**PROBABILITY AND STATISTICS PMA-303**

**[LAB ASSIGNMENT]**



|  |  |
| --- | --- |
| **SUBMITTED BY: - HRITHIK GUPTA** | **SUBMITTED TO: - DR. RAJNISH KUMAR RAI** |
| **BRANCH: - MSC MATHS AND COMPUTING** | **ROLL NUMBER: - 302303005** |

**Thapar Institute of Engineering & Technology, Patiala**

**Department of Mathematics**

**LAB Experiment 1: Basics of R programming**

**Q1. Write an R program to generate a sequence of numbers from 1 to 20.**

CODE:

numbers<-1:20

print(numbers)

OUTPUT:

> # Ques-1: Write an R program to generate a sequence of numbers from 1 to 20.> numbers<-1:20> print(numbers) [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

**Q2. Create a vector c = [10, 20, 30, 40, 50, 60] and write a program which returns the**

**maximum and minimum of this vector.**

CODE:

c<-c(10,20,30,40,50,60)

c\_maximum<-max(c)

c\_mininmum<-min(c)

cat("Maximum value:",c\_maximum," and Minimum value:",c\_mininmum)

OUTPUT:

> # Ques-2: Create a vector c = [10, 20, 30, 40, 50, 60] and write a program which returns the > # maximum and minimum of this vector. > c<-c(10,20,30,40,50,60)> c\_maximum<-max(c)> c\_mininmum<-min(c)> cat("Maximum value:",c\_maximum," and Minimum value:",c\_mininmum)Maximum value: 60 and Minimum value: 10

**Q3. Write an R program to plot a simple line graph of the sin function.**

CODE:

x<-seq(0,2\*pi,length.out=100)

y<-sin(x)

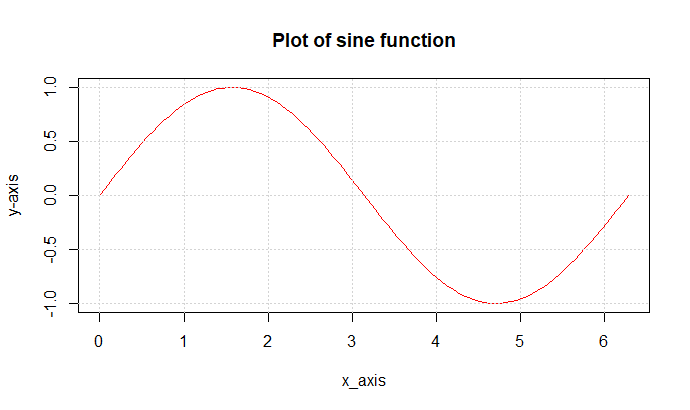
plot(x,y,type='l',col='red',lwd=1,

main="Plot of sine function",

xlab="x\_axis",ylab="y-axis")

grid()

OUTPUT:



**Q4. Write a program in R to find factorial of a number by taking input from user. Please print error message if the input number is negative.**

CODE:

factorial<-function(n){

if(n==0)

{

return(1)

}else{

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

}

}

input\_number<-as.integer(readline(prompt = "Enter a number:"))

if (input\_number<0)

{

print("Give a valid number")

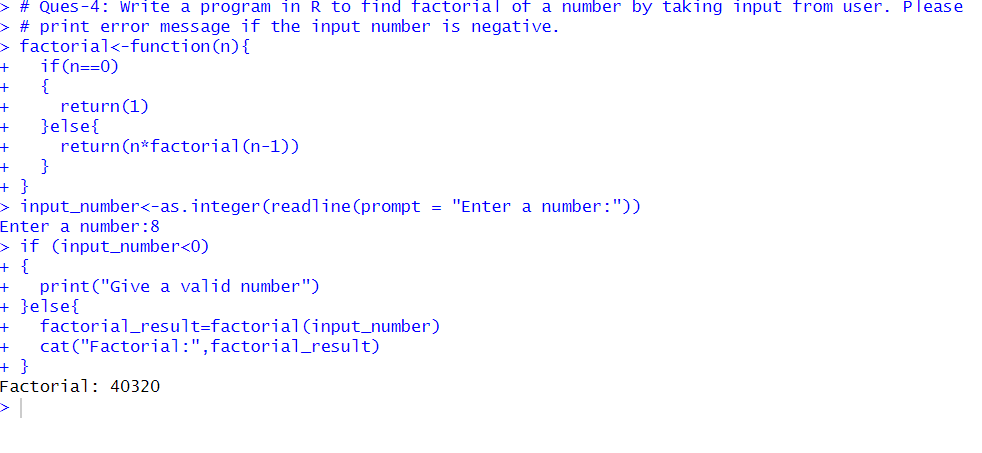
}else{

factorial\_result=factorial(input\_number)

cat("Factorial:",factorial\_result)

}

OUTPUT:



**Q5. Write an R program to calculate the mean of a given numeric vector.**

CODE:

vec<-c(10,20,30,40,50)

result<-mean(vec)

cat("Result:",result)

OUTPUT:  
> # Ques-5: Write an R program to calculate the mean of a given numeric vector.> vec<-c(10,20,30,40,50)>

result<-mean(vec)> cat("Result:",result)Result: 30

**Q6. Write a program to write first n terms of a Fibonacci sequence. You may take n as an input from the user.**

CODE:

fibonacci <- function(n) {

if (n == 0) {

return(0) # Fibonacci(0) is 0

} else if (n == 1) {

return(1) # Fibonacci(1) is 1

} else {

return(fibonacci(n - 1) + fibonacci(n - 2)) # Recursive case

}

}

# Take input from user

number <- as.integer(readline(prompt = "Enter a number: "))

# Check for valid input

if (is.na(number) || number < 0) {

print("Please enter a valid non-negative integer.") # Error message for invalid input

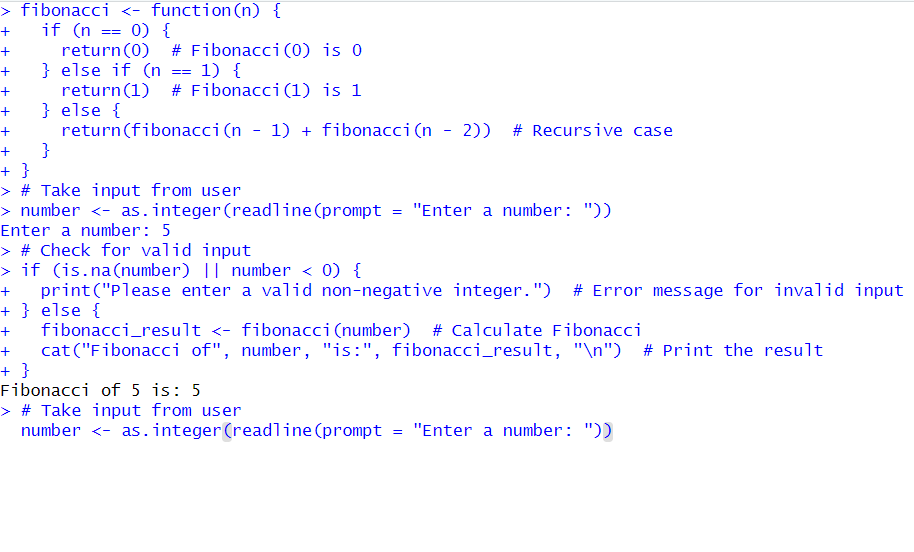
} else {

fibonacci\_result <- fibonacci(number) # Calculate Fibonacci

cat("Fibonacci of", number, "is:", fibonacci\_result, "\n") # Print the result

}

OUTPUT:



**Q7. Write an R program to make a simple calculator which can add, subtract, multiply and divide.**

CODE:

n1<-as.integer(readline(prompt="Enter the first number:"))

oper<-readline(prompt="Enter the operator(+,-,/,\*)")

n2<-as.integer(readline(prompt="Enter the second number:"))

result <- switch(oper,

"+" = n1 + n2,

"-" = n1 - n2,

"\*" = n1 \* n2,

"/" = {

if (n2 == 0) {

return("Error: Division by zero is not allowed.")

} else {

n1 / n2

}

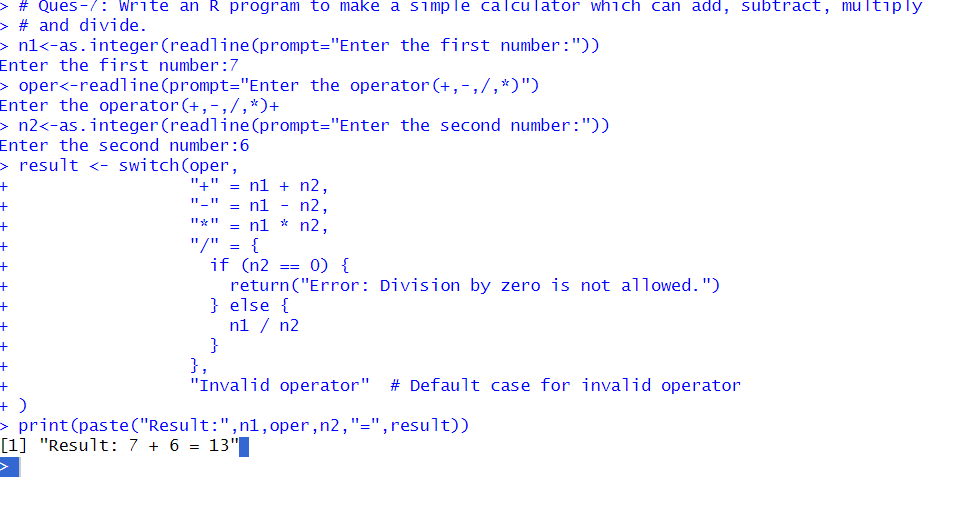
},

"Invalid operator" # Default case for invalid operator

)

print(paste("Result:",n1,oper,n2,"=",result))

OUTPUT:



**Q8. Explore plot, pie, barplot etc. (the plotting options) which are built-in functions in R.**

CODE:

x<-seq(0,2\*pi,length.out=100)

y<-sin(x)

plot(x,y,type="l",col="blue",lwd=3,main="Sin Graph",xlab="x-axis",ylab="y-axis")

data<-c(10,30,40,9,11)

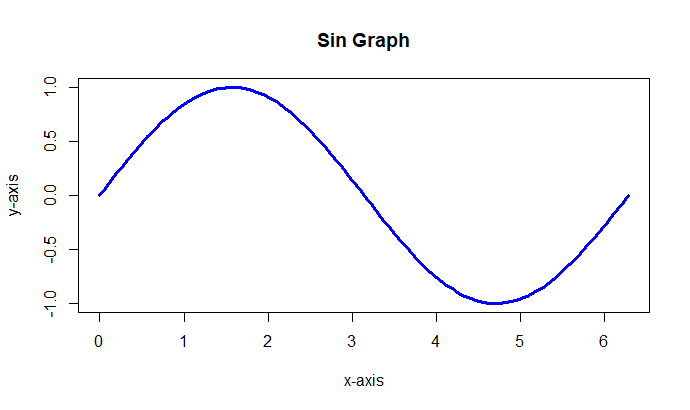
label<-c("A","B","C","D","E")

