# Qno_06

Arpan Sapkota

2023-05-06

## a

```
#Defining integers from 1 to 15 in 3 approaches
I <- 1:15
I <- seq(1, 15, by = 1)
I <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15)
```

## b

```
N <- c(1.1, 2.2, 3.3, 4.4, 5.5)
```

## c

```
add <- I + N
add
```

```
##  [1]  2.1  4.2  6.3  8.4 10.5  7.1  9.2 11.3 13.4 15.5 12.1 14.2 16.3 18.4 20.5
```

```
subtract <- I - N
subtract
```

```
##  [1] -0.1 -0.2 -0.3 -0.4 -0.5  4.9  4.8  4.7  4.6  4.5  9.9  9.8  9.7  9.6  9.5
```

```
multiply <- I * N
multiply
```

```
##  [1]  1.1  4.4  9.9 17.6 27.5  6.6 15.4 26.4 39.6 55.0 12.1 26.4 42.9 61.6 82.5
```

```
divide <- I / N
divide
```

```
##  [1]  0.9090909  0.9090909  0.9090909  0.9090909  0.9090909  5.4545455
##  [7]  3.1818182  2.4242424  2.0454545  1.8181818 10.0000000  5.4545455
## [13]  3.9393939  3.1818182  2.7272727
```

```
#Adding I to N: the first element of the result is 2.1, which is obtained by adding 1 to 1.1. Similarly, the seco
nd element of the result is 4.2, which is obtained by adding 2 to 2.2, and so on.

#Subtracting N from I: The first element of the result is -0.1, which is obtained by subtracting 1.1 from 1. Simi
larly, the second element of the result is -0.2, which is obtained by subtracting 2.2 from 2, and so on.

#Multiplying I and N: the first element of the result is 1.1, which is obtained by multiplying 1 by 1.1. Similarl
y, the second element of the result is 4.84, which is obtained by multiplying 2 by 2.2, and so on.

#Dividing I by N: the first element of the result is 0.9090909, which is obtained by dividing 1 by 1.1. Similarly
, the second element of the result is 1.3636364, which is obtained by dividing 2 by 2.2, and so on.
```

## d Matrix multiplication

```
I <- 1:15
N <- c(1.1, 2.2, 3.3, 4.4, 5.5)


#It is not possible to perform matrix multiplication between the vector I and the vector N because they have diff
erent lengths.

#Matrix multiplication is only defined between matrices with compatible dimensions, which means that the number o
f columns in the first matrix must be equal to the number of rows in the second matrix.
```

## e Transpose I and N and Matrix Multiplication

```
I <- 1:15
N <- c(1.1, 2.2, 3.3, 4.4, 5.5)

I <- matrix(I, nrow = 1)
N <- matrix(N, nrow = 5)

I_transposed <- t(I)
N_transposed <- t(N)

I_transposed %*% N_transposed
```

```
##       [,1] [,2] [,3] [,4] [,5]
##  [1,]  1.1  2.2  3.3  4.4  5.5
##  [2,]  2.2  4.4  6.6  8.8 11.0
##  [3,]  3.3  6.6  9.9 13.2 16.5
##  [4,]  4.4  8.8 13.2 17.6 22.0
##  [5,]  5.5 11.0 16.5 22.0 27.5
##  [6,]  6.6 13.2 19.8 26.4 33.0
##  [7,]  7.7 15.4 23.1 30.8 38.5
##  [8,]  8.8 17.6 26.4 35.2 44.0
##  [9,]  9.9 19.8 29.7 39.6 49.5
## [10,] 11.0 22.0 33.0 44.0 55.0
## [11,] 12.1 24.2 36.3 48.4 60.5
## [12,] 13.2 26.4 39.6 52.8 66.0
## [13,] 14.3 28.6 42.9 57.2 71.5
## [14,] 15.4 30.8 46.2 61.6 77.0
## [15,] 16.5 33.0 49.5 66.0 82.5
```

```
#We have taken the dot product of the transpose of N and I. This is equivalent to multiplying each element of N w
ith the corresponding element of I, summing the products, and returning a scalar value.
```