**EXPERIMENT – 7**

**Aim:** Introduction to R Programming and implementation of basic operations in R

**Introduction:** R is a programming language and free software developed by Ross Ihaka and Robert Gentleman in 1993. R possesses an extensive catalog of statistical and graphical methods. It includes machine learning algorithms, linear regression, time series, statistical inference to name a few. Most of the R libraries are written in R, but for heavy computational tasks, C, C++ and Fortran codes are preferred. R is not only entrusted by academic, but many large companies also use R programming language, including Uber, Google, Airbnb, Facebook and so on.

Data analysis with R is done in a series of steps; programming, transforming, discovering, modelling and communicate the results.

**Program:** R is a clear and accessible programming tool

**Transform:** R is made up of a collection of libraries designed specifically for data science

**Discover:** Investigate the data, refine your hypothesis and analyse them

**Model:** R provides a wide array of tools to capture the right model for your data

**Communicate:** Integrate codes, graphs, and outputs to a report with R Markdown or build

Shiny apps to share with the world

What is R used for?

1. Statistical inference
2. Data analysis
3. Machine learning algorithm

**Assignment Operators**: Used to assign values in R, such as <- or =, e.g., x <- 3 assigns the value 3 to variable x.

**Basic Arithmetic**: R can perform basic arithmetic operations following the PEMDAS order of operations, including parentheses, exponentiation, multiplication, division, addition, and subtraction.

**R Objects**:

**Vectors:** Arrays in R holding multiple data values of the same type, starting indexing from 1, created using functions like c() or seq().

**Lists:** Generic objects in R consisting of an ordered collection of objects, one-dimensional and heterogeneous, created using the list() function.

**Matrices:** Rectangular arrangements of numbers in rows and columns, created using the matrix() function, with rows and columns named using rownames() and colnames().

**Arrays:** Data storage structures with a fixed number of dimensions, including uni-dimensional arrays (vectors) and multi-dimensional arrays, created using the array() function.

**Basic Operations Implementation :**

#Numeric

num\_var <- 3.14

print(num\_var)

* Integer

int\_var <- 42L

print(int\_var)

# Character

char\_var <- "Hello, R!"

print(char\_var)

# Logical

logical\_var <- TRUE

print(logical\_var)

1. <- 10
2. <- 5

# Addition

addition\_result <- x + y

print(addition\_result)

#Subtraction

subtraction\_result <- x – y

print(subtraction\_result)

#Multiplication

multiplication\_result <- x \* y

print(multiplication\_result)

* Division

division\_result <- x / y

print(division\_result)

#Exponentiation

exponentiation\_result <- x ^ y

print(exponentiation\_result)

# Modulo

modulo\_result <- x %% y

print(modulo\_result)

# Matrix

matrix\_data <- matrix(1:6, nrow = 2, ncol = 3)

print(matrix\_data)

# Creating a numeric vector

numeric\_vector <- c(1, 2, 3, 4, 5)

print(numeric\_vector)

* Creating a character vector

char\_vector <- c("apple", "banana", "cherry")

print(char\_vector)

**Output:**