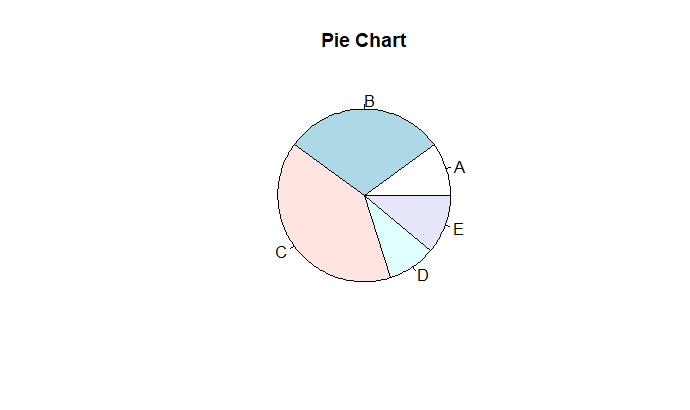
pie(data,labels = label,main="Pie Chart")

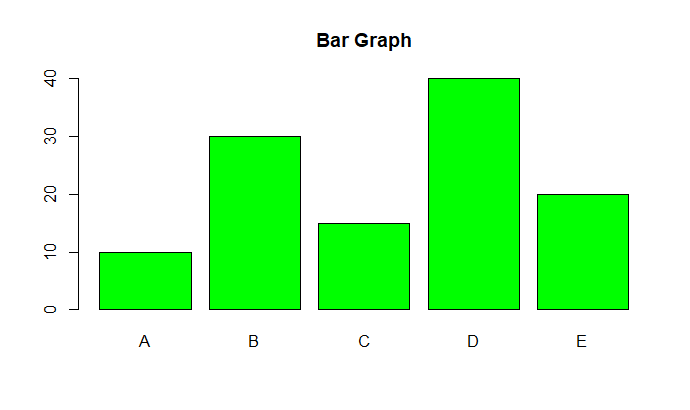
value<-c(10,30,15,40,20)

type<-c("A","B","C","D","E")

barplot(value,names.arg=type,main="Bar Graph",col="green")

OUTPUT:





**Thapar Institute of Engineering & Technology, Patiala**

**Department of Mathematics**

**LAB Experiment 2: Sample space, Total probability and Bayes theorem**

**Q1. The iris data-set is a built-in data-set in R that contains measurements on 4 different attributes (in centimeters) for 150 flowers from 3 different species. Load this data-set and do the following:**

**(a) Print first few rows of this data-set.**

**(b) Find the structure of this data-set.**

**(c) Find the range of the data regarding the sepal length of flowers.**

**(d) Find the mean of the sepal length.**

**(e) Find the median of the sepal length.**

**(f) Find the first and the third quartiles and hence the interquartile range.**

**(g) Find the standard deviation and variance.**

**(h) Try doing the above exercises for sepal.width, petal.length and petal.width.**

**(i) Use the built-in function summary on the data-set Iris.**

CODE:

data(iris)

head(iris)

str(iris)

# Calculate the range of Sepal.Length

sepal\_length\_range <- range(iris$Sepal.Length)

# Display the range

print(sepal\_length\_range)

# Calculate the mean of Sepal.Length

mean\_sepal\_length <- mean(iris$Sepal.Length)

mean\_sepal\_length

# Calculate the median of Sepal.Length

median\_sepal\_length <- median(iris$Sepal.Length)

median\_sepal\_length

q1<-quantile(iris$Sepal.Length,0.25)

print(q1)

q3<-quantile(iris$Sepal.Length,0.75)

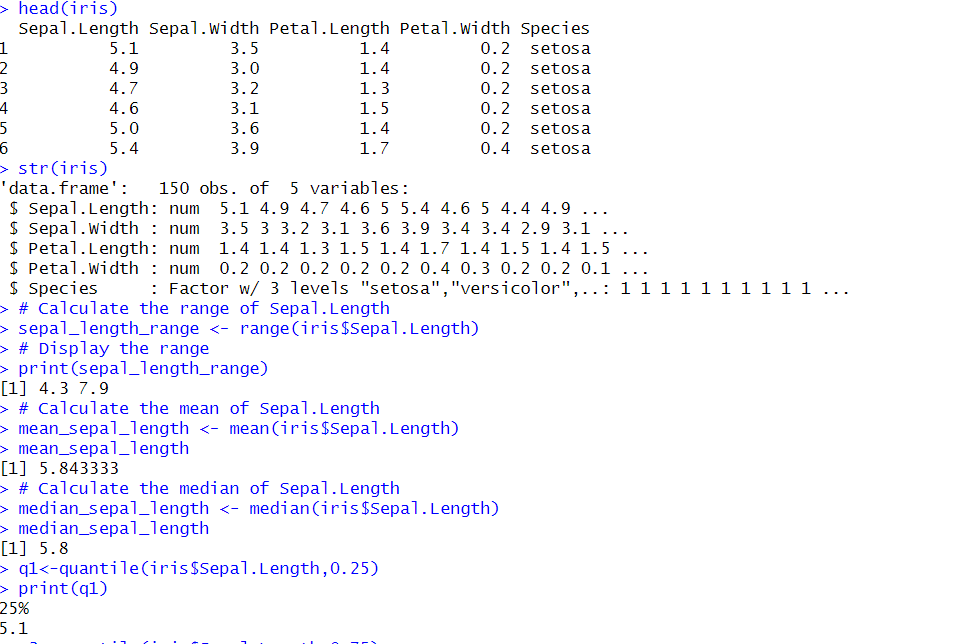
print(q3)

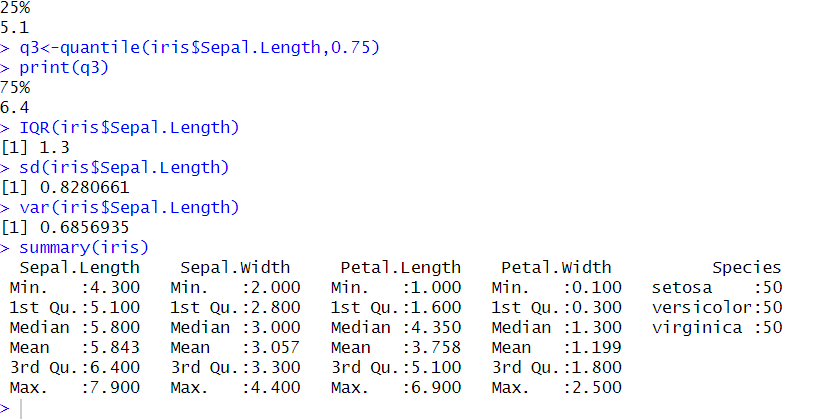
IQR(iris$Sepal.Length)

sd(iris$Sepal.Length)

var(iris$Sepal.Length)

summary(iris)

OUTPUT:



**Q2. R does not have a standard in-built function to calculate mode. So we create a user function to calculate mode of a data set in R. This function takes the vector as input and gives the mode value as output.**

CODE:

calculate\_mode <- function(x) {

# Remove NA values

x <- na.omit(x)

# Get the unique values and their frequencies

unique\_values <- unique(x)

frequencies <- table(x)#table uniquely count the value of vector

# Find the maximum frequency

max\_freq <- max(frequencies)

# Get the values that have the maximum frequency (mode)

modes <- unique\_values[frequencies == max\_freq]

return(modes)

}

# Example usage

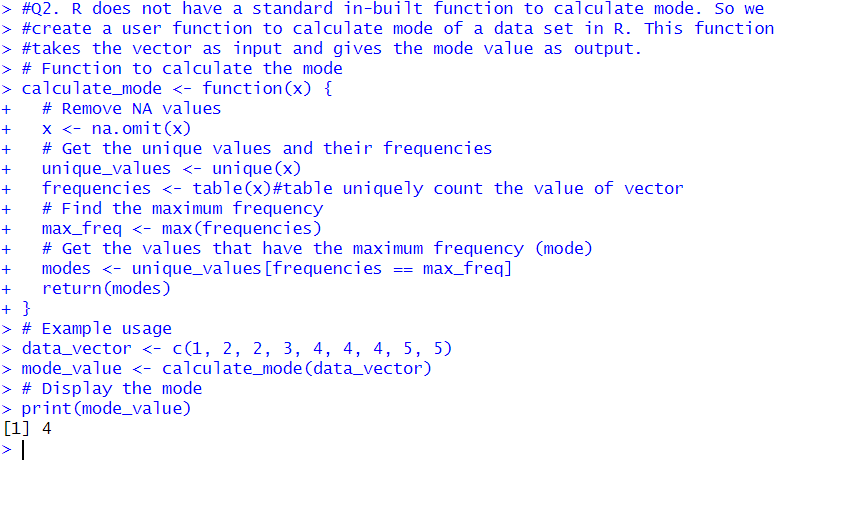
data\_vector <- c(1, 2, 2, 3, 4, 4, 4, 5, 5)

mode\_value <- calculate\_mode(data\_vector)

# Display the mode

print(mode\_value)

OUTPUT:



**Q3. A room has M people, and each has an equal chance of being born on any of the 365 days of the year. (For simplicity, we will ignore leap years). What is the probability that two people in the room have the same birthday ?**

**(a) Use an R simulation to estimate this for various M.**

**(b) Find the smallest value of M for which the probability of a match is greater**

**than 0.5.**

CODE:

M<-50 #out of 5000 we consider 50 persons

N<-5000 #number of simulation total no of persons exhaustive cases

fav<-0 #favourable cases

for(val in 1:N){

a<-as.integer(any(duplicated(sample(365,M,replace=TRUE))))

fav<-fav+a

}

prob<-fav/N

print(prob)

M <- 2

prob <- 0 # Initialize probability

while (TRUE) {

prod <- 1

for (i in 0:(M - 1)) {

prod <- prod \* (1 - i / 365)

