**Q.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.**

read -p "Enter a Number " num

pow=1

for (( i=0; i<= $num; i++ ))

do

echo $i $pow

pow = 2 \* $pow

done

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

echo Enter a number

read n

i=1

sum=0

while [ $i -le $n ]

do

sum=`expr $sum + \\( 10000 / $i \\)`

i=`expr $i + 1`

done

echo Sum n series is

i=1

while [ $i -le 5 ]

do

a=`echo $sum | cut -c $i`

echo -e \"$a\\c\"

if [ $i -eq 1 ]

then

echo -e \".\\c\"

fi

i=`expr $i + 1`

done

echo

Output:-

HP@DESKTOP-EJVD731 MINGW64 ~/Desktop/practice/Assignment/Day6 (main)

$ bash Q2.sh

Enter a number

5

Sum n series is

"2"."2"8"3"3

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

echo -e "Enter Number : "

read n

for((i=2; i<=$n/2; i++))

do

ans=$(( n%i ))

if [ $ans -eq 0 ]

then

echo "$n is not a prime number."

exit 0

fi

done

echo "$n is a prime number."

**Output:**

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ touch Q3.sh

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ notepad Q3.sh

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ bash Q3.sh

Enter Number :

4

4 is not a prime number.

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ bash Q3.sh

Enter Number :

11

11 is a prime number.

**Q.4 Extend the program to take a range of number as input and output the Prime Numbers in that range.**

echo -e "Enter the range m and n:"

read m n

echo "the prime no between $m And $n Are: "

for i in $(seq $m $n)

do

flag=0

for j in $(seq 2 $(( $i-1 )))

do

ans=$(( i%j ))

if [ $ans -eq 0 ]

then

flag=1

fi

done

if [ $flag -eq 0 ]

then

echo $i

fi

done

Output :

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ notepad Q3.sh

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ bash Q3.sh

Enter the range m and n:

1 10

the prime no between 1 And 10 Are:

1

2

3

5

7

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

Answer:-

read -p "Enter a number" num

fact=1

for i in $(seq $num)

do

fact=$((fact\*i))

done

echo $fact

Output:

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ notepad Q5.sh

admin@Feb-01 MINGW64 ~/Desktop/practice/Assignment/Day6

$ bash Q5.sh

Enter a number 5

120

Q**.6 Write a program to compute Factors of a number N using prime factorization method.**

**Logic -> Traverse till i\*i <= N instead of i <= N for efficiency.**

**O/P -> Print the prime factors of number N.**

Answer:-

read -p "Enter a number " num

echo "All Prime Factors of $num are: "

for (( i=2; i\*i<=$num; i++ ))

do

if [ $num%i == 0 ]

then

prime=1

for (( j=2; j<=i/2; j++ ))

do

if [ i % j == 0 ]

then

prime=0

fi

done

if [ $prime == 1 ]

then

echo $prime

fi

fi

done

**Repetition Practice Problems with while loop**

Q.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..

echo "Think of a no. in between 1 to 100"

number=$((RANDOM%100+1))

echo $number

m=$(( number/ 2 ))

if [ $number -le $m ]

then

echo "Number is Less Than N/2"

else

echo "Number is greater Than N/2"

fi

no= $(( (number-1)%9 ))

if [[ $no -eq 0 ]]

then

echo "$number is a Magic Number"

else

echo "$number is Not Magic Number"

fi

Output :-

$ notepad Q7.sh

HP@DESKTOP-EJVD731 MINGW64 ~/Desktop/practice/Assignment/Day6 (main)

$ bash Q7.sh

Think of a no. in between 1 to 100

83

Number is greater Than N/2

Q7.sh: line 13: 1: command not found

83 is a Magic Number

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

Answer:=

flips=1

heads=0

tails=0

while [ $flips -le 11 ]

do

Result=$((RANDOM%2))

flips = $(( flips+1 ))

if [[ ${Result} -eq 0 ]]

then

echo "HEADS"

head = $(( heads+1 ))

elif [[ ${Result} -eq 1 ]]

then

echo "TAILS"

tails =$(( heads+1 ))

fi

done

echo "You Got $heads HEADS And $tails TAILS"

Q.4 Write a Program where a gambler starts with Rs 100 and places Re 1 bet

until he/she goes broke i.e. no more money to gamble or reaches the

goal of Rs 200. Keeps track of number of times won and number of bets

made.

**Q.5 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. degF = (degC \* 9/5) + 32**

**b. degC = (degF – 32) \* 5/9**

**Answer**

echo "1. Convert Celsius temperature into Fahrenheit"

echo "2. Convert Fahrenheit temperatures into Celsius"

echo -n "Select your choice (1-2) : "

read choice

case $choice in

1)echo -n "Enter Temperature (C): "

read tc

tf=$(( (tc\*9/5) + 32 ))

echo "$tc C = $tf F

;;

2)

echo -n "Enter temperature (F) : "

read tf

tc=$(( (tf-32)\*5/9 ))

echo "$tf = $tc"

;;

\*)

echo "Wrong choice entered"

;;

Esac

Output:

HP@DESKTOP-EJVD731 MINGW64 ~/Desktop/Assignments/Day6 (master)

$ bash Q11.sh

1. Convert Celsius temperature into Fahrenheit

2. Convert Fahrenheit temperatures into Celsius

Select your choice (1-2) : 1

Enter Temperature (C): 98

98 C = 208 F

HP@DESKTOP-EJVD731 MINGW64 ~/Desktop/Assignments/Day6 (master)

$ bash Q11.sh

1. Convert Celsius temperature into Fahrenheit

2. Convert Fahrenheit temperatures into Celsius

Select your choice (1-2) : 2

Enter temperature (F) : 32

32 = 0

**Q.12 Write a function to check if the two numbers are Palindromes**

echo "Enter the two number"

read n

read n1

function pal()

{

number=$n

number1=$n1

reverse=0

local s=$1

for i in $s

do

while [ $i -gt 0 ]

do

a=`expr $i % 10 `

n=`expr $i / 10 `

reverse=`expr $reverse \\* 10 + $a`

done

done

if [[ $number -eq $reverse && $number1 -eq $reverse ]]

then

echo "Number is palindrome"

else

echo "Number is not palindrome"

fi

}

r=`pal $n $n1`

echo "$r"

**Q.13 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**

Answer :=

function palindromeNo(){

number=$1

while [ $1 -gt 0 ]

do

a= $(( $number % 10 ))

rev=$( echo ${rev}${a} )

r=$(( $1 / 10 ))

done

echo $reverse

if [ $number -eq $reverse ]

then

echo "Number is palindrome"

else

echo "Number is not palindrome"

fi

}

function primeNo(){

while [ $temp -le $1/2 ]

do

ans=$(( $1%i ))

if [ $ans -eq 0 ]

then

echo "$1 is not a prime number."

else

echo "$1 is a prime number."

fi

done

palindromeNo $1

}

echo -e "Enter Number : \c"

read n

primeNo $n