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.

```
  GNU nano 6.4                                  1_powerOf2.sh
n=$1

for ((i=0; i<=n; i++)); do
  result=$((2**i))
  echo "2^$i = $result"
done
```

```
Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 1_powerOf2.sh
2^0 = 1

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 1_powerOf2.sh 1
2^0 = 1
2^1 = 2

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 1_powerOf2.sh 15
2^0 = 1
2^1 = 2
2^2 = 4
2^3 = 8
2^4 = 16
2^5 = 32
2^6 = 64
2^7 = 128
2^8 = 256
2^9 = 512
2^10 = 1024
2^11 = 2048
2^12 = 4096
2^13 = 8192
2^14 = 16384
2^15 = 32768
```

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

```
  GNU nano 6.4                          2_HarmonicNumbers.sh
n=$1
sum=0
precision=1000

for ((i=1; i<=n; i++)); do
  sum=$((sum + precision/i))
done

intN=$((sum / precision))
fraN=$((sum % precision))
echo "Harmonic Number: $intN.$fraN"
```

```
Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 2_HarmonicNumbers.sh
Harmonic Number: 0.0

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 2_HarmonicNumbers.sh 5
Harmonic Number: 2.283

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 2_HarmonicNumbers.sh 52
Harmonic Number: 4.517

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 2_HarmonicNumbers.sh 2
Harmonic Number: 1.500

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 2_HarmonicNumbers.sh 1
Harmonic Number: 1.0
```

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

```
  GNU nano 6.4                                3_CheckPrime.sh
n=$1
prime=1

if ((n <= 1)); then
  prime=0
else
  for ((i=2; i*i<=n; i++)); do
    if ((n % i == 0)); then
      prime=0
      break
    fi
  done
fi

if ((prime == 1)); then
  echo "Prime"
else
  echo "Not Prime"
fi
```

```
Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 3
sh: 3: No such file or directory

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 3_CheckPrime.sh
Not Prime

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 3_CheckPrime.sh 5
Prime

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 3_CheckPrime.sh 54
Not Prime

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 3_CheckPrime.sh 1
Not Prime

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 3_CheckPrime.sh 4
Not Prime

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 3_CheckPrime.sh 7
Prime
```

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

```
  GNU nano 6.4                                4_PrimeInRange.sh
start=$1
end=$2

for ((n=start; n<=end; n++)); do
  prime=1

  if ((n <= 1)); then
    prime=0
  else
    for ((i=2; i*i<=n; i++)); do
      if ((n % i == 0)); then
        prime=0
        break
      fi
    done
  fi

  if ((prime == 1)); then
    echo "$n"
  fi
done
```

```
Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 4_PrimeInRange.sh 10 12
11

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 4_PrimeInRange.sh 0 12
2
3
5
7
11

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 4_PrimeInRange.sh 0 28
2
3
5
7
11
13
17
19
23
```

5. Write a program that computes a factorial of a number taken as input.
   5 Factorial – 5! = 1 * 2 * 3 * 4 * 5

```
  GNU nano 6.4                                  5_Factorial.sh
n=$1
factorial=1

for ((i=1; i<=n; i++)); do
  factorial=$((factorial * i))
done

echo "Factorial: $factorial"
```

```
Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 5_Factorial.sh 9
Factorial: 362880

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 5_Factorial.sh 2
Factorial: 2

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 5_Factorial.sh 5
Factorial: 120
```

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.

```
  GNU nano 6.4                                  6_PrimeFactor.sh
n=$1

for ((i=2; i*i<=n; i++)); do
  while ((n % i == 0)); do
    echo "$i"
    n=$((n / i))
  done
done

if ((n > 1)); then
  echo "$n"
fi
```

```
Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 6_PrimeFactor.sh 4
2
2

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 6_PrimeFactor.sh 7
7

Shri@PRODUCTIVITY-4 MINGW64 ~/Testing_Bridge/repoPortal/repo1/D6 (main)
$ sh 6_PrimeFactor.sh 9
3
3
```