}

prob <- 1 - prod # Probability of at least one match

if (prob > 0.5) {

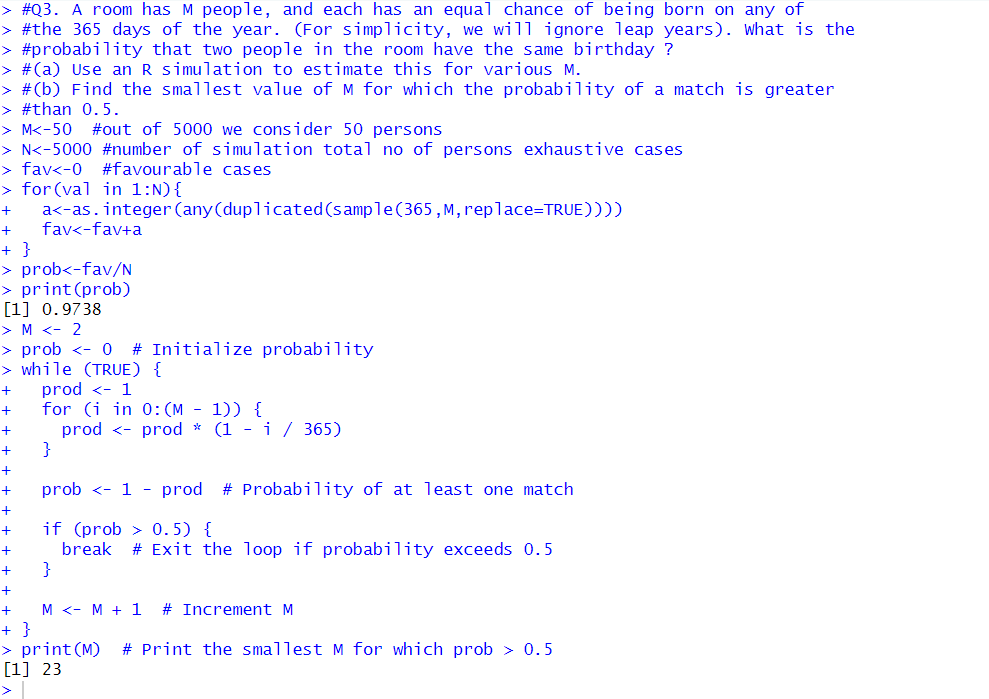
break # Exit the loop if probability exceeds 0.5

}

M <- M + 1 # Increment M

}

print(M) # Print the smallest M for which prob > 0.5

OUTPUT:

**Q4. Write an R function for computing conditional probability. Call this function to do the following problem: suppose the probability of the weather being cloudy is 40%. Also suppose the prob- ability of rain on a given day is 20% and that the probability of clouds on a rainy day is 85%. If it’s cloudy outside on a given day, what is the probability that it will rain that day?**

CODE:

cond\_prob<-function(pR,pCR,pC){ #pR=rain, pCR= rain under cloudy, pC=cloudy

pRC<-(pR\*pCR)/pC

return(pRC)

}

#Define probability

pRain<-0.2

pCloudy<-0.4

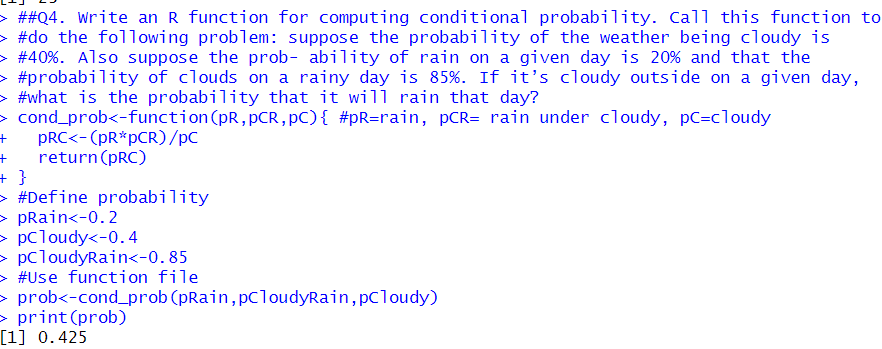
pCloudyRain<-0.85

#Use function file

prob<-cond\_prob(pRain,pCloudyRain,pCloudy)

print(prob)

OUTPUT:



**Q5. Write an R function for computing conditional probability. Call this function to do the following problem: Three urns I, II and III contain 8 red, 4 white; 6 red, 6 white; and 5 red, 7 white balls, respectively. If a ball is drawn at random and found**

**to be red, what is the probability that it is a drawn from (i) urn I, (ii) urn III?**

CODE:

cond\_prob<-function(PA, PE\_given\_A, PE) {

PA\_given\_E <- (PA\*PE\_given\_A)/PE

return(PA\_given\_E)

}

# Problem setup

# Urn probabilities (assuming each urn is equally

#likely to be chosen)

P\_urn\_I<-1/3

P\_urn\_II<-1/3

P\_urn\_III<-1/3

# Probability of drawing a red ball given the urn

P\_red\_given\_urn\_I<-8/(8 + 4)

P\_red\_given\_urn\_II<-6/(6 + 6)

P\_red\_given\_urn\_III<-5/(5 + 7)

# Total probability of drawing a red ball

P\_red <- (P\_urn\_I\*P\_red\_given\_urn\_I) +

(P\_urn\_II\*P\_red\_given\_urn\_II) +

(P\_urn\_III\*P\_red\_given\_urn\_III)

# Calculate the conditional probabilities

P\_urn\_I\_given\_red <- cond\_prob(P\_urn\_I, P\_red\_given\_urn\_I, P\_red)

P\_urn\_III\_given\_red <- cond\_prob(P\_urn\_III, P\_red\_given\_urn\_III, P\_red)

# Print the results

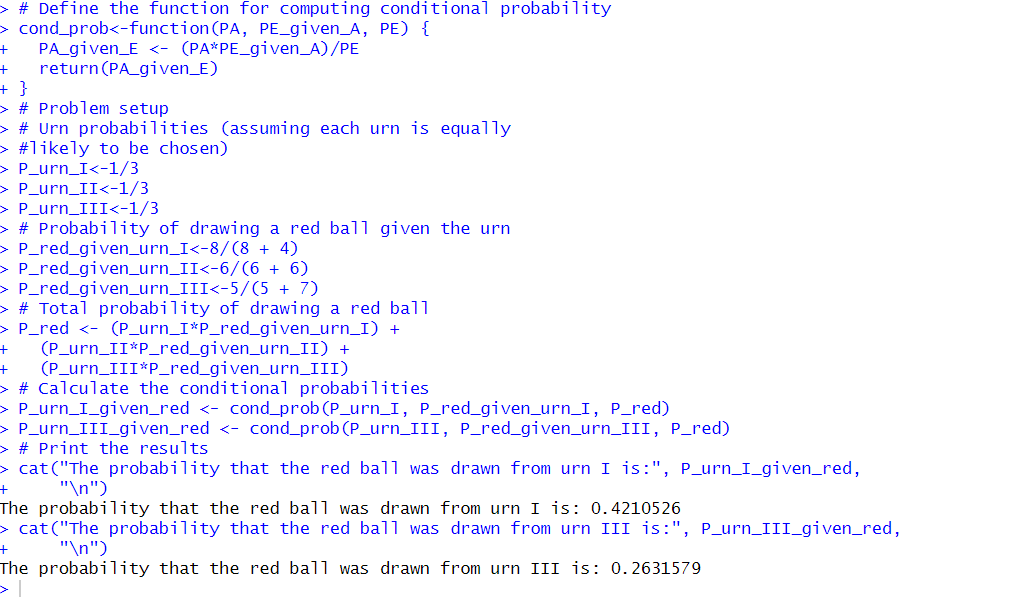
cat("The probability that the red ball was drawn from urn I is:", P\_urn\_I\_given\_red,

"\n")

cat("The probability that the red ball was drawn from urn III is:", P\_urn\_III\_given\_red,

"\n")

OUTPUT:



**Q6. Write an R function for computing conditional probability. Call this function to do the following problem: Companies A, B and C produces cars. The production capacity of company A is twice that of B while company B and C produces same**

**number of cars in a given period. It is known that 2% of A, 3% of B and 4% of C are defective. All the cars produced are put into one showroom and then one car is chosen at random.**

**(a) Find the probability that the car is defective.**

**(b) Suppose a car chosen is defective, what is the probability that this is produced**

**by company A?**

CODE:

cond\_prob<-function(P\_A, PE\_given\_A, PE) {

PA\_given\_E <- (P\_A\*PE\_given\_A)/PE

return(PA\_given\_E)

}

P\_A<-2/4

P\_B<-1/4

P\_C<-1/4

P\_defective\_given\_A<-0.02

P\_defective\_given\_B<-0.03

P\_defective\_given\_C<-0.04

p\_defective<-(P\_A\*P\_defective\_given\_A)+

(P\_B\*P\_defective\_given\_B)+

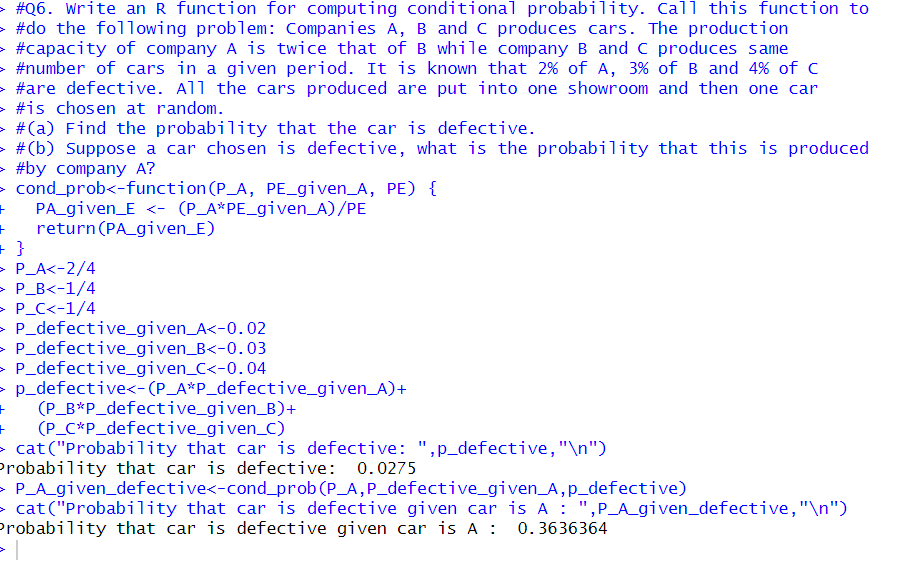
(P\_C\*P\_defective\_given\_C)

cat("Probability that car is defective: ",p\_defective,"\n")

P\_A\_given\_defective<-cond\_prob(P\_A,P\_defective\_given\_A,p\_defective)

cat("Probability that car is defective given car is A : ",P\_A\_given\_defective,"\n")

OUTPUT:



**Q7. Write an R function for computing conditional probability. Call this function to do the following problem: The LED bulbs producing factories A, B and C supply LED bulbs to the market in the ratio 2:3:5. It is found that 1% of the items produced**

**in factory A, 2% of the items produced in factory B and 3% of the items produced in factory C are defective. If a bulb is selected at random from the market what is the probability that it is a defective one? Also, if a randomly selected bulb is found to be defective then find the probability that it was produced by factory (i) A, (ii) B, (iii) C?**

CODE:

cond\_prob<-function(P\_A, PE\_given\_A, PE) {

PA\_given\_E <- (P\_A\*PE\_given\_A)/PE

return(PA\_given\_E)

}

P\_A<-2/10

P\_B<-3/10

P\_C<-5/10

P\_defective\_given\_A<-0.01

P\_defective\_given\_B<-0.02

P\_defective\_given\_C<-0.03

p\_defective<-(P\_A\*P\_defective\_given\_A)+

(P\_B\*P\_defective\_given\_B)+

(P\_C\*P\_defective\_given\_C)

cat("Probability that car is defective: ",p\_defective,"\n")

P\_A\_given\_defective<-cond\_prob(P\_A,P\_defective\_given\_A,p\_defective)

cat("Probability that car is defective given car is A : ",P\_A\_given\_defective,"\n")

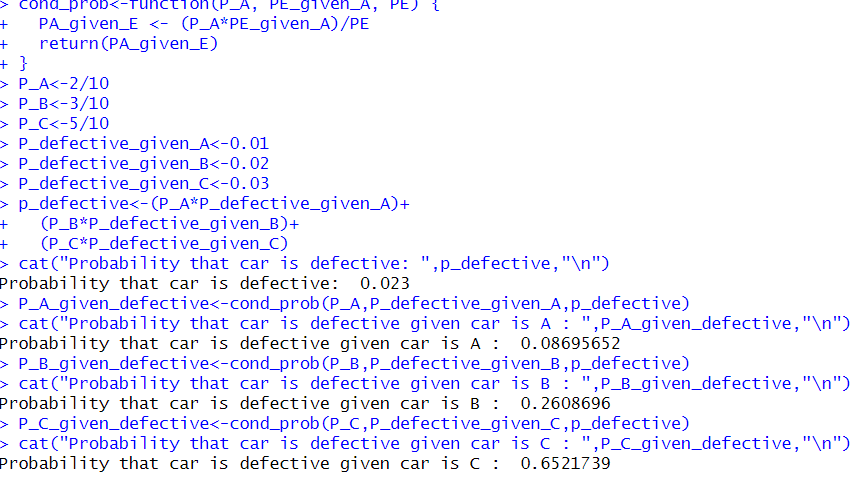
P\_B\_given\_defective<-cond\_prob(P\_B,P\_defective\_given\_B,p\_defective)

cat("Probability that car is defective given car is B : ",P\_B\_given\_defective,"\n")

P\_C\_given\_defective<-cond\_prob(P\_C,P\_defective\_given\_C,p\_defective)

cat("Probability that car is defective given car is C : ",P\_C\_given\_defective,"\n")

OUTPUT:



**Thapar Institute of Engineering & Technology, Patiala**

**Department of Mathematics**

**LAB Experiment 3: Mathematical Expectation, Moments and Functions of Random Variables**

**Q1. The probability distribution of X, the number of imperfections per 10 meters of a synthetic fabric in continuous rolls of uniform width, is given as**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **X** | **0** | **1** | **2** | **3** | **4** |
| **P(X = x)** | **0.41** | **0.37** | **0.16** | **0.05** | **0.01** |

**Use R-code to find the average number of imperfections per 10 meters of this fabric. (Try function sum(), weight.mean(), c(a %\*% b) to find expected value/mean. Further use it to find E(X2), E(2X + 3), V ar(2X − 3).**

CODE:

X<-c(0,1,2,3,4)

P\_X<-c(0.41,0.37,0.16,0.05,0.01)

#Calculate the expected value of E(X)

E\_X<-sum(X\*P\_X)

cat("E(X):",E\_X,"/n")

#OR

Exp\_Val<-weighted.mean(X,P\_X)

print(Exp\_Val)

#OR

Exp\_Val<-c(X %\*% P\_X) # %\*% can be used for calculating sum of

# element wise product between two vectors

print(Exp\_Val)

#Calculate E(X^2)

E\_X2<-sum((X^2)\*P\_X)

cat("E(X^2): ",E\_X2,"\n")

#Calculte E(2X + 3)

E\_2X\_plus\_3<-2\*E\_X+3

cat("E(2X+3): ",E\_2X\_plus\_3,"\n")

#Calculate the variance Var(X)

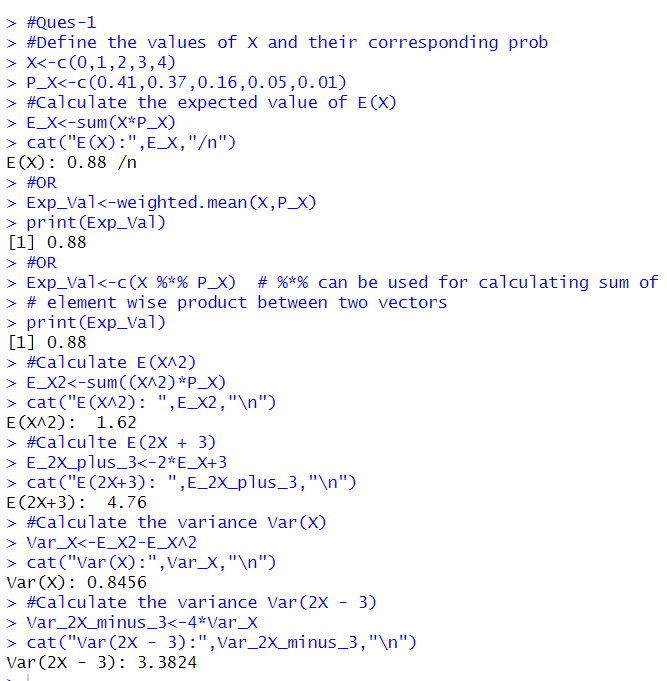
Var\_X<-E\_X2-E\_X^2

cat("Var(X):",Var\_X,"\n")

#Calculate the variance Var(2X - 3)

Var\_2X\_minus\_3<-4\*Var\_X

cat("Var(2X - 3):",Var\_2X\_minus\_3,"\n")

OUTPUT:

**Q2. A bookstore purchases three copies of a book at $12 each and sells them for $24 each. Unsold copies are returned for $4 each. Let X = number of copies sold and Y = Net revenue. If the probability mass function of X is**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **X** | **0** | **1** | **2** | **3** |
| **P(X=x)** | **0.1** | **0.2** | **0.2** | **0.5** |

**Use R-code to find the expected value and variance of Y**

CODE:

#Revenue Y=h(x)= 24X+4(3-X)-36 = 20X-24

x<-c(0,1,2,3)

probx<-c(0.1,0.2,0.2,0.5)

y<-20\*x-24

proby<-probx #E(phi(x))=sum phi(x) P(X=x)=sum(10X-12) P(X=x)

E\_Y<-sum(y\*proby)

cat("E(Y): ",E\_Y,"\n")

#Calculate the expected value of Y^2

E\_Y2<-sum((y^2)\*proby)

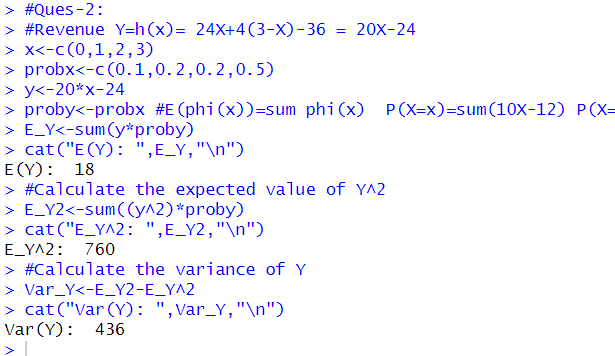
cat("E\_Y^2: ",E\_Y2,"\n")

#Calculate the variance of Y

Var\_Y<-E\_Y2-E\_Y^2

cat("Var(Y): ",Var\_Y,"\n")

OUTPUT:



**Q3. Following is the cumulative probability distribution function of a discrete random**

**variable X:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **X** | **-3** | **-1** | **0** | **1** | **2** | **3** | **5** | **8** |
| **F(x)** | **0.10** | **.30** | **.45** | **.5** | **.75** | **.90** | **.95** | **1** |

**(a) Write R code to find the probability mass function of X.**

**(b) Write R code to find P(X = Even), P(1 ≤ X ≤ 8) and P(X ≥ 3|X > 0).**

CODE:

#QUES-3:

x<-c(-3,-1,0,1,2,3,5,8)

cdf\_x<-c(0.10,0.30,0.45,0.5,0.75,0.90,0.95,1.0)

i=0

while(i<8){

pdf\_x<-c(cdf\_x[i+1]-cdf\_x[i])

i=i+1

print(pdf\_x)

}

for(j in 1:8){

if(x[j]%%2==0){

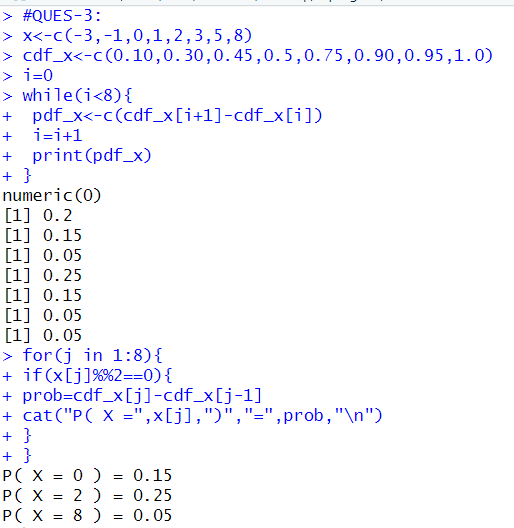
prob=cdf\_x[j]-cdf\_x[j-1]

cat("P( X =",x[j],")","=",prob,"\n")

}

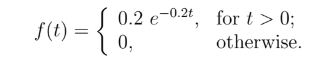
}

OUTPUT:



**Q4. The time T, in days, required for the completion of a contracted project is a random**

**variable with probability density function**

****

**Use R-code to find the expected value and variance of T & 2T − 3. Use function integrate() to find the expected value and variance of continuous random variable T & 2T − 3.**

CODE:

f1<-function(t){0.2\*t\*exp(-0.2\*t)

}

Exp\_t<-integrate(f1,lower=0,upper=Inf)$value

print(Exp\_t)

#OR:

f\_T<-function(t){

ifelse(t>0,0.2\*t\*exp(-0.2\*t),0)

}

Exp\_t\_f\_T<-integrate(function(t) t\*f\_T(t),lower=0,upper=Inf)$value

print(Exp\_t\_f\_T)

#E(2t-3)

Exp\_2t\_3<-2\*Exp\_t-3

print(Exp\_2t\_3)

#E(t^2)

f2<-function(t)((0.2\*t\*t)\*exp(-0.2\*t))

Exp\_t\_2<-integrate(f2,lower=0,upper=Inf)$value

print(Exp\_t\_2)

#var(t)

Var\_T<-Exp\_t\_2-Exp\_t^2

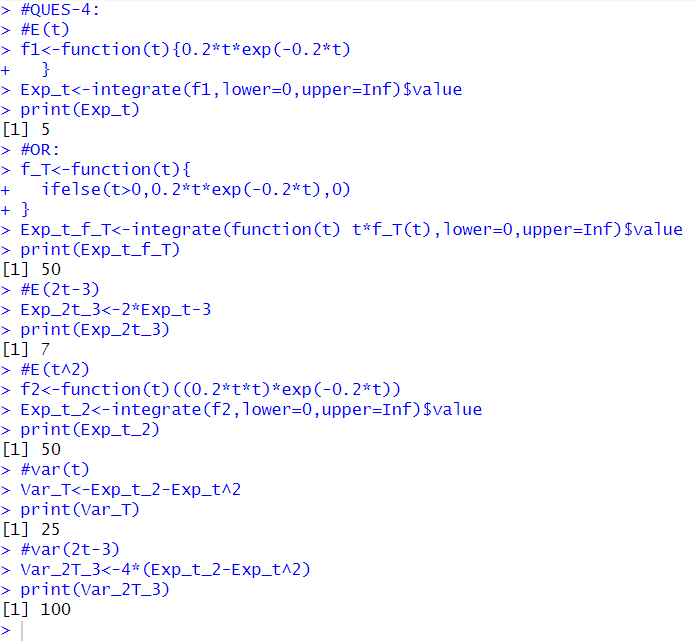
print(Var\_T)

#var(2t-3)

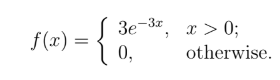
Var\_2T\_3<-4\*(Exp\_t\_2-Exp\_t^2)

print(Var\_2T\_3)

OUTPUT:



**Q5. Write R-code to find the first and second moments about the origin of the random variable X with probability density function**

****

**Further use the results to find mean and variance of X and 2X + 3.**

CODE:

f1<-function(x)(3\*x\*exp(-3\*x))

Exp\_t<-integrate(f1,0,Inf)$value

print(Exp\_t)

#Mean(2x+3)

Exp\_2x\_plus\_3<-2\*Exp\_t+3

print(Exp\_2x\_plus\_3)

#Second moment of X

f2<-function(x)(3\*x\*x\*exp(-3\*x))

Exp\_t\_2<-integrate(f2,0,Inf)$value

print(Exp\_t\_2)

#VAR(X)

Var\_x<-Exp\_t\_2-Exp\_t^2

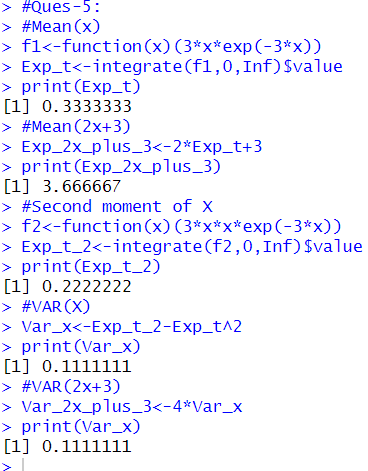
print(Var\_x)

#VAR(2x+3)

Var\_2x\_plus\_3<-4\*Var\_x

print(Var\_x)

OUTPUT:



**Q7. Two unbiased dice are thrown. Find the expected value and the variance of the sum of number of points on both.**

CODE:

sums<-2:12

#prob for each sum

probs<-c(1,2,3,4,5,6,5,4,3,2,1)/36

#expected value

E\_X<-sum(sums\*probs)

print(E\_X)

#E(x2)

E\_X2<-sum(sums^2\*probs)

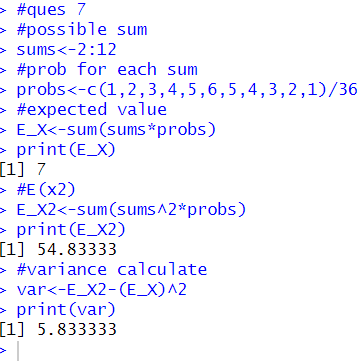
print(E\_X2)

#variance calculate

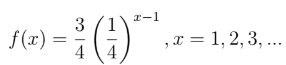
var<-E\_X2-(E\_X)^2

print(var)

OUTPUT:



**Q8. Let X be a geometric random variable with probability distribution**

****

**Write a function to find the probability distribution of the random variable Y = X2 and find probability of Y for X = 3. Further, use it to find the expected value and variance of Y for X = 1, 2, 3, 4, 5.**

CODE:

func<-function(y){

(3/4)\*(1/4)^(sqrt(y)-1)

}

x<-as.integer(readline(prompt="Enter value of X : "))

y<-x^2

proby<-func(y)

print(proby)

#to find exp value and variance

x<-c(1,2,3,4,5)

y<-x^2

proby<-func(y)

print(proby)

#exp Y

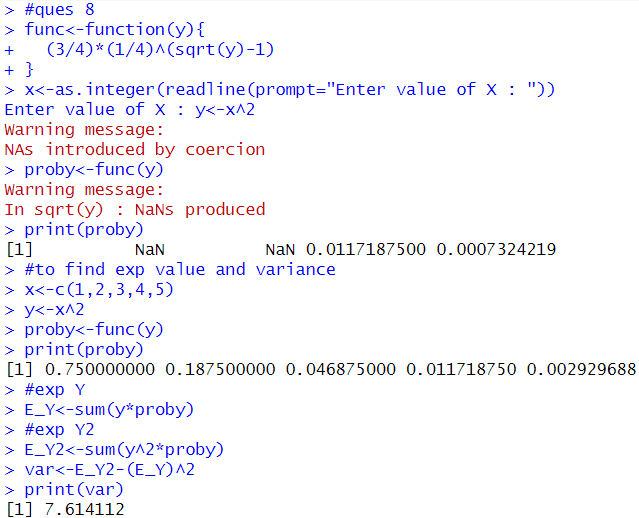
E\_Y<-sum(y\*proby)

#exp Y2

E\_Y2<-sum(y^2\*proby)

var<-E\_Y2-(E\_Y)^2

print(var)

OUTPUT:  


**Thapar Institute of Engineering & Technology, Patiala**

**Department of Mathematics**

**LAB Experiment 4: Discrete probability distribution**

**Q1. Roll 12 dice simultaneously, and let X denotes the number of 6’s that appear. (Try using the function pbinom & dbinom; If we set S = get a 6 on one roll, P(S) = 1/6**

**and the rolls constitute Bernoulli trials; thus X∼ binom(size=12, Prob=1/6). Write R-code to find probability of getting**

**(a) 8, 6’s i.e., P(X = 8),**

**(b) at most 7, 6’s i.e., P(X ≤ 7),**

**(c) more than 6, 6’s i.e., P(X > 6),**

**(d) getting 7, 8 or 9, 6’s, i.e., P(7 ≤ X ≤ 9),**

**(e) getting 7 or 8, 6’s, i.e., P(7 ≤ X < 9).**

CODE:

bino\_pmf<-function(x,n,p){

P\_X<-choose(n,x)\*p^(x)\*(1-p)^(n-x)

return(P\_X)

}

#a=8

n<-12

p<-1/6

P\_8<-bino\_pmf(8,n,p)

print(P\_8)

P\_8<-dbinom(8,n,p)

print(P\_8)

############## part b ######################

n<-12

p<-1/6

P\_X\_Lesser\_than\_7<-sum(bino\_pmf(0:7,n,p))

print(P\_X\_Lesser\_than\_7)

#c part

p\_x\_greater\_6<-1-sum(bino\_pmf(0:6,n,p))

print(p\_x\_greater\_6)

p\_x\_greater\_6<-pbinom(6,size=12,prob=1/6,lower.tail=F)

print(p\_x\_greater\_6)

####### part d ######

p<- pbinom(9,size=12,prob=1/6) - pbinom(6,size=12,prob=1/6)

print(p)

p<-sum(bino\_pmf(0:9,12,1/6))-sum(bino\_pmf(0:6,12,1/6))

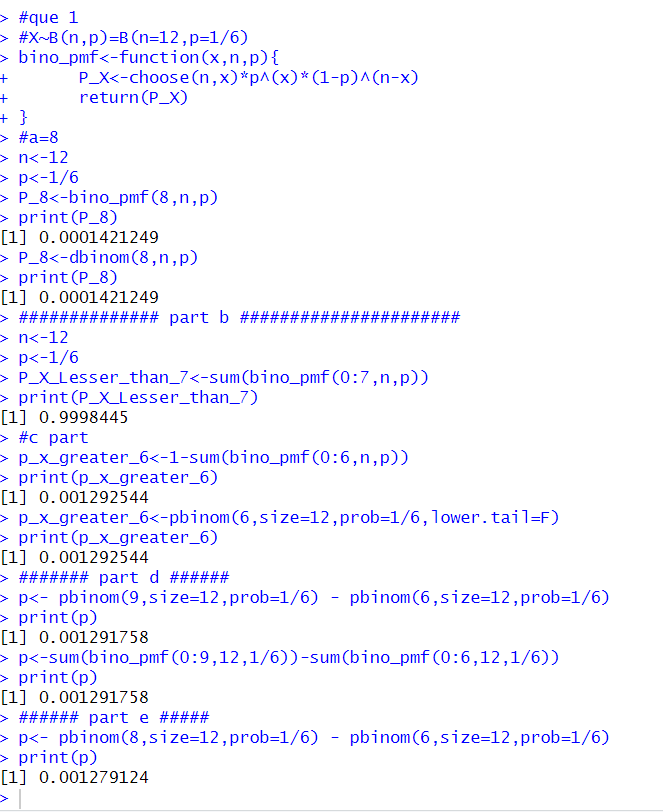
print(p)

###### part e #####

p<- pbinom(8,size=12,prob=1/6) - pbinom(6,size=12,prob=1/6)

print(p)

OUTPUT:



**Q2. A recent national study showed that approximately 44.7% of college students have used Wikipedia as a source in at least one of their term papers. Let X equal the**

**number of students in a random sample of size n = 31 who have used Wikipedia as**

**a source. Write R-code to find**

**(a) How is X distributed?**

**(b) Sketch the probability mass function.**

**(c) Sketch the cumulative distribution function.**

**(d) Find mean, variance and standard deviation of X.**

CODE:

n<-31

p<-0.447

x<-0:n

pmf<-dbinom(x,n,p)

print(pmf)

plot(x,pmf,xlab='x-axis',ylab='y-axis',

main='Pmf for binomial distribution')

cdf<-pbinom(x,n,p)

print(cdf)

plot(x,cdf,xlab='x-axis',ylab='y-axis',

main='cdf for binomial distribution')

E\_X<-sum(x\*pmf)

print(E\_X)

E\_X\_2<-sum(x\*x\*pmf)

print(E\_X\_2)

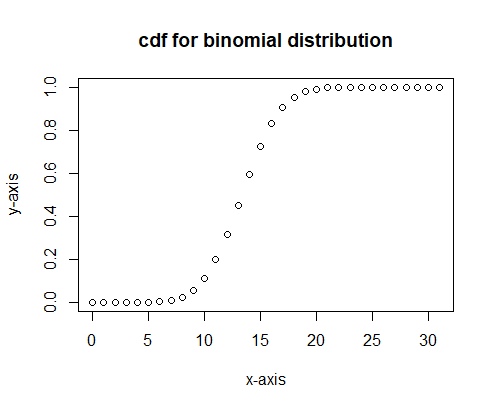
V\_X<-E\_X\_2-(E\_X)^2

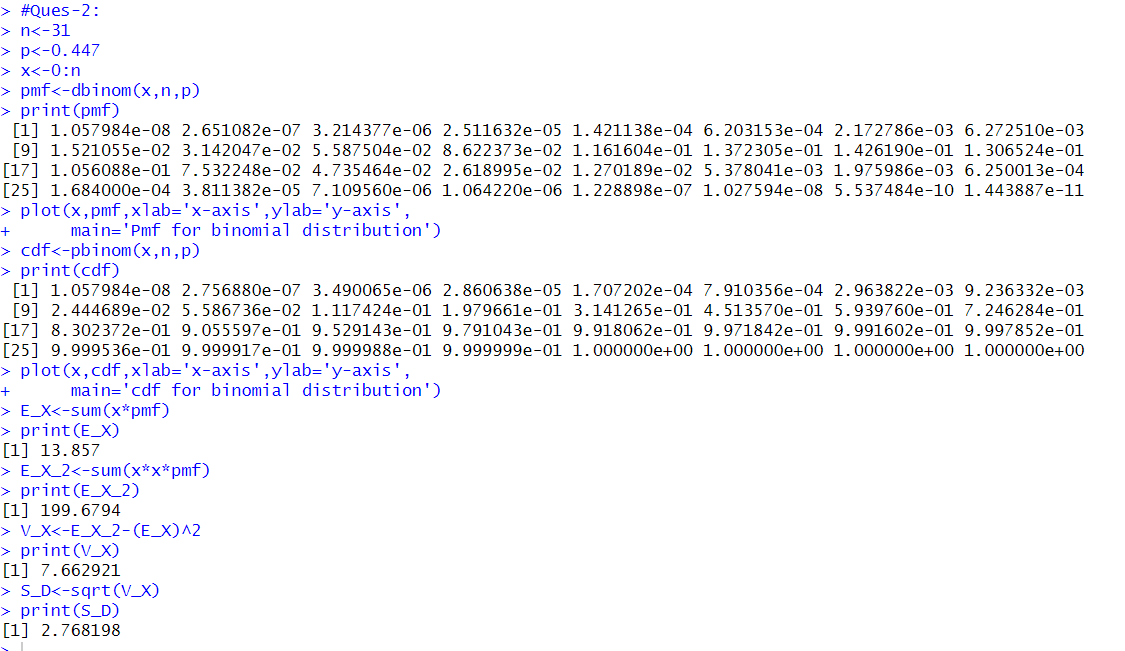
print(V\_X)

S\_D<-sqrt(V\_X)

print(S\_D)

OUTPUT:





**Q3. If the probability that an individual suffer a bad reaction from an injection of a given serum is 0.001. Write R-code to determine the probability that out of 2000 individuals,**

**(a) exactly 3; and**

**(b) more than two individuals will suffer a bad reaction.**

CODE:

n<-2000

p<-0.001

lambda<-n\*p

x<-3

P\_At\_X\_3<-dpois(x,lambda)

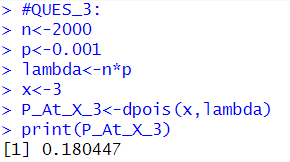
print(P\_At\_X\_3)

#P(X>2)=1-P(X<=2)=1-F(2)

P\_Atleast\_2<-1-ppois(2,lambda)

print(P\_Atleast\_2)

OUTPUT:



**Q4. A space craft has 100, 000 components. The probability of any one component**

**being defective is 2×10−5.The mission will be in danger if five or more components become defective. Write R-code to find the probability of such an event.**

CODE:

n<-100000

p<-0.00002

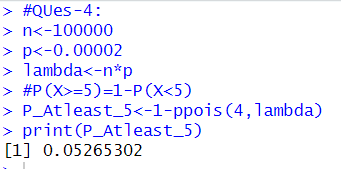
lambda<-n\*p

#P(X>=5)=1-P(X<5)

P\_Atleast\_5<-1-ppois(4,lambda)

print(P\_Atleast\_5)

OUTPUT:



**Q5. On the average, five cars arrive at a particular car wash every hour. Let X count**

**the number of cars that arrive from 10AM to 11AM, then X ∼ Poisson (λ = 5). What is probability that no car arrives during this time. Next, suppose the carwash above is in operation from 8AM to 6PM, and we let Y be the number of customers that appear in this period. Since this period covers a total of 10 hours, we get that Y ∼ Poisson (λ = 5 × 10 = 50). Write R-code to find the probability that there are between 48 and 50 customers, inclusive?**

CODE:

lambda<-5

x<-0

P\_X\_0<-dpois(x,lambda)

print(P\_X\_0)

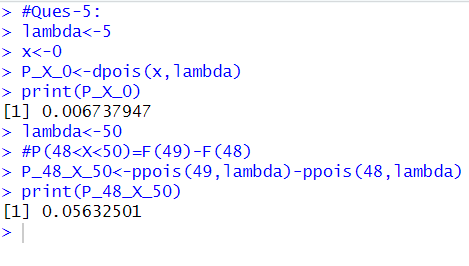
lambda<-50

#P(48<X<50)=F(49)-F(48)

P\_48\_X\_50<-ppois(49,lambda)-ppois(48,lambda)

print(P\_48\_X\_50)

OUTPUT:



**Q6. Suppose that a trainee solder shoots a target according to geometric distribution. If probability that a target hits in any shot is 0.6. Write R-code to find the probability that it takes an odd number of shots.**

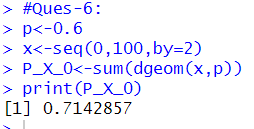
CODE:  
p<-0.6

x<-seq(0,100,by=2)

P\_X\_0<-sum(dgeom(x,p))

print(P\_X\_0)

OUTPUT:



**Q7. A boy is throwing stones at a target, Write R-code to find the probability that his 10th throw is his 5th hit, if the probability of hitting the target at any trial is 0.05.**

CODE:

p<-0.05

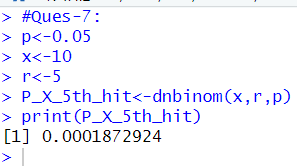
x<-10

r<-5

P\_X\_5th\_hit<-dnbinom(x,r,p)

print(P\_X\_5th\_hit)

OUTPUT:



**Thapar Institute of Engineering & Technology, Patiala**

**Department of Mathematics**

**LAB Experiment 5: Continuous probability distribution**

**Q1. If X is uniformly distributed over the interval [−2, 2], write R-code to find:**

**(i) P(X < 0)**

**(ii) P|X − 1| ≥ ½**

**using cumulative distribution function (C.D.F.) approach.**

CODE:

a<--2

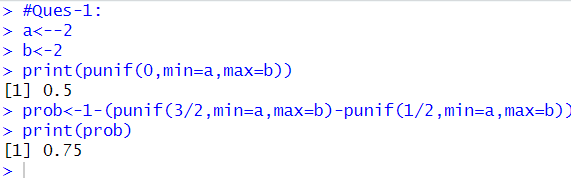
b<-2

print(punif(0,min=a,max=b))

prob<-1-(punif(3/2,min=a,max=b)-punif(1/2,min=a,max=b))

print(prob)

OUTPUT:



**Q2. Consider that X is the time (in minutes) that a person has to wait in order to take a flight. If each flight takes off each hour X ∼ U(0, 60). Write R-code to find the probability that**

**(i) waiting time is more than 35 minutes**

**(ii) waiting time lies between 10 and 25 minutes.**

CODE:

#P(X>35)=1-P(X<=35)=1-F(35)

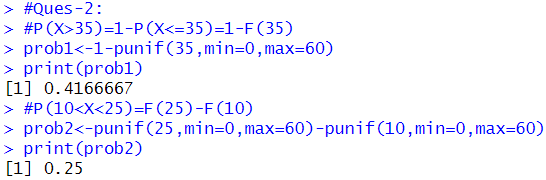
prob1<-1-punif(35,min=0,max=60)

print(prob1)

#P(10<X<25)=F(25)-F(10)

prob2<-punif

OUTPUT:



**Q3. The time (in hours) required to repair a machine is an exponential distributed**

**random variable with parameter λ = 1/2.**

**(i) Find the value of density function at x = 4.**

**(ii) Plot the graph of exponential probability distribution for 0 ≤ x ≤ 10.**

**(iii) Find the probability that a repair time takes at most 4 hours.**

**(iv) Plot the graph of cumulative exponential probabilities for 0 ≤ x ≤ 10.**

**(v) Simulate 1000 exponential distributed random numbers with λ = 1/2 and plot**

**the simulated data.**

CODE:

Exp\_pdf<-function(x,lambda){

f\_x<-lambda\*exp(-lambda\*x)

return(f\_x)

}

a<-as.integer(readline(prompt='Enter the value of a:'))

p\_x\_4<-Exp\_pdf(a,1/2)

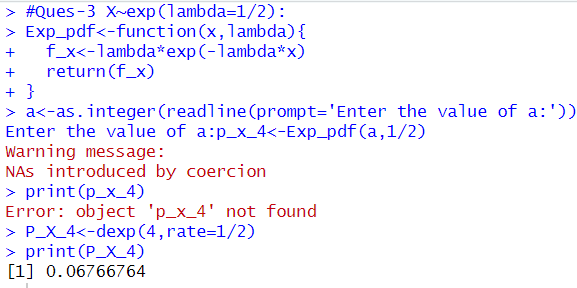
print(p\_x\_4)

#OR

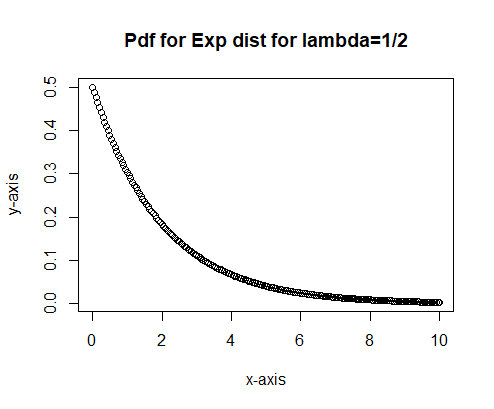
P\_X\_4<-dexp(4,rate=1/2)

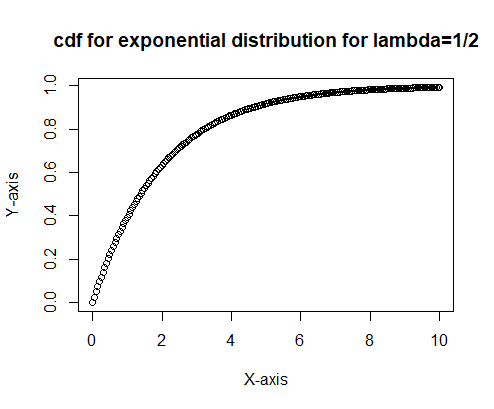
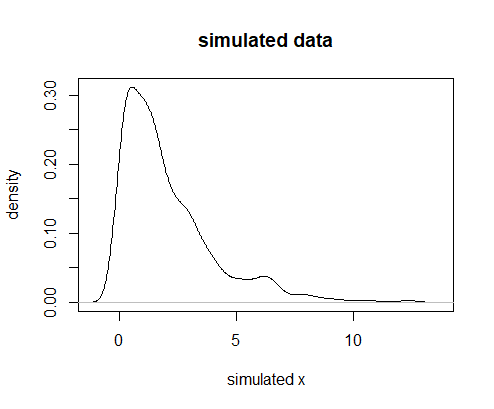
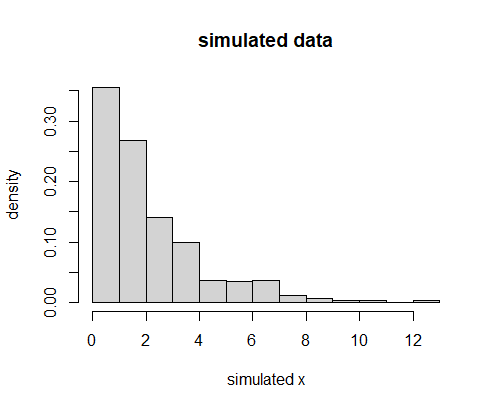
print(P\_X\_4)

OUTPUT:







**Q4. The length of the shower on a tropical island during rainy season has an exponential distribution with parameter 2, time being measured in minutes. Write R-code to find:**

**(i) the probability that a shower will last more than 3 minutes?**

**(ii) If a shower has already lasted for 2 minutes, write R-code to find probability**

**that it will last for at least one more minute?**

CODE:

#P(X>3)=1-P(X<=3)=1-F(3)

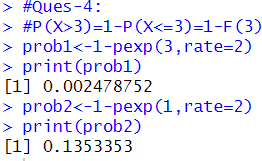
prob1<-1-pexp(3,rate=2)

print(prob1)

prob2<-1-pexp(1,rate=2)

print(prob2)

OUTPUT:



**Q5. The marks obtained by a class of M.Sc. second year students in a mathematics**

**course are found to be normally distributed with mean 64.5 and standard deviation**

**5. If class strength is 300, write R-code to find the number of students having marks**

**(i) less than 57,**

**(ii) between 57 to 72.**

CODE:

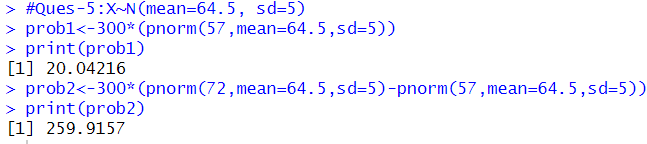
prob1<-300\*(pnorm(57,mean=64.5,sd=5))

print(prob1)

prob2<-300\*(pnorm(72,mean=64.5,sd=5)-pnorm(57,mean=64.5,sd=5))

print(prob2)

OUTPUT:



**Q6. The lifetime of certain equipment is described by a random variable X that follows a Gamma distribution with parameters α = 2 and β =**

**(i) Find the probability that the lifetime of equipment is at least 1 unit of time.**

**(ii) What is the value of c, if P(X ≤ c) ≥ 0.70?**

**(Hint: try quantile function qgamma())**

CODE:

#P(X>=1)=1-P(X<1)=1-P(X<=1)=1-F(1)

P\_i<-1-pgamma(1, shape=2,scale=1/3)

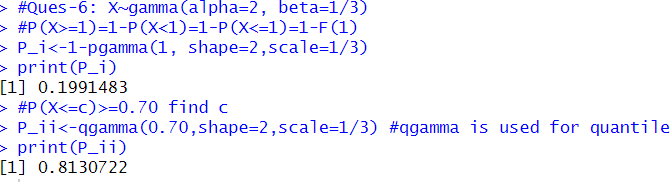
print(P\_i)

#P(X<=c)>=0.70 find c

P\_ii<-qgamma(0.70,shape=2,scale=1/3) #qgamma is used for quantile

print(P\_ii)

OUTPUT:



**Q7. At a certain examination 10% of the students who appeared for the paper in Advanced Mathematics got less than 30 marks and 97% of the students got less than 62 marks. Assuming the distribution is normal, write R-code to find the mean and the standard deviation of the distribution.**

CODE:

q1<-30 #10th percentile

p1<-0.10 #probability corresponding to q1

q2<-62 #97th percentile

p2<-0.97 #probability corresponding to q2

#use qnorm to find z value corresponding to p1 and p2

z1<-qnorm(p1) #z value for 10th percentile

z2<-qnorm(p2) #z value for 97th percentile

#z1<-q1-mu/sigma & z2<-q2-mu/sigma

#q1<=mu+Sigma \* z1 and q2<=mu+sigma \* z2

#After solving we get:

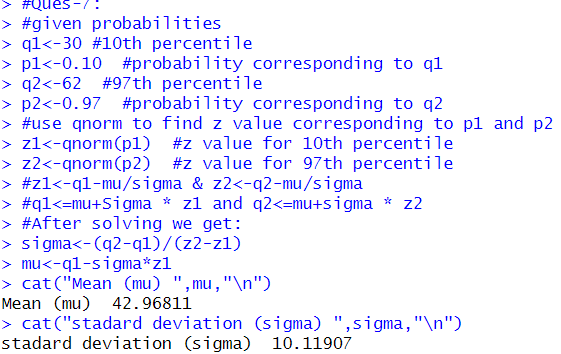
sigma<-(q2-q1)/(z2-z1)

mu<-q1-sigma\*z1

cat("Mean (mu) ",mu,"\n")

cat("stadard deviation (sigma) ",sigma,"\n")

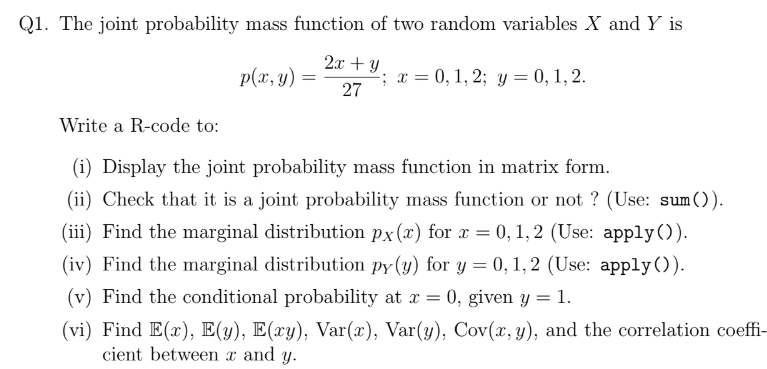
OUTPUT:



**Thapar Institute of Engineering & Technology, Patiala**

**Department of Mathematics**

**LAB Experiment 6: Joint probability mass and density functions**

****

CODE:

#f(x,y)=(2x+y)/27 x=0,1,2 and y=0,1,2

f<-function(x,y){

f1<-(2\*x+y)/27

return(f1)

}

x<-c(0:2)

y<-c(0:2)

M1<-matrix(c(f(0,0:2),f(1,0:2),f(2,0:2)),nrow = 3, ncol = 3,byrow=TRUE)

print(M1) #byrow used for row-wise means row-wise y hai and column-wise x hai

#check if it is jpmf or not

sum(M1) #sum se double summation work krta hai

#Marginal of x for x=0,1,2

px<-apply(M1,1,sum) #1 represent row sum of matrix M1

print(px)

#Marginal of y for x=0,1,2

py<-apply(M1,2,sum) #2 represent column sum of matrix M1

print(py)

#?apply - used to check what apply command does

# conditional prob for P(X=0|Y=1)

M1[1,2]

py[2]

P\_x0\_y1<-M1[1,2]/py[2]

print(P\_x0\_y1)

x<-c(0:2)

y<-c(0:2)

#E(x),E(y),var(x),var(y),cov(x,y)

E\_X<-sum(x\*px)

E\_X

E\_X2<-sum(x\*x\*px)

E\_X2

Var\_X<-E\_X2-E\_X^2

Var\_X

E\_Y<-sum(y\*py)

E\_Y

E\_Y2<-sum(y\*y\*py)

E\_Y2

Var\_Y<-E\_Y2-E\_Y^2

Var\_Y

x<-c(0:2)

y<-c(0:2)

f1<-function(x,y){x\*y\*(2\*x+y)/27}

M2<-matrix(c(f1(0,0:2),f1(1,0:2),f1(2,0:2)),nrow=3,ncol=3,byrow=T)

M2

E\_XY<-sum(M2)

