Practical 1

**Aim:** Write a R program to make a Simple Calculator.  Take input from the user and display the result.

**Code:**

rep <- TRUE

while(rep) {

cat("\n\n.....Simple Calculator.....\n")

cat("1] for Addition\n")

cat("2] for Subtraction\n")

cat("3] for Multiplication\n")

cat("4] for Division\n")

cat("5] for Modulo\n")

cat("6] for Exit\n")

cat("Enter your choice : ")

choice <- as.integer(readline())

if(choice >= 1 && choice <= 6) {

if(choice == 6) {

cat("\n\nThanks for visit")

rep <- FALSE

} else {

cat("\nEnter number 1 : ")

num1 <- as.integer(readline())

cat("\nEnter number 1 : ")

num2 <- as.integer(readline())

if(choice == 1) {

cat("\nThe addition of ", num1, " and ", num2, " is : ", (num1 + num2))

} else if(choice == 2) {

cat("\nThe subtraction of ", num1, " and ", num2, " is : ", (num1 - num2))

} else if(choice == 3) {

cat("\nThe multiplication of ", num1, " and ", num2, " is : ", (num1 \* num2))

} else if(choice == 4) {

cat("\nThe division of ", num1, " and ", num2, " is : ", (num1 / num2))

} else if(choice == 5) {

cat("\nThe modulo of ", num1, " and ", num2, " is : ", (num1 %% num2))

}

}

} else {

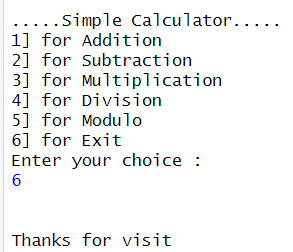
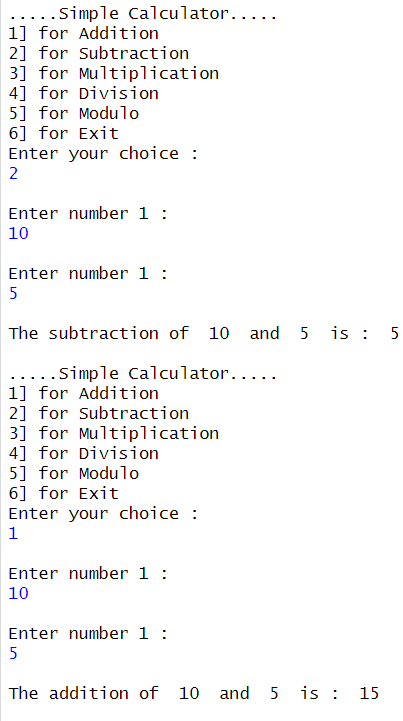
cat("\n\nInvalid Input")

rep <- FALSE

}

}

**Output:**



Practical 2

**Aim**: Write a R program to get the first 5 Fibonacci numbers.

**Code:**

cat("\n\n.....Fibonacci Series.....\n")

fib = function(n) {

a <- 0

b <- 1

c <- 0

cat(a, b)

for(i in 1:(n-2)) {

c <- (a + b)

cat("", c)

a <- b

b <- c

}

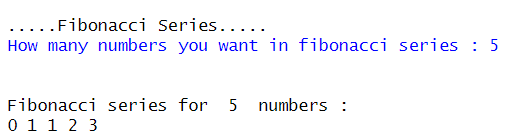
}

n <- as.integer(readline(prompt = "How many numbers you want in fibonacci series : "))

cat("\n\nFibonacci series for ", n, " numbers : \n")

fib(n)

**Output:**

****

Practical 3

**Aim:** Write a R program to Find the Factorial of a Number Using Recursion.

**Code:**

cat("\n\n.....Recursive Factorial.....\n")

fact = function(num) {

if(num == 1) {

return(1)

} else {

return(num \* factorial(num - 1))

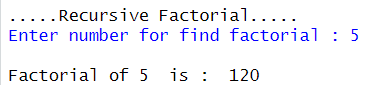
}

}

num <- as.numeric(readline(prompt = "Enter number for find factorial : "))

cat("\nFactorial of", num, " is : ", fact(num))

**Output:**

****

Practical 4

**Aim:**Write a R program to get all prime numbers up to a given number.

**Code:**

cat("\n\n.....Prime Number in Given Range.....\n")

primeCheck = function(min, max) {

for(i in min:max) {

flag <- TRUE

for(j in 2:(i - 1)) {

if(i %% j == 0) {

flag <- FALSE

break

}

}

if(flag == TRUE) {

cat("", i)

