ITA0450-STATISTICS WITH R-PROGRAMMING

PROGRAM:1.NAME & AGE

```
name <- readline(prompt="enter your name")
age <-as.integer(readline(prompt="enter your age"))
print(paste("naveen",name,"next year you will be",age,"years old"))
print(R.version.string)
```

PROGRAM:2.OBJECTS IN MEMORY

```
name="python";
n1=10;
n2=0.5
nums = c(10,20,30,40,50,60)
print(ls())
print("Details of the objects in memory:")
print(ls.str())
```

PROGRAM:4.CREATE A VECTOR

```
v=sample(-50:50,10, replace=TRUE)
print(“Content of the vector:”)
print(“10 random integer values between -50 and +50:”)
print(v)
```

PROGRAM:5.FIBONACCI SERIES

```
nterms = as.integer(readline(prompt="How many terms? "))
n1 = 0
n2 = 1
count = 2
if(nterms <= 0) {
```

```
 print("Plese enter a positive integer")
 } else {
  if(nterms == 1) {
  print("Fibonacci sequence:")
  print(n1)
 } else {
  print("Fibonacci sequence:")
  print(n1)
  print(n2)
 while(count < nterms) {
  nth = n1 + n2
  print(nth)
  # update values
  n1 = n2
  n2 = nth
  count = count + 1
   }
 }
}
```

PROGRAM:6.PRIME NUMBERS

```
  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)
```

PROGRAM:7.FIZZBUZZ

```
for (n in 1:100) {
if (n %% 3 == 0 & n %% 5 == 0) {print("FizzBuzz")}
else if (n %% 3 == 0) {print("Fizz")}
else if (n %% 5 == 0) {print("Buzz")}
else print(n)
}
```

PROGRAM:8.ENGLISH LETTER IN LOWER & UPPER CASES

```
print("First 10 letters in lower case:")
t = head(letters, 10)
Print(t)
print("Last 10 letters in upper case:")
t = tail(LETTERS, 10)
print(t)
print("Letters between 22nd to 24th letters in upper case:")
e = tail(LETTERS[22:24])
print(e)
```

PROGRAM:9.TO FIND FACTORS OF A NUMBER

```
print_factors = function(n) {
  print(paste("The factors of",n,"are:"))
  for(i in 1:n) {
    if((n %% i) == 0) {
     print(i)
```

```
  }
 }
}
print_factors(4)
print_factors(7)
print_factors(12)
```

PROGRAM:10.MAX & MIN VALUES OF VECTOR

```
nums = c(10, 20, 30, 40, 50, 60)
print('Original vector:')
print(nums)
print(paste("Maximum value of the said vector:",max(nums)))
print(paste("Minimum value of the said vector:",min(nums)))
```

PROGRAM:11.UNIQUE ELEMENTS S IN STRING & VECTOR ,LABEL

```
str1 = "The quick brown fox jumps over the lazy dog."
print("Original vector(string)")
print(str1)
print("Unique elements of the said vector:")
print(unique(tolower(str1)))
nums = c(1, 2, 2, 3, 4, 4, 5, 6)
print("Original vector(number)")
print(nums)
print("Unique elements of the said vector:")
print(unique(nums))
```

PROGRAM:12.CREATE A 3*3 MATRIX

```
a<-c(1,2,3)
b<-c(4,5,6)
c<-cbind(a,b,c)
print(“Content of the said matrix:”
print(m)
```

PROGRAM:13.NORMAL dISTRIBUTION & COUT OCCURANS

```
n = floor(rnorm(1000, 50, 100))
print('List of random numbers in normal distribution:')
print(n)
t = table(n)
print("Count occurrences of each value:")
print(t)
```

PROGRAM: 14.READ.CSV.FILE

```
movie_data = read.csv(file="movies.csv", header=TRUE, sep="
print(   "Content of the .csv file:")
print(movie_data)
```

PROGRAM:15.CREATE THREE TYPES VECTORS

```
a = c(1, 2, 5, 3, 4, 0, -1, -3)
b = c("Red", "Green", "White")
c = c(TRUE, TRUE, TRUE, FALSE, TRUE, FALSE)
print(a)
print(typeof(a))
print(b)
print(typeof(b))
print(c)
print(typeof(c))
```

PROGRAM:18.CREATE AN ARRAY WITH 3(COLS,NORM

```
v1 = c(1, 3, 5, 7)
v2 = c(2, 4, 6, 8, 10)
arra1 = array(c(v1, v2),dim = c(3,3,2))
print(arra1)
```

PROGRAM:19.CREATE A LIST OF ELEMENTS

```
l = list(
c(1, 2, 2, 5, 7, 12),
month.abb,
matrix(c(3, -8, 1, -3), nrow = 2),
asin
)
print("Content of the list:")
```

PROGRAM:20.PLOT EMPTY GRAPHS

```
#print("Empty plot:")
plot.new()
#print("Empty plot specify the axes limits of the graphic:")
plot(1, type="n", xlab="", ylab="", xlim=c(0, 20), ylim=c(0, 20))
```