Repetition Practice Problems with for loop

Prime number:

read -p "enter a number:" a

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

do

if [ $(($a%i)) == 0 ]

then

echo "$a is not a prime num."

exit

fi

done

echo "$a is a prime num."

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash prime.sh

enter a number:3

3 is a prime num.

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash prime.sh

enter a number:9

9 is not a prime num.

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash prime.sh

enter a number:29

29 is a prime num.

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash prime.sh

enter a number:21

21 is not a prime num.

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ cat prime.sh

read -p "enter a number:" a

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

do

if [ $(($a%i)) == 0 ]

then

echo "$a is not a prime num."

exit

fi

done

echo "$a is a prime num."

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash -x prime.sh

+ read -p 'enter a number:' a

enter a number:3

+ (( i=2 ))

+ (( i<=a/2 ))

+ echo '3 is a prime num.'

3 is a prime num

Factorial problem:

read -p "enter a number:" a

fact=1

for ((i=2;i<=a;i++))

{

fact=$((fact\*i))

}

echo $fact

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ nano factorial.sh

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash factorial.sh

enter a number:4

24

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash factorial.sh

enter a number:7

5040

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash factorial.sh

enter a number:8

40320

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ cat factorial.sh

read -p "enter a number:" a

fact=1

for ((i=2;i<=a;i++))

{

fact=$((fact\*i))

}

echo $fact

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash -x factorial.sh

+ read -p 'enter a number:' a

enter a number:4

+ fact=1

+ (( i=2 ))

+ (( i<=a ))

+ fact=2

+ (( i++ ))

+ (( i<=a ))

+ fact=6

+ (( i++ ))

+ (( i<=a ))

+ fact=24

+ (( i++ ))

+ (( i<=a ))

+ echo 24

24

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash factorial2.sh

enter a number:3

1

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash factorial2.sh

enter a number:6

6

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash factorial2.sh

enter a number:9

24

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash factorial2.sh

enter a number:5

2

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ cat factorial2.sh

read -p "enter a number:" a

fact=1

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

{

fact=$((fact\*i))

}

echo $fact

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash -x factorial2.sh

+ read -p 'enter a number:' a

enter a number:4

+ fact=1

+ (( i=2 ))

+ (( i<=a/2 ))

+ fact=2

+ (( i++ ))

+ (( i<=a/2 ))

+ echo 2

2

Repetition Practice Problems with while loop

3) HEAD TAIL

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ nano while.sh

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash while.sh

1 head

1 tail

2 head

2 tail

3 head

3 tail

4 head

4 tail

5 head

5 tail

6 head

6 tail

7 head

7 tail

8 head

8 tail

9 head

9 tail

10 head

10 tail

11 head

11 tail

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ cat while.sh

i=1

while [ $i -lt 12 ]

do

echo "$i head"

echo "$i tail"

((i++))

done

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems

$ bash -x while.sh

+ i=1

+ '[' 1 -lt 12 ']'

+ echo '1 head'

1 head

+ echo '1 tail'

1 tail

+ (( i++ ))

+ '[' 2 -lt 12 ']'

+ echo '2 head'

2 head

+ echo '2 tail'

2 tail

+ (( i++ ))

+ '[' 3 -lt 12 ']'

+ echo '3 head'

3 head

+ echo '3 tail'

3 tail

+ (( i++ ))

+ '[' 4 -lt 12 ']'

+ echo '4 head'

4 head

+ echo '4 tail'

4 tail

+ (( i++ ))

+ '[' 5 -lt 12 ']'

+ echo '5 head'

5 head

+ echo '5 tail'

5 tail

+ (( i++ ))

+ '[' 6 -lt 12 ']'

+ echo '6 head'

6 head

+ echo '6 tail'

6 tail

+ (( i++ ))

+ '[' 7 -lt 12 ']'

+ echo '7 head'

7 head

+ echo '7 tail'

7 tail

+ (( i++ ))

+ '[' 8 -lt 12 ']'

+ echo '8 head'

8 head

+ echo '8 tail'

8 tail

+ (( i++ ))

+ '[' 9 -lt 12 ']'

+ echo '9 head'

9 head

+ echo '9 tail'

9 tail

+ (( i++ ))

+ '[' 10 -lt 12 ']'

+ echo '10 head'

10 head

+ echo '10 tail'

10 tail

+ (( i++ ))

+ '[' 11 -lt 12 ']'

+ echo '11 head'

11 head

+ echo '11 tail'

11 tail

+ (( i++ ))

+ '[' 12 -lt 12 ']'

(2)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.

function highestPowerof2($n)

{

$res = 0;

for ($i = $n; $i >= 1; $i--)

{

if ((($i & ($i - 1)) == 0)

{

$res = $i; break;

}

}

return $res;

}

echo highestPowerof2($n);

OUTPUT;-

Enter the number 4=2^2

4)

PALINDROME

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ nano palindrome.sh

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash palindrome.sh

Enter the number

123

321

Number is not palindrome

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ cat palindrome.sh

echo "Enter the number"

read n

function pal

{

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

}

r=`pal $n`

echo "$r"

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash -x palindrome.sh

+ echo 'Enter the number'

Enter the number

+ read n

123

++ pal 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

++ '[' 123 -eq 321 ']'

++ echo 'Number is not palindrome'

+ r='321

Number is not palindrome'

+ echo '321

Number is not palindrome'

321

Number is not palindrome

(8) 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

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash prime.sh

enter number

4

4 is not a prime number.r

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash prime.sh

enter number

3

3 is a prime number.r

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ cat prime.sh

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""r"

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash -x prime.sh

+ echo 'enter number'

enter number

+ read num

4

++ prime

++ (( i=2 ))

++ (( i<=num/2 ))

++ '[' 0 -eq 0 ']'

++ echo '4 is not a prime number.'

++ exit

+ r='4 is not a prime number.'

+ echo '4 is not a prime number.r'

4 is not a prime number.r

B)

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ nano b2.sh

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ nano palidrr.sh

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash palidrr.sh

Enter the number

123

321

Number is not palindrome

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ cat palidrr.sh

echo "Enter the number"

read n

function pal

{

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

}

r=`pal $n`

echo "$r"

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash -x palindrr.sh

bash: palindrr.sh: No such file or directory

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash -x palidrr.sh

+ echo 'Enter the number'

Enter the number

+ read n

123

++ pal 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

++ '[' 123 -eq 321 ']'

++ echo 'Number is not palindrome'

+ r='321

Number is not palindrome'

+ echo '321

Number is not palindrome'

321

Number is not palindrome

C)

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash prpal.sh

Enter the number

125

521

Number is not palindrome

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash prpal.sh

Enter the number

3

3

Number is palindrome

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash prpal.sh

Enter the number

5

5

Number is palindrome

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ cat prpal.sh

echo "Enter the number"

read n

function pal

{

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

}

r=`pal $n`

echo "$r"

Priyanka@LAPTOP-E2NR1CA4 MINGW64 ~/Desktop/practice problems/not

$ bash -x prpal.sh

+ echo 'Enter the number'

Enter the number

+ read n

125

++ pal 125

++ number=125

++ reverse=0

++ '[' 125 -gt 0 ']'

+++ expr 125 % 10

++ a=5

+++ expr 125 / 10

++ n=12

+++ expr 0 '\*' 10 + 5

++ reverse=5

++ '[' 12 -gt 0 ']'

+++ expr 12 % 10

++ a=2

+++ expr 12 / 10

++ n=1

+++ expr 5 '\*' 10 + 2

++ reverse=52

++ '[' 1 -gt 0 ']'

+++ expr 1 % 10

++ a=1

+++ expr 1 / 10

++ n=0

+++ expr 52 '\*' 10 + 1

++ reverse=521

++ '[' 0 -gt 0 ']'

++ echo 521

++ '[' 125 -eq 521 ']'

++ echo 'Number is not palindrome'

+ r='521

Number is not palindrome'

+ echo '521

Number is not palindrome'

521

Number is not palindrome

(6) Write a program that takes a command-line argument n and prints the nth harmonic number. Harmonic Number is of the form Hn=1/1+1/2+1/3+1/4+ …+1/n

function nthHarmonic($N)

{

$harmonic = 1.00;

for ($i = 2; $i=>$N; $i++)

{

$harmonic += (float)1 / $i;

}

return $harmonic;

}

// Driver Code

$N = 8;

echo nthHarmonic($N)