}

}

}

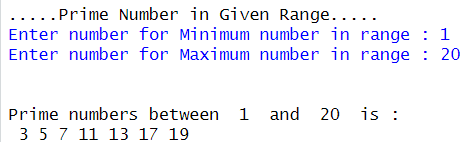
min <- as.numeric(readline(prompt = "Enter number for Minimum number in range : "))

max <- as.numeric(readline(prompt = "Enter number for Maximum number in range : "))

cat("\n\nPrime numbers between ", min, " and ", max, " is : \n")

primeCheck(min, max)

**Output:**

****

Practical 5

**Aim:** Write a R program to find the maximum and the minimum value of a given vector (values: 5,10,20,23,39).

**Code:**

cat("\n\n.....Find Minimum and Maximum from Given Vector.....\n")

values <- readline(prompt = "Enter values for vector (space-separated) : ")

input <- as.integer(strsplit(values, " ")[[1]])

i <- 0

min <- 0

max <- 0

for(value in input) {

if(i == 0) {

min <- value

i <- i + 1

}

if(min > value) {

min <- value

}

if(max < value) {

max <- value

}

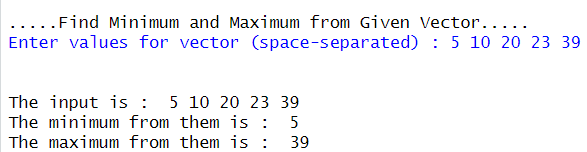
}

cat("\n\nThe input is : ", input)

cat("\nThe minimum from them is : ", min)

cat("\nThe maximum from them is : ", max)

**Output:**

****

Practical 6

**Aim:** Write a R program to create a simple bar plot of your last semester subjects marks.

**Code:**

cat("\n\n.....Plot of Last Semester Subject Marks.....\n")

conm <- as.integer(readline(prompt = "Enter marks for Computer Oriented Numerical Method : "))

os <- as.integer(readline(prompt = "Enter marks for Operating System : "))

awt <- as.integer(readline(prompt = "Enter marks for Advance Web Technology : "))

co <- as.integer(readline(prompt = "Enter marks for Computer Organization : "))

ds <- as.integer(readline(prompt = "Enter marks for Data Structure : "))

string <- paste(conm, os, awt, co, ds , sep = " ")

marks <- as.integer(strsplit(string, " ")[[1]])

subjects <- c("CONM", "OS", "AWT", "CO", "DS")

if(conm > 100 || os > 100 || awt > 100 || co > 100 || ds > 100) {

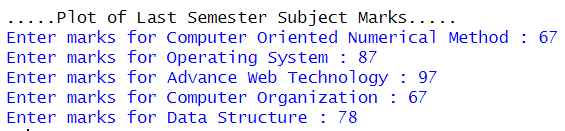
cat("\n\nInvalid Input for Marks. Please Try Again")

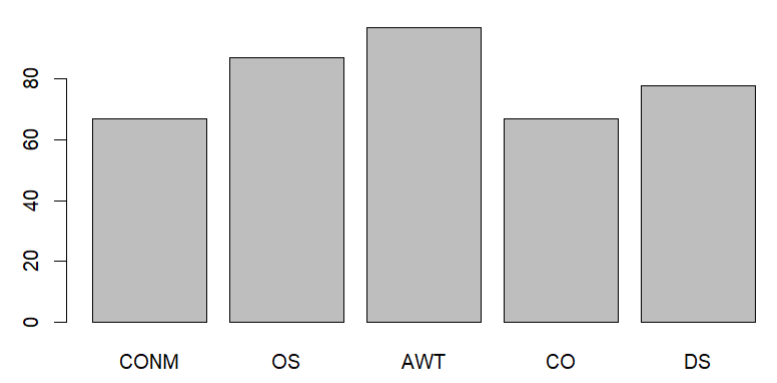
} else {

barplot(marks, names.arg = subjects)

}

**Output:**

****

****

Practical 7:

Write a R program to find sum of natural numbers.

**Code:**

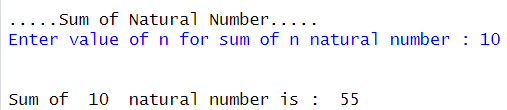
cat("\n\n.....Sum of Natural Number.....\n")

n <- as.numeric(readline(prompt = "Enter value of n for sum of n natural number : "))

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

cat("\n\nSum of ", n, " natural number is : ", sum)

**Output:**

****

Practical 8

**Aim:** Write a R program to print the multiplication table of a number**.**

**Code:**

cat("\n\n.....Multiplication Table for Given Number.....\n")

num <- as.numeric(readline(prompt = "Enter number for multiplication table : "))

cat("\n\nMultiplication table of ", num, " is : \n")

for(i in 1:10) {

if(i == 10) {

cat(num, " X ", i, " = ", (num \* i), "\n")

