```
#q1.Create a vector c = [5,10,15,20,25,30] and write a program which returns the max
#imum and minimum of this vector.
vect=c(5,10,15,20,25,30)
max(vect)
min(vect)

#q2 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.
num = as.integer(readline(prompt="Enter a number: "))
res = 1

if (num < 0) {
  print("Error: Factorial of a negative number does not exist.")
} else {
  for (i in 1:num) {
    res = res * i
  }
  print(paste("Factorial is", res))
}

##q3.Write a program to write first n terms of a Fibonacci sequence. You may take n as an
#input from the user.
number = as.integer(readline(prompt="enter no."))
temp1=0
temp2=1
res_vect=c(0,1)
for(i in 2:number){
  temp2=temp1+temp2
  temp1=temp2-temp1
  res_vect=append(res_vect,temp2)
}
res_vect

##q4  Write an R program to make a simple calculator which can add, subtract, multiply
#and divide.
add=function(x,y){
  return(x+y)
}
sub=function(x,y){
  return(x-y)
}
mul=function(x,y){
  return(x*y)
}
```

```
div=function(x,y){
  return(x/y)
}
num1=as.integer(readline(prompt="enter num1"))
num2=as.integer(readline(prompt="enter num2"))
choice=as.integer(readline(prompt="enter choice"))
res=switch(choice,add(num1,num2),sub(num1,num2),mul(num1,num2),div(num1,num2))
res

##q5  Explore plot, pie, barplot etc. (the plotting options) which are built-in functions in R
p=c(1,2,34,5,6)
plot(p)
barplot(p)
pval=c("a","b","c")
pratio=c(70,15,15)
pie(pval,pratio)


#q6  Suppose there is a chest of coins with 20 gold, 30 silver and 50 bronze coins.
#You randomly draw 10 coins from this chest. Write an R code which will give us the
#sample space for this experiment. (use of sample(): an in-built function in R)
coins=c(rep("gold",20),rep("silver",30),rep("bronze",30))
sample(coins,10,replace=TRUE)

# In a surgical procedure, the chances of success and failure are 90% and 10%
#respectively. Generate a sample space for the next 10 surgical procedures performed.
#(use of prob(): an in-built function in R
s=c("success","failure")
sample(s,10,replace=TRUE,prob=c(0.9,0.1))

##q. A room has n people, and each has an equal chance of being born on any of the 365
#days of the year. (For simplicity, we'll ignore leap years). What is the probability
#that two people in the room have the same birthday?
n=qbirthday(prob=0.5,classes=365,coincident=2)
pbirthday(n, classes = 365, coincident = 2)

#Find the smallest value of n for which the probability of a match is greater than
#.5
smallest=qbirthday(prob=0.5,classes=365,coincident=2)
smallest


#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?
cp=function(cloudy,rainy,clouds_on_rainy_day){
  return(clouds_on_rainy_day*cloudy/rainy)
}
cloudy=0.4
rain=0.2
clouds_on_rainy_day=0.85
res=cp(cloudy,rain,clouds_on_rainy_day)
res

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

#a) Print first few rows of this dataset.
head(iris_dataset)

#b) Find the structure of this dataset.
str(iris_dataset)

#c)  Find the range of the data regarding the sepal length of flowers.
range(iris_dataset$Sepal.Length)
range(iris_dataset$Sepal.Width)
range(iris_dataset$Petal.Length)
range(iris_dataset$Petal.Width)

#(d) Find the mean of the sepal length.
mean(iris_dataset$Sepal.Length)
mean(iris_dataset$Sepal.Width)
mean(iris_dataset$Petal.Length)
mean(iris_dataset$Petal.Width)

# (e) Find the median of the sepal length
median(iris_dataset$Sepal.Length)
median(iris_dataset$Sepal.Width)
median(iris_dataset$Petal.Length)
median(iris_dataset$Petal.Width)

# (f) Find the first and the third quartiles and hence the interquartile range
q1=quantile(iris_dataset$Sepal.Length,0.25)
```

```
q3=quantile(iris_dataset$Sepal.Length,0.75)
iqr=IQR(iris_dataset$Sepal.Length)
iqr

q1=quantile(iris_dataset$Sepal.Width,0.25)
q3=quantile(iris_dataset$Sepal.Width,0.75)
iqr=IQR(iris_dataset$Sepal.Width)
iqr

q1=quantile(iris_dataset$Petal.Length,0.25)
q3=quantile(iris_dataset$Petal.Length,0.75)
iqr=IQR(iris_dataset$Petal.Length)
iqr

q1=quantile(iris_dataset$Petal.Length,0.25)
q3=quantile(iris_dataset$Petal.Length,0.75)
iqr=IQR(iris_dataset$Petal.Length)
iqr

# (g) Find the standard deviation and variance.
sd(iris_dataset$Sepal.Length)
sd(iris_dataset$Sepal.Width)
sd(iris_dataset$Petal.Length)
sd(iris_dataset$Petal.Width)

var(iris_dataset$Sepal.Length)
var(iris_dataset$Sepal.Width)
var(iris_dataset$Petal.Length)
var(iris_dataset$Petal.Width)


#R does not have a standard built-in function to calculate mode. So, we create user
#function to calculate mode of a dataset in R. This function takes the vector as input
#and gives the mode value as output.
mode_vec=c(1,2,3,3,3,4,5,6,3,3,2,2,4)
rept=tabulate(mode_vec)
which.max(match(rept,mode_vec))


## Roll 12 dice simultaneously, and let X denotes the number of 6's that appear. Calcu
#late the probability of getting 7, 8 or 9, 6's using R. (Try using the function pbinom
#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) and we are looking for P(7 ≤ X ≤ 9)
p=pbinom(9,12,1/6)-pbinom(6,12,1/6)
```

p

```
#Assume that the test scores of a college entrance exam fits a normal distribution.
#Furthermore, the mean test score is 72, and the standard deviation is 15.2. What is
#the percentage of students scoring 84 or more in the exam?
q=pnorm(83,72,15.2,lower.tail = FALSE)
q

#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 car wash 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). What is the probability that there are between 48 and 50
#customers, inclusive?
r=dpois(0,5)
r
s=ppois(50,50)-ppois(47,50)
s


#Suppose in a certain shipment of 250 Pentium processors there are 17 defective pro
#cessors. A quality control consultant randomly collects 5 processors for inspection to
#determine whether or not they are defective. Let X denote the number of defectives
#in the sample. Find the probability of exactly 3 defectives in the sample, that is, find
#P(X =3).
t=dhyper(3,17,250-17,5)
t

#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.
#(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.
## ans. Binaomial Distribution
x=c(0:31)
n=31
prob=0.447
res_pdf=dbinom(x,31,0.447)
plot(x,res_pdf)
```

```
res_cdf=pbinom(x,31,0.447)
plot(x,res_cdf)


mean=n*prob
mean
variance=n*prob*(1-prob)
variance
standard_deviation=sqrt(variance)
variance


##
x=c(0,1,2,3,4)
px=c(0.41,0.37,0.16,0.05,0.01)
m=sum(x*px)
m
wm=weighted.mean(x,px)
wm

#The time T, in days, required for the completion of a contracted project is a random
#variable with probability density function f(t) = 0.1 e(-0.1t) for t > 0 and 0 otherwise. Find
#the expected value of T.
f=function(t)
{
  t*0.1*exp(-0.1*t)
}
a=integrate(f,lower=0,upper=Inf)
a$value

#A bookstore purchases three copies of a book at $6.00 each and sells them for $12.00
#each. Unsold copies are returned for $2.00 each. Let X = {number of copies sold} and
#Y = {net revenue}. If the probability mass function of X is
x1=c(0,1,2,3)
px1=c(0.1,0.2,0.3,0.5)
y=12*x1+2*(3-x1)-18
expected_y=weighted.mean(y,px1)
expected_y


#Find the first and second moments about the origin of the random variable X with
#probability density function f(x) = 0.5e-|x|, 1 < x < 10 and 0 otherwise. Further use the
#results to find Mean and Variance.
```

```
f1=function(x){
  x*0.5*exp(-abs(x))
}
moment1=integrate(f1,lower=1,upper=10)
moment1$value

f2=function(x){
  x*x*0.5*exp(-abs(x))
}
moment2=integrate(f2,lower=1,upper=10)
moment2$value


varianc=moment2$value-(moment1$value*moment1$value)
varianc


#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.
ff=function(x)
{
  0.75*(0.25**(x-1))
}
x=c(1,2,3,4,5)
at3=ff(3)
at3

fm=weighted.mean(x**2,ff(x))

sm=weighted.mean(x**4,ff(x))


variancee=sm-(fm*fm)
variancee
```