E\_XY

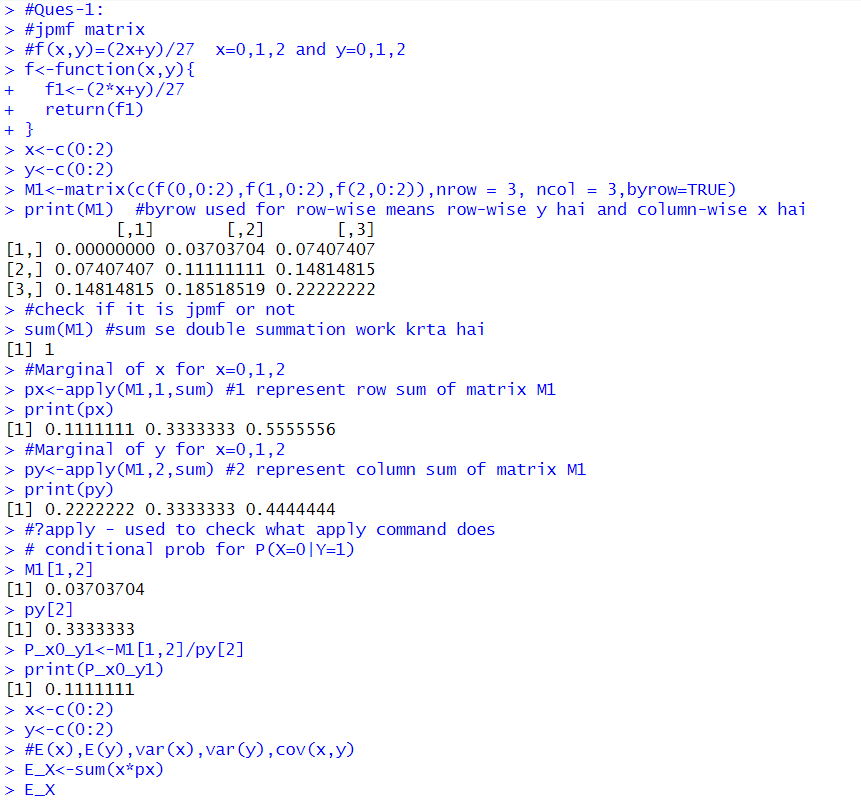
cov\_XY<-E\_XY-E\_X\*E\_Y

cov\_XY

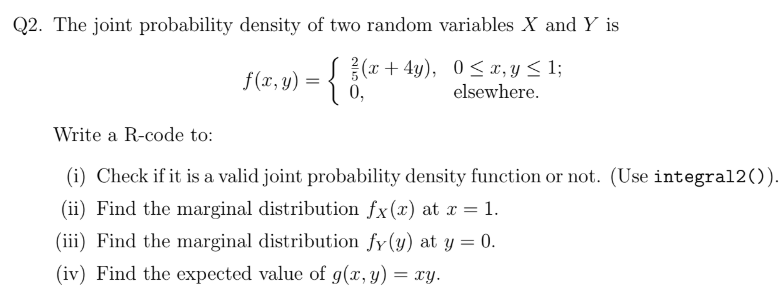
rho<-cov\_XY/sqrt(Var\_X\*Var\_Y)

Rho

OUTPUT:







CODE:

#(a) f(x,y)=2/5\*(x+4y) 0<x,y<1

library(pracma)

f<-function(x,y){2\*(x+4\*y)/5}

I<-integral2(f, xmin = 0, xmax = 1, ymin = 0,ymax = 1)

print(I$Q)

#marginal distribution fx(X) at x=1

fx1<-function(y) f(1,y)

gx1<-integral(fx1,0,1)

print(gx1)

#marginal distribution fy(Y) at y=0

fy0<-function(x) f(x,0)

hy0<-integral(fy0,0,1)

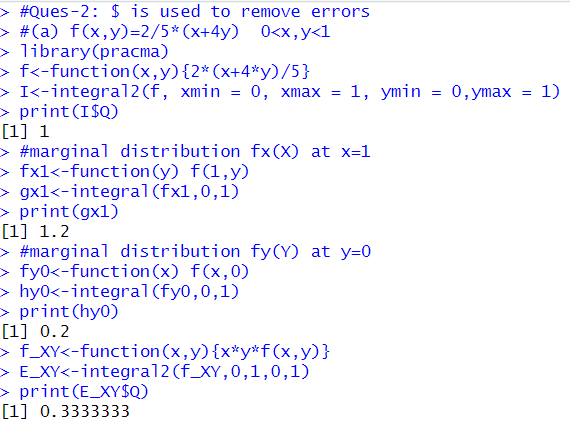
print(hy0)

f\_XY<-function(x,y){x\*y\*f(x,y)}

E\_XY<-integral2(f\_XY,0,1,0,1)

print(E\_XY$Q)

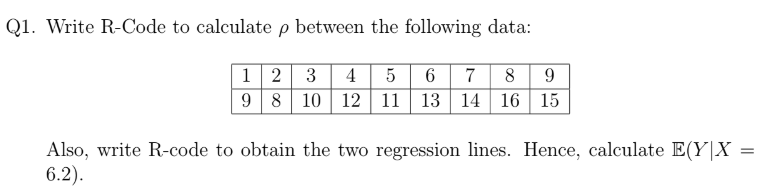
OUTPUT:



**Thapar Institute of Engineering & Technology, Patiala**

**Department of Mathematics**

**LAB Experiment 7: Correlation and Regression**

****

CODE:

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

y<-c(9,8,10,12,11,13,14,16,15)

n<-9

E\_x<-sum(x)/n

E\_x

E\_y<-sum(y)/n

E\_y

E\_x\_2<-sum(x^2)/n

E\_x\_2

E\_y\_2<-sum(y^2)/n

E\_y\_2

E\_xy<-sum(x\*y)/n

cov\_xy<-E\_xy-E\_x\*E\_y

var\_x<-E\_x\_2-E\_x^2

var\_y<-E\_y\_2-E\_y^2

rho<-cov\_xy/sqrt(var\_x\*var\_y)

rho

#OR using inbuilt function!

rho<-cor(x,y)

cat("Corr, coeff, rho b/w x and y= ",rho,"\n")

#ALso write Rcode to find lines of regression!

#Y on X i.e. E\_YonX

reg\_Y\_on\_X<-lm(y~x)

cat("Reg line Y on X: Y =",coef(reg\_Y\_on\_X)[1],"+",coef(reg\_Y\_on\_X)[2],

"\*x\n")

#Calculate at x=6.2

reg\_Y\_on\_X<-lm(y~x)

cat("Reg line Y on X=6.2: Y =",coef(reg\_Y\_on\_X)[1],"+",coef(reg\_Y\_on\_X)[2],

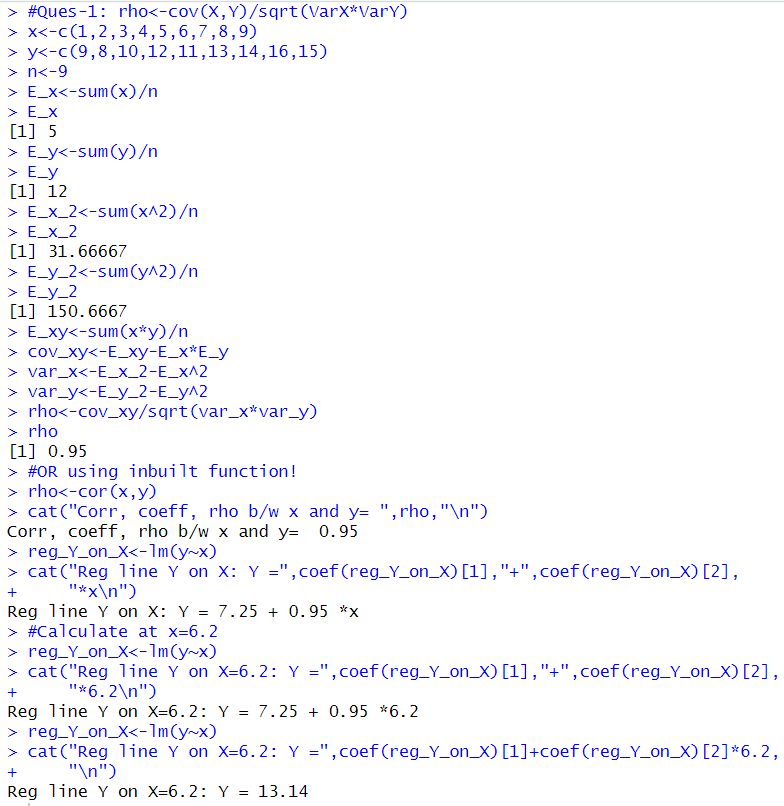
"\*6.2\n")

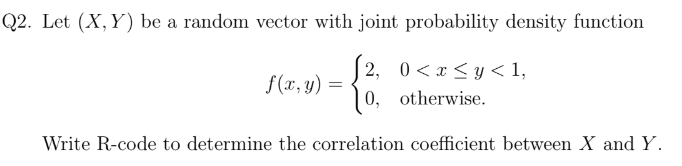
reg\_Y\_on\_X<-lm(y~x)

cat("Reg line Y on X=6.2: Y =",coef(reg\_Y\_on\_X)[1]+coef(reg\_Y\_on\_X)[2]\*6.2,

"\n")

OUTPUT:



****

CODE:

reg\_Y\_on\_x<-lm(y~x)

cat("Reg line Y on X: Y =",coef(reg\_Y\_on\_X)[1],"+",coef(reg\_Y\_on\_X)[2],

"\*x\n")

#f(x,y)=2 0<x<y<=1

n<-100000

#get sum for y from a uniform distribution in (0 1)

y<-runif(n,min=0,max=1)

#get sum for x from a unif dist in 0<x<=y

x<-runif(n,min=0,max=y)

#calculate corr coeff b/w x and y

corr\_coeff<-cor(x,y)

cat("Approx corr coeff(rho): ", corr\_coeff,"\n")

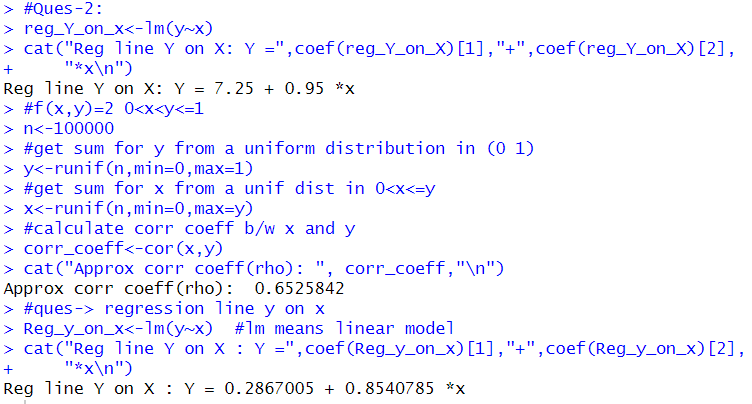
#ques-> regression line y on x

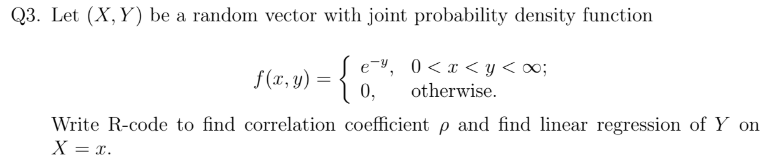
Reg\_y\_on\_x<-lm(y~x) #lm means linear model

cat("Reg line Y on X : Y =",coef(Reg\_y\_on\_x)[1],"+",coef(Reg\_y\_on\_x)[2],

"\*x\n")

OUTPUT:



****

CODE:

n<-100000

#Generate samples for Y using exponential distr (rate=1)

y<-rexp(n,rate = 1)

#generate x conditional y so that 0<x<y

x<-runif(n, min=0, max=y)

#calc the coerr coeff b/w x and y

corr\_coeff<-cor(x,y)

cat("Approx corr coeff(rho): ", corr\_coeff,"\n")

Reg\_y\_on\_x<-lm(y~x) #lm means linear model

cat("Reg line Y on X : Y =",coef(Reg\_y\_on\_x)[1],"+",coef(Reg\_y\_on\_x)[2],

"\*x\n")

OUTPUT:

