**Progressive Education Society's**

**Modern College of Engineering, Pune MCA Department** **A.Y.2023-24**

**(410908) Data Science laboratory**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Class: SY-MCA Shift / Div : B Roll Number : 51119

Name: Harsh Ghodke Assignment No:2 Date of Implementation: 10-08-2024 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q.1) Write a R program to create three vectors a,b,c with 3 integers. Combine the three vectors to become a 3×3 matrix where each column represents a vector.Print the content of the matrix.

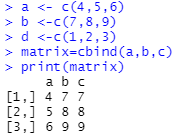
Ans : > a <- c(4,5,6)

> b <-c(7,8,9)

> d <-c(1,2,3)

> matrix=cbind(a,b,c)

> print(matrix)



2) Write a R program to create a list containing a vector, a matrix and a list and give names to the elements in the list. Access the first and second element of the list.

Ans : vector <- c(1, 2, 3)

matrix <- matrix(1:9, nrow=3, byrow=TRUE)

inner\_list <- list(a = 1, b = 2)

my\_list <- list(my\_vector = vector, my\_matrix = matrix, my\_inner\_list = inner\_list)

print("Entire List:")

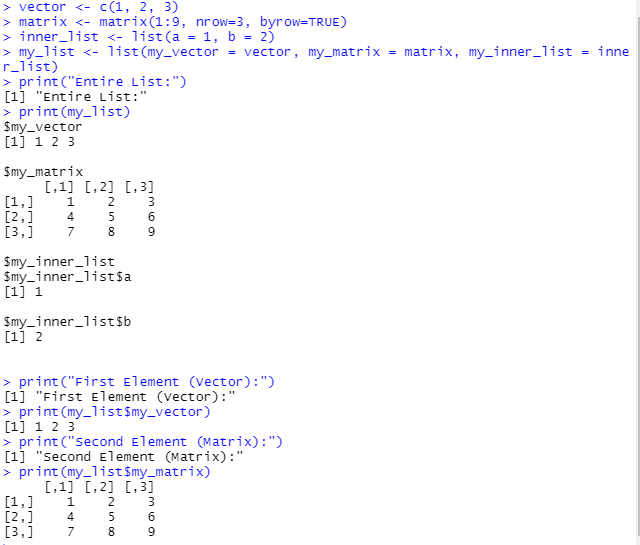
print(my\_list)

print("First Element (Vector):")

print(my\_list$my\_vector)

print("Second Element (Matrix):")

print(my\_list$my\_matrix)



Q.3) Write a R program to create an array with three columns, three rows, and two &quot;tables&quot;, taking two vectors as input to the array. Print the array.

Ans :

vector1 <- c(1, 2, 3, 4, 5, 6, 7, 8, 9)

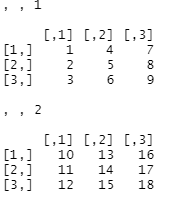
vector2 <- c(10, 11, 12, 13, 14, 15, 16, 17, 18)

combined\_vector <- c(vector1, vector2)

my\_array <- array(combined\_vector, dim = c(3, 3, 2))

print("The Array:")

print(my\_array)



4) Write a R program to create a data frame from four given vectors name = c('Anastasia','Dima','Katherine','James','Emaily','Michael','Matthew',Laura','Kevin','Jonas') score = c(12.5, 9, 16.5, 12, 9, 20, 14.5, 13.5, 8, 19) attempts = c(1, 3, 2, 3, 2, 3, 1, 1, 2, 1) qualify=c('yes','no','yes','no','no','yes','yes','no','no','yes')

Ans :

name <- c('Anastasia','Dima','Katherine','James','Emily','Michael','Matthew','Laura','Kevin','Jonas')

score <- c(12.5, 9, 16.5, 12, 9, 20, 14.5, 13.5, 8, 19)

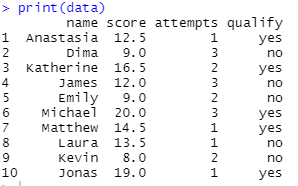
attempts <- c(1, 3, 2, 3, 2, 3, 1, 1, 2, 1)

qualify <- c('yes','no','yes','no','no','yes','yes','no','no','yes')

data <- data.frame(name, score, attempts, qualify)

print("The Data Frame:")

print(data)



Q.5) Write a R program to create a factor corresponding to height of women data set, which contains height and weights for a sample of women

Ans : height <- c(58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70)

weight <- c(115, 117, 120, 123, 126, 129, 132, 135, 139, 142, 146, 150, 154)

women\_data <- data.frame(height, weight)

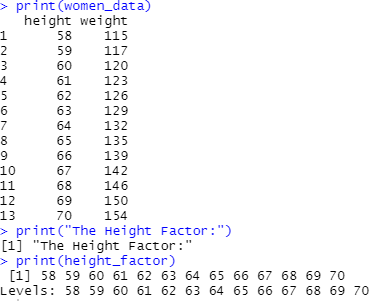
height\_factor <- factor(women\_data$height)

print("The Women Data Frame:")

print(women\_data)

print("The Height Factor:")

print(height\_factor)



Q.6) Use R to create the following two matrices and do the indicated matrix multiplication. matrix of 2row and 3 coulumn having element (7,9,12,2,4,13) and create matrix of 3 row and 4coulumn having element (1,7,12,19,2,8,13,20,3,9,14,21) What is the resulting matrix?

Ans :

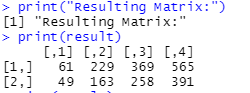
matrix1 <- matrix(c(7, 9, 12, 2, 4, 13), nrow = 2, ncol = 3, byrow = TRUE)

matrix2 <- matrix(c(1, 7, 12, 19, 2, 8, 13, 20, 3, 9, 14, 21), nrow = 3, ncol = 4, byrow = TRUE)

result <- matrix1 %\*% matrix2

print("Resulting Matrix:")

print(result)



Q.7) WAP to Print the Fibonacci Sequence.

Ans : fibonacci\_sequence <- function(n) {

fib <- numeric(n)

fib[1] <- 0

if (n > 1) {

fib[2] <- 1

} for (i in 3:n) {

fib[i] <- fib[i-1] + fib[i-2]

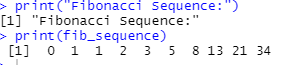
}return(fib)

}n <- 10

fib\_sequence <- fibonacci\_sequence(n)

print("Fibonacci Sequence:")

print(fib\_sequence)



Q.10) Write a R program to create an array of two 3x3 matrices each with 3 rows and

3 columns from two vectors. Print the second row of the second matrix of the

array and the element in the 3rd row and 3rd column of the 1st matrix.

Ans :

vector1 <- c(1, 2, 3, 4, 5, 6, 7, 8, 9)

vector2 <- c(10, 11, 12, 13, 14, 15, 16, 17, 18)

matrix1 <- matrix(vector1, nrow = 3, byrow = TRUE)

matrix2 <- matrix(vector2, nrow = 3, byrow = TRUE)

my\_array <- array(c(matrix1, matrix2), dim = c(3, 3, 2))

second\_row\_matrix2 <- my\_array[2, , 2]

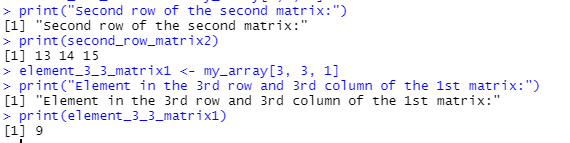
print("Second row of the second matrix:")

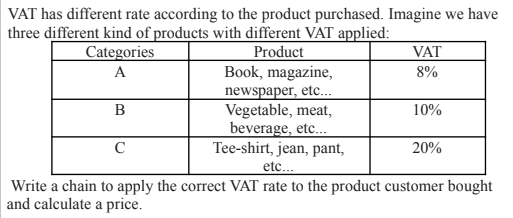
print(second\_row\_matrix2)

element\_3\_3\_matrix1 <- my\_array[3, 3, 1]

print("Element in the 3rd row and 3rd column of the 1st matrix:")

print(element\_3\_3\_matrix1)



Q.11)

Ans:

calculate\_price\_with\_vat <- function(product\_category, original\_price) {

vat\_rates <- list(

A = 0.08,

B = 0.10,

C = 0.20 )if (!product\_category %in% names(vat\_rates)) {

stop("Invalid product category.")

}vat\_rate <- vat\_rates[[product\_category]]

final\_price <- original\_price \* (1 + vat\_rate)

return(final\_price)

}product\_category <- "C"

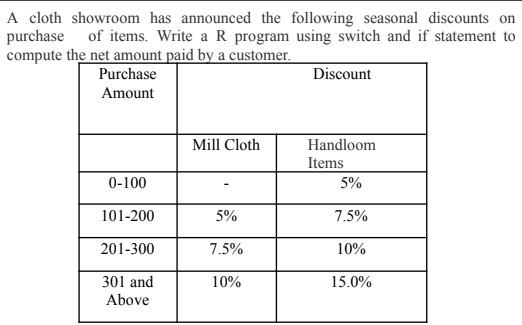
original\_price <- 100

final\_price <- calculate\_price\_with\_vat(product\_category, original\_price)

cat("The final price including VAT is:", final\_price, "\n")



Q.12)



Ans :

calculate\_net\_amount <- function(purchase\_amount, item\_type) {

if (purchase\_amount <= 100) {

discount\_percent <- 0.05

} else if (purchase\_amount <= 200) {

discount\_percent <- ifelse(item\_type == "Mill", 0.05, 0.075)

} else if (purchase\_amount <= 300) {

discount\_percent <- ifelse(item\_type == "Mill", 0.075, 0.10)

} else {

discount\_percent <- ifelse(item\_type == "Mill", 0.10, 0.15)

}

discount\_amount <- purchase\_amount \* discount\_percent

net\_amount <- purchase\_amount - discount\_amount

return(net\_amount)

}

purchase\_amount <- 250

item\_type <- "Handloom"

net\_amount <- calculate\_net\_amount(purchase\_amount, item\_type)

cat("The net amount to be paid is:", net\_amount, "\n")



Q.13. Find Sum of Series 12+22+32+.....+n2.

Ans :

sum\_of\_squares <- function(n) {

if (n <= 0) {

stop("The value of n must be a positive integer.")

}

sum <- (n \* (n + 1) \* (2 \* n + 1)) / 6

return(sum)

}

n <- 5

result <- sum\_of\_squares(n)

cat("The sum of squares from 1^2 to", n^2, "is:", result, "\n")



Q.14) Write a R program to print the numbers from 1 to 100 and print "Fizz" for multiples of

3, print "Buzz" for multiples of 5, and print "FizzBuzz" for multiples of both.

Ans :

fizz\_buzz <- function() {

for (i in 1:100) {

if (i %% 3 == 0 && i %% 5 == 0) {

print("FizzBuzz")

} else if (i %% 3 == 0) {

print("Fizz")

} else if (i %% 5 == 0) {

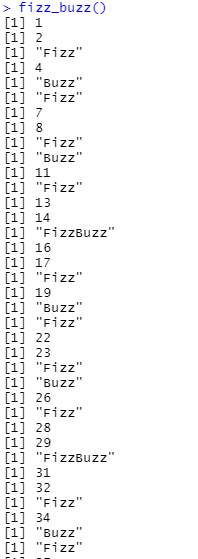
print("Buzz")

} else {

print(i)

}}}

fizz\_buzz()



Q.15) Write a R Program to find the sum of digits of a number reducing it to one digit using

repeat loop.

Ans :

digital\_root <- function(n) {

if (n <= 0) {

stop("The number must be a positive integer.")

}

sum\_of\_digits <- n

repeat {

digits <- as.numeric(strsplit(as.character(sum\_of\_digits), NULL)[[1]])

sum\_of\_digits <- sum(digits)

if (sum\_of\_digits < 10) {

break}

} return(sum\_of\_digits)

}number <- 9875

result <- digital\_root(number)

cat("The digital root of", number, "is:", result, "\n")

