime number:"
exit
fi
done
echo "$n is a prime number"
}
r=`prime $number`
echo "$r"

```

```

number=$n
reverse=0
while [ $n -gt 0 ]
do
a=`expr $n % 10`
n=`expr $n / 10`
reverse=`expr $reverse \* 10 + $a`
done
echo $reverse
if [ $number -eq $reverse ]
then
    echo "Number is palindrome"
else
    echo "Number is not palindrome"
fi

```

Output:

```

$ ./func3.sh
+ echo 'Enter number:'
Enter number:
+ read n
11
++ prime
++ (( i=2 ))
++ (( i<=n/2 ))
++ use this
./func3.sh: line 8: use: command not found
++ it is working
./func3.sh: line 9: it: command not found
++ '[' 1 -eq 0 ']'
++ (( i++ ))
++ (( i<=n/2 ))
++ use this
./func3.sh: line 8: use: command not found
++ it is working
./func3.sh: line 9: it: command not found
++ '[' 2 -eq 0 ']'
++ (( i++ ))
++ (( i<=n/2 ))
++ use this
./func3.sh: line 8: use: command not found
++ it is working
./func3.sh: line 9: it: command not found
++ '[' 3 -eq 0 ']'
++ (( i++ ))
++ (( i<=n/2 ))
++ use this

```

```
./func3.sh: line 8: use: command not found
++ it is working
./func3.sh: line 9: it: command not found
++ '[' 1 -eq 0 ']'
++ (( i++ ))
++ (( i<=n/2 ))
++ echo '11 is a prime number'
+ r='11 is a prime number'
+ echo '11 is a prime number'
11 is a prime number
```

:

