**MAT2001 – Statistics for Engineers - ELA (R Code Studio), Winter Semester 2020-2021**

**Lab Assessment - I**

**By: Jonathan Rufus Samuel (20BCT0332) Date: 17.6.2021**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Q1) Create the following graphs using R:**

A: Code is as follows:

#Q1.1

f1 = function(x){

return(abs(sin(x)))

}

curve(expr = f1, from = -4\*pi, to = 4\*pi, col="blue", main="plot of abs(sin(x))")

#Q1.2

f1 = function(x){

return(sin(x))

}

curve(expr = f1, from = -2\*pi, to = 2\*pi, col="green", main="plot of sin(x)")

f2 = function(x){

return(cos(x))

}

curve(expr = f1, from = -2\*pi, to = 2\*pi, col="red", main="plot of cos(x)")

#Q1.3

plot(x,y,type="n")

segments(x[-length(x)],y[-length(x)],x[-1],y[-length(x)])

points(x[-length(x)],y[-length(x)],pch=16)

points(x[-1],y[-length(x)],pch=1)

#Q1.4

f1 = function(x){

return(abs(x))

}

curve(expr = f1, from = -4\*pi, to = 4\*pi, col="blue", main="plot of abs(x)")

#Q1.5

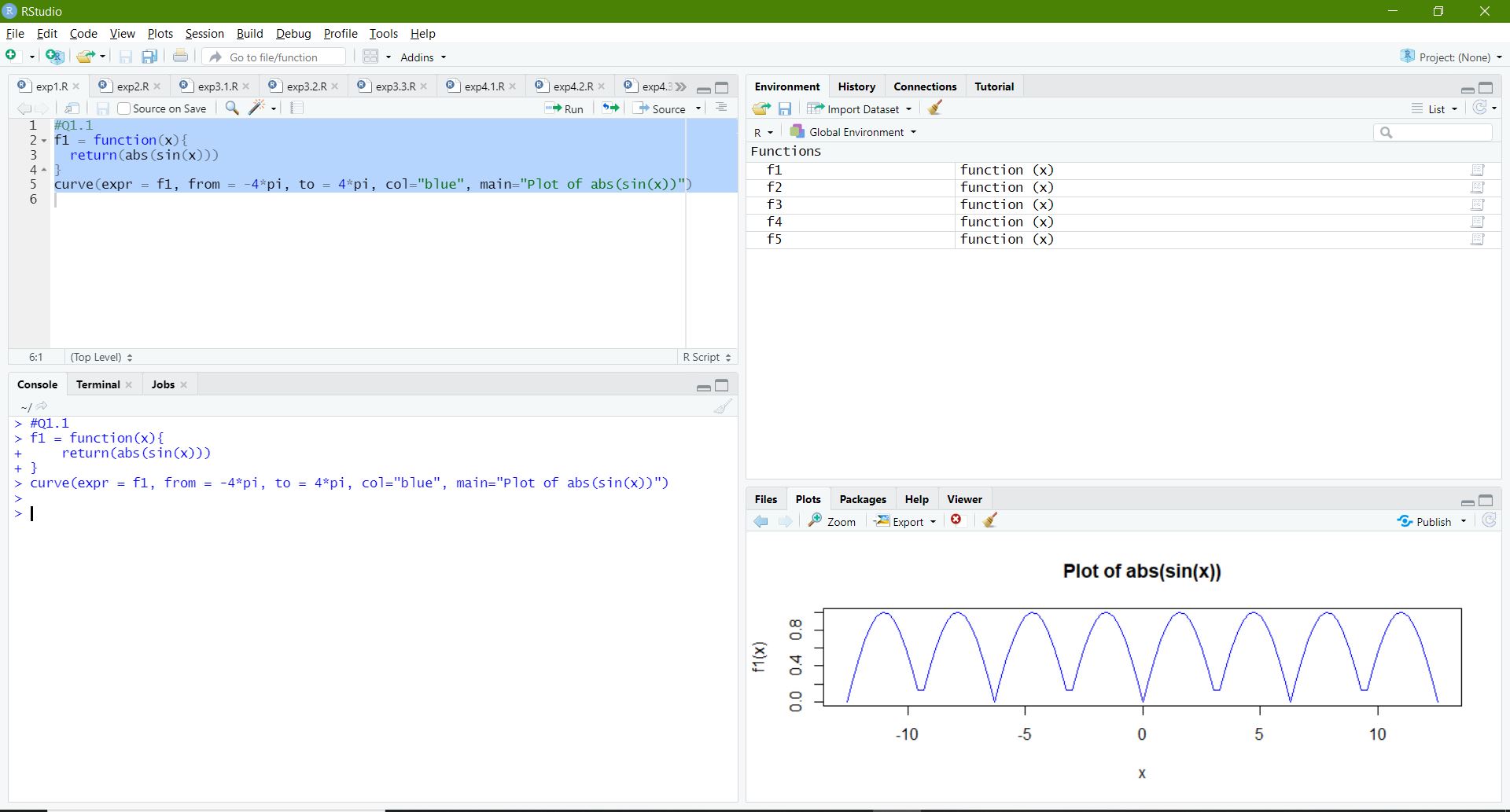
f1 = function(x){

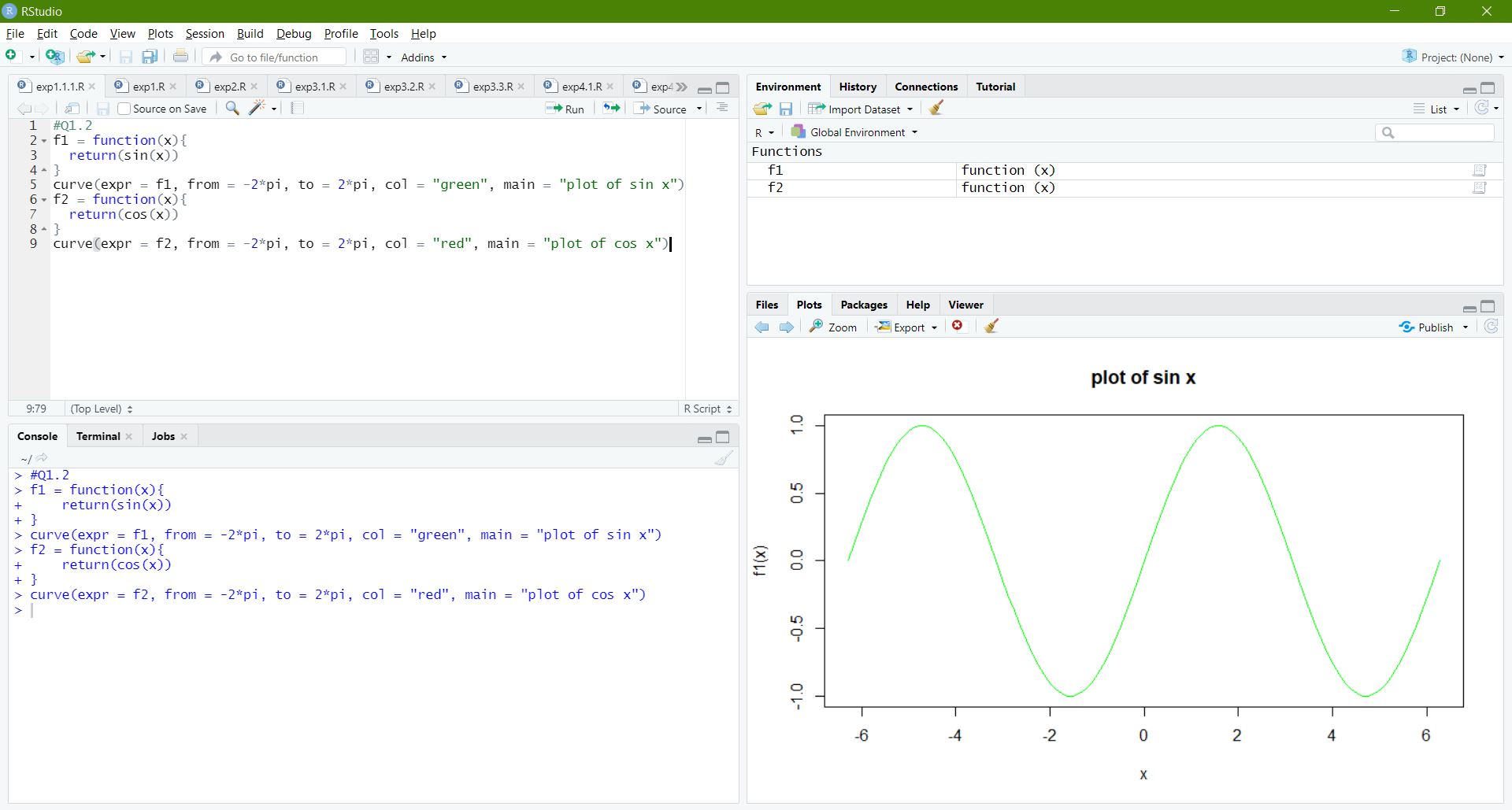
return(x^2)

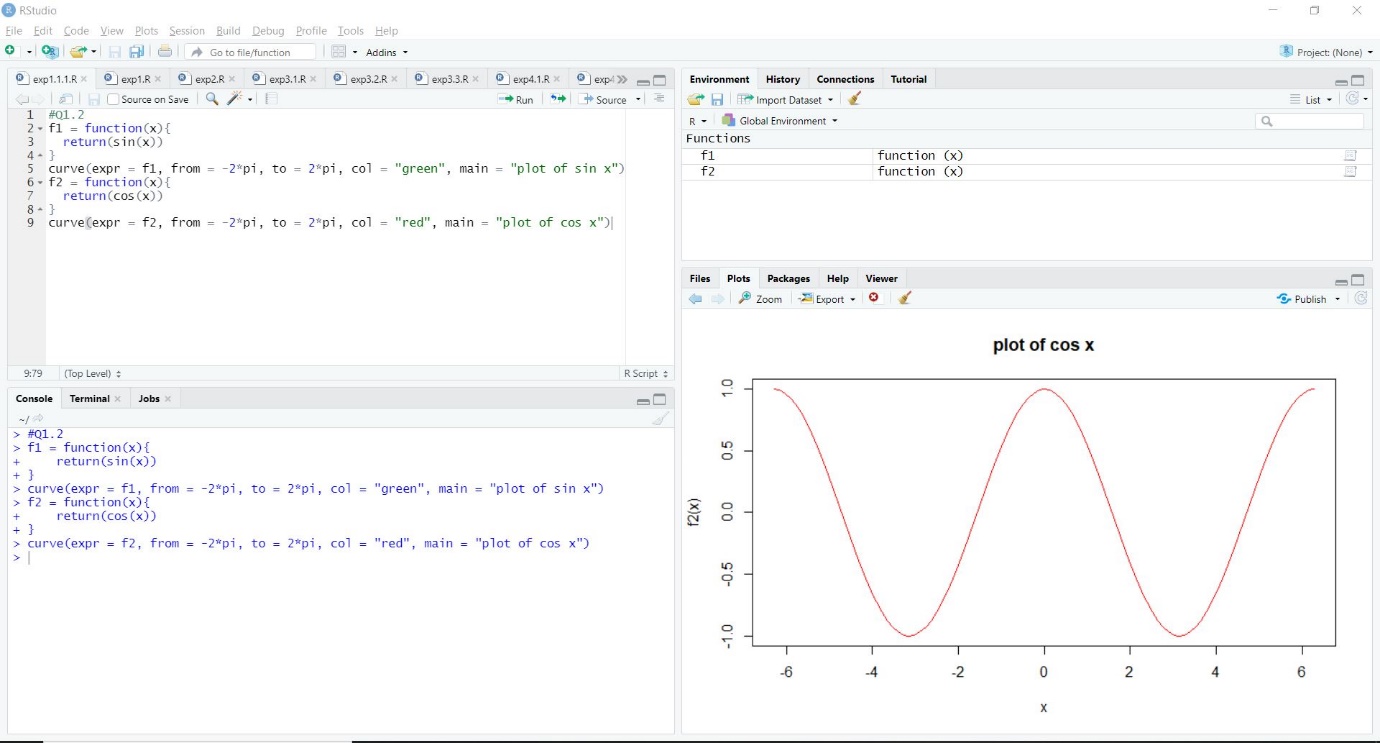
}

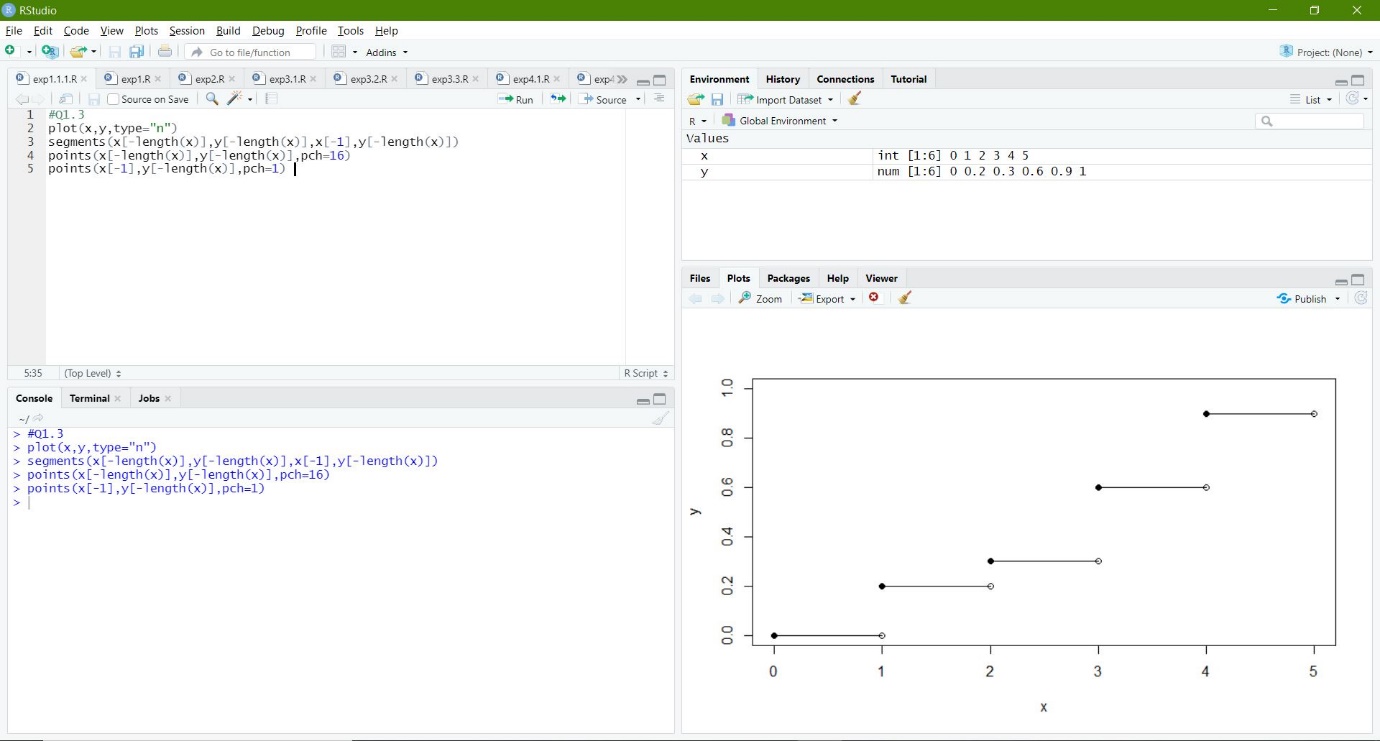
curve(expr = f1, from = -4\*pi, to = 4\*pi, col="blue", main="plot of x^2")

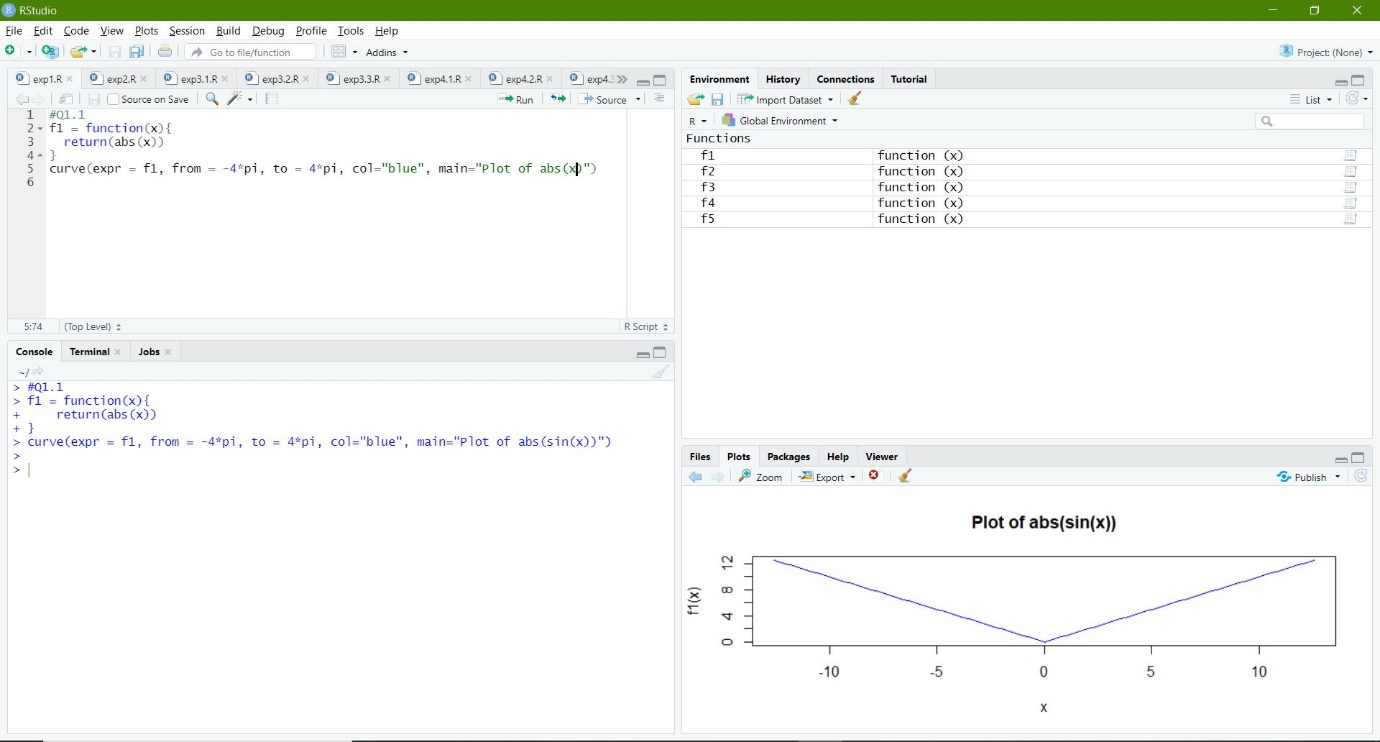
**Output (via Command Window):**

