LABWORK -07

User defined function to check whether a given number is odd or even:

```
myNum = as.integer(readline(prompt="Enter a number: "))
Enter a number: 5
> is.even <- function(myNum) myNum %% 2 == 0
> is.even(myNum)
[1] FALSE
> is.odd <- function(myNum) myNum %% 2 != 0
> is.odd(myNum)
[1] TRUE

> myNum = as.integer(readline(prompt="Enter a number: "))
Enter a number: 10
> is.even <- function(myNum) myNum %% 2 == 0
> is.even(myNum)
[1] TRUE
> is.odd <- function(myNum) myNum %% 2 != 0
> is.odd(myNum)
[1] FALSE
```

User defined function to check whether a given number is prime or not:

```
> isprime <- function(myNum) {
+ n <- myNum/2
+ prime <- TRUE
+ for( i in 2:n) {
+ if(myNum %% i == 0)
+ prime <- FALSE
+ }
+
+ if(myNum==2)
+ prime <- TRUE
+ if(prime)
+ print(paste(myNum," is a Prime Number"))
+ else
+ print(paste(myNum," is a Composite Number"))
+ }
> isprime(10)
[1] "10  is a Composite Number"
> isprime(7)
[1] "7  is a Prime Number"
```

User defined function which get three number as arguments and finding the greatest of three numbers:

```
> maximum = function(num1,num2,num3)
+    {
+    max=num1;
+    if(num2>max)
+    max=num2;
+    if(num3>max)
+    max=num3;
+    return(max);
+    }
> maximum(10,20,30)
[1] 30
> maximum(-10,-20,-30)
[1] -10
```

Concatenation of strings using paste function:

```
> paste ("A", 1:6, sep = "")
[1] "A1" "A2" "A3" "A4" "A5" "A6"
> paste ("Today is", date( ) )
[1] "Today is Mon Sep 14 17:53:00 2020"
```

Formatting numbers and strings:

```
> myString <- "Hello my name is Beulah"          # printing myString
> print(myString)
[1] "Hello my name is Beulah"

> print(myString, quote = FALSE)                 # printing myString without quotes
[1] Hello my name is Beulah

# concatenating and printing using cat() method
> myString <- "Hello my name is Beulah"
> cat(myString, "Evanjalin\n")
Hello my name is Beulah Evanjalin

> cat(myString, "Evanjalin\n", sep="--")
Hello my name is Beulah—Evanjalin

# format ( ) function
> format(1)
[1] "1"
> format(13.4354636)
[1] "13.43546"
> format(13.4354677)
[1] "13.43547"
```

```
> format(13.4354677, nsmal = 3)
[1] "13.43547"
> format(13, nsmal = 3)
[1] "13.000"
> format(13.8, nsmal = 3)
[1] "13.800"

# format ( ) function to align the given input
> format("hello",  justify = "centre", width = 10)
[1]  "  hello   "
> format("hello",  justify = "left", width = 10)
[1]  "hello     "
> format("hello",  justify = "right", width = 10)
[1]  "     hello"
> format("hello",  justify = "none", width = 10)
[1]  "hello"
```

Counting number of characters in a string:

```
> myVecOfChars <- c ( "10345" , "!@#$%^" , "{a,b,c}" , "o" , "What a mighty language!" )
> nchar (myVecOfChars)
[1]   5   6   7   1  23
```

Changing case of the string:

```
> myName <- "Beulah Evanjalin"
> toupper (myName)
[1]  "BEULAH EVANJALIN"
> tolower (myName)
[1]  "beulah evanjalin"
```

Extracting substring from a string:

```
> substring("Beulah", 2)
[1] "eulah"
> substring("Beulah", 2, 2)
[1] "e"
> substring("Beulah", 2, 4)
[1] "eul"
> substring("Beulah", 4)
[1] "lah"
> substring("Beulah", 0, 4)
[1] "Beul"
```