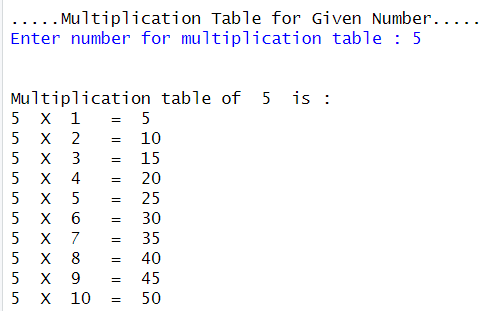
} else {

cat(num, " X ", i, " = ", (num \* i), "\n")

}

}

**Output:**

****

Practical 9

**Aim:** Write a R program to create a Data Frames which contain details of 3 Students and display summary of the data. (Name , ID , Branch , Semester).

**Code:**

cat("\n\n.....Data Frame for Student Details.....\n")

num <- as.integer(readline(prompt = "Enter no of student :"))

NameVector <- ""

for(i in 1:num) {

cat("\n\nEnter Data for Student", i, "\n")

Name <- readline(prompt = "Enter Your Name : ")

ID <- as.integer(readline(prompt = "Enter Your ID No. : "))

Branch <- readline(prompt = "Enter Your Branch : ")

Sem <- as.integer(readline(prompt = "Enter Your Semester : "))

if(i == 1) {

NameVector <- Name

IDVector <- ID

BranchVector <- Branch

SemVector <- Sem

} else {

NameVector <- paste(NameVector, Name, sep = " ")

IDVector <- paste(IDVector, ID, sep = " ")

BranchVector <- paste(BranchVector, Branch, sep = " ")

SemVector <- paste(SemVector, Sem, sep = " ")

}

}

names <- strsplit(NameVector, " ")[[1]]

IDs <- strsplit(IDVector, " ")[[1]]

Branches <- strsplit(BranchVector, " ")[[1]]

Semesters <- strsplit(SemVector, " ")[[1]]

Data\_Frame <- data.frame(

name = names,

Id = IDs,

Branch = Branches,

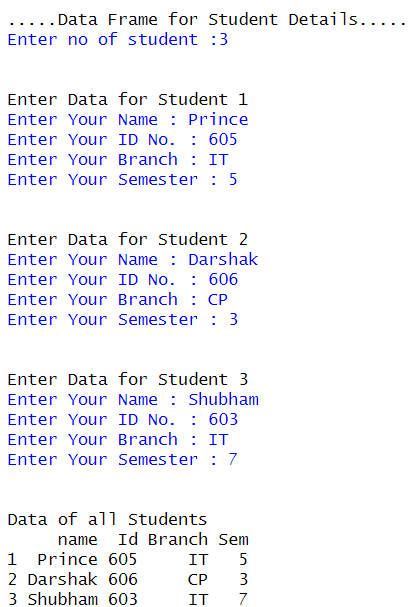
Sem = Semesters

)

cat("\n\nData of all Students\n")

print(Data\_Frame)

**Output:**

****

Practical 10

**Aim:** Write a R program to create an array of two 3x3 matrices each with 3 rows and 3 columns from two vectors. (Vector 1: 1,2,3,4   Vector 2:20,21,22,23,24,25).

**Code:**

cat("\n\n.....Matrix 3 X 3 with Array.....\n")

vector1 <- c(1, 2, 3, 4)

vector2 <- c(20, 21, 22, 23, 24, 25)

cat("\n\nVector 1 :", vector1)

cat("\nVector 2 :", vector2)

column.names <- c("COL1","COL2","COL3")

row.names <- c("ROW1","ROW2","ROW3")

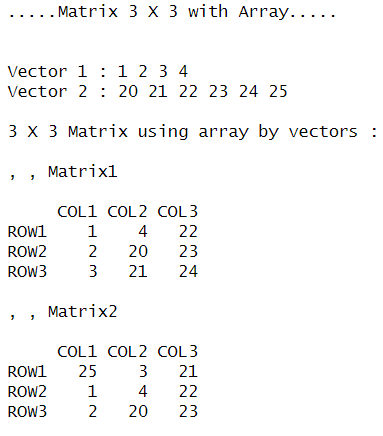
matrix.names <- c("Matrix1","Matrix2")

arr <- array(c(vector1, vector2), dim = c(3, 3, 2), dimnames = list(row.names, column.names, matrix.names))

cat("\n\n3 X 3 Matrix using array by vectors :\n\n")

print(arr)

**Output:**

****