ITA0464-Statistics with R Programming

LAB EXPERIMENT(DAY-1)

1. Write a R program to take input from the user (name and age) and display the values. Also print the version of R installation.

```
> name <- readline(prompt="Enter your name: ")
Enter your name: sriram
> age <- as.numeric(readline(prompt="Enter your age: "))
Enter your age: 18
> cat("Your name is:", name, "\n")
Your name is: sriram
> cat("Your age is:", age, "\n")
Your age is: 18
> cat("R version:", R.version.string, "\n")
R version: R version 4.3.2 (2023-10-31 ucrt)
>
```

2. Write a R program to get the details of the objects in memory

```
> x <- 1:10
> y <- "Hello"
> z <- list(a = 1, b = 2, c = 3)
> object_names <- ls()
> object_sizes <- sapply(object_names, function(obj) object.size(get(obj)))
> cat("Details of objects in memory:\n")
Details of objects in memory:
> for (i in 1:length(object_names)) {
+   cat("Object:", object_names[i], "- Size:", object_sizes[i], "bytes\n")
+ }
Object: age - Size: 56 bytes
Object: character_data - Size: 248 bytes
Object: logical_data - Size: 64 bytes
Object: name - Size: 112 bytes
Object: numeric_data - Size: 80 bytes
Object: x - Size: 96 bytes
Object: y - Size: 112 bytes
Object: z - Size: 608 bytes
```

3. Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91

```
> sequence_20_to_50 <- 20:50
> mean_20_to_60 <- mean(20:60)
> sum_51_to_91 <- sum(51:91)
> cat("Sequence of numbers from 20 to 50:", sequence_20_to_50, "\n")
Sequence of numbers from 20 to 50: 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
> cat("Mean of numbers from 20 to 60:", mean_20_to_60, "\n")
Mean of numbers from 20 to 60: 40
> cat("Sum of numbers from 51 to 91:", sum_51_to_91, "\n")
Sum of numbers from 51 to 91: 2911
>
```

4. Write a R program to create a vector which contains 10 random integer values between -50 and +50.

```
> set.seed(42)
> random_values <- sample(-50:50, 10, replace = TRUE)
> cat("Vector containing 10 random integer values between -50 and +50:\n")
Vector containing 10 random integer values between -50 and +50:
> print(random_values)
 [1]  -2  50  14 -26  23  49 -33  -2  -4 -27
>
```

5. Write a R program to get the first 10 Fibonacci numbers.

```
> fibonacci <- function(n) {
+   fib <- c(0, 1)
+   for (i in 3:n) {
+     fib[i] <- fib[i-1] + fib[i-2]
+   }
+   return(fib[1:n])
+ }
> first_10_fibonacci <- fibonacci(10)
> print(first_10_fibonacci)
 [1]  0  1  1  2  3  5  8 13 21 34
>
```

6. Write a R program to get all prime numbers up to a given number (based on the sieve of Eratosthenes)

```
> prime_numbers <- function(n) {
+ if (n >= 2) {
+  x = seq(2, n)
+  prime_nums = c()
+  for (i in seq(2, n)) {
+  if (any(x == i)) {
+  prime_nums = c(prime_nums, i)
+  x = c(x[(x %% i) != 0], i)
+  }
+  }
+  return(prime_nums)
+  }
+  else
+  {
+  stop("Input number should be at least 2.")
+  }
+  }
> prime_numbers(12)
[1]  2  3  5  7 11
>
```

7. 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.

```
> 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)
+   }
+ }
[1] 1
[1] 2
[1] "Fizz"
[1] 4
[1] "Buzz"
[1] "Fizz"
[1] 7
[1] 8
[1] "Fizz"
[1] "Buzz"
[1] 11
[1] "Fizz"
[1] 13
[1] 14
[1] "FizzBuzz"
[1] 16
[1] 17
[1] "Fizz"
[1] 19
[1] "Buzz"
[1] "Fizz"
[1] 22
[1] 23
[1] "Fizz"
[1] "Buzz"
[1] 26
```

8. Write a R program to extract first 10 English letters in lower case and last 10 letters in upper case and extract letters between 22nd to 24th letters in upper case.

```
> first_10_lower <- letters[1:10]
> last_10_upper <- toupper(tail(letters, 10))
> letters_between_22_24_upper <- toupper(letters[22:24])
> print("First 10 English letters in lower case:")
[1] "First 10 English letters in lower case:"
> print(first_10_lower)
 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j"
>
> print("\nLast 10 letters in upper case:")
[1] "\nLast 10 letters in upper case:"
> print(last_10_upper)
 [1] "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z"
>
> print("\nLetters between 22nd to 24th letters in upper case:")
[1] "\nLetters between 22nd to 24th letters in upper case:"
> print(letters_between_22_24_upper)
[1] "V" "W" "X"
```

9. Write a R program to find the factors of a given number

```
> find_factors <- function(n) {
+   factors <- c()
+   for (i in 1:n) {
+     if (n %% i == 0) {
+       factors <- c(factors, i)
+     }
+   }
+   return(factors)
+ }
>
> # Example usage:
> number <- 36  # Change this to any number you want to find factors for
> factors_of_number <- find_factors(number)
> print(paste("Factors of", number, "are:", paste(factors_of_number, collapse = ", ")))
[1] "Factors of 36 are: 1, 2, 3, 4, 6, 9, 12, 18, 36"
>
```

10. Write a R program to find the maximum and the minimum value of a given vector

```
> find_max_min <- function(vector) {
+   max_value <- max(vector)
+   min_value <- min(vector)
+   return(list(maximum = max_value, minimum = min_value))
+ }
> input_vector <- c(3, 7, 2, 9, 1, 5)
> result <- find_max_min(input_vector)
> print(paste("Maximum value:", result$maximum))
[1] "Maximum value: 9"
> print(paste("Minimum value:", result$minimum))
[1] "Minimum value: 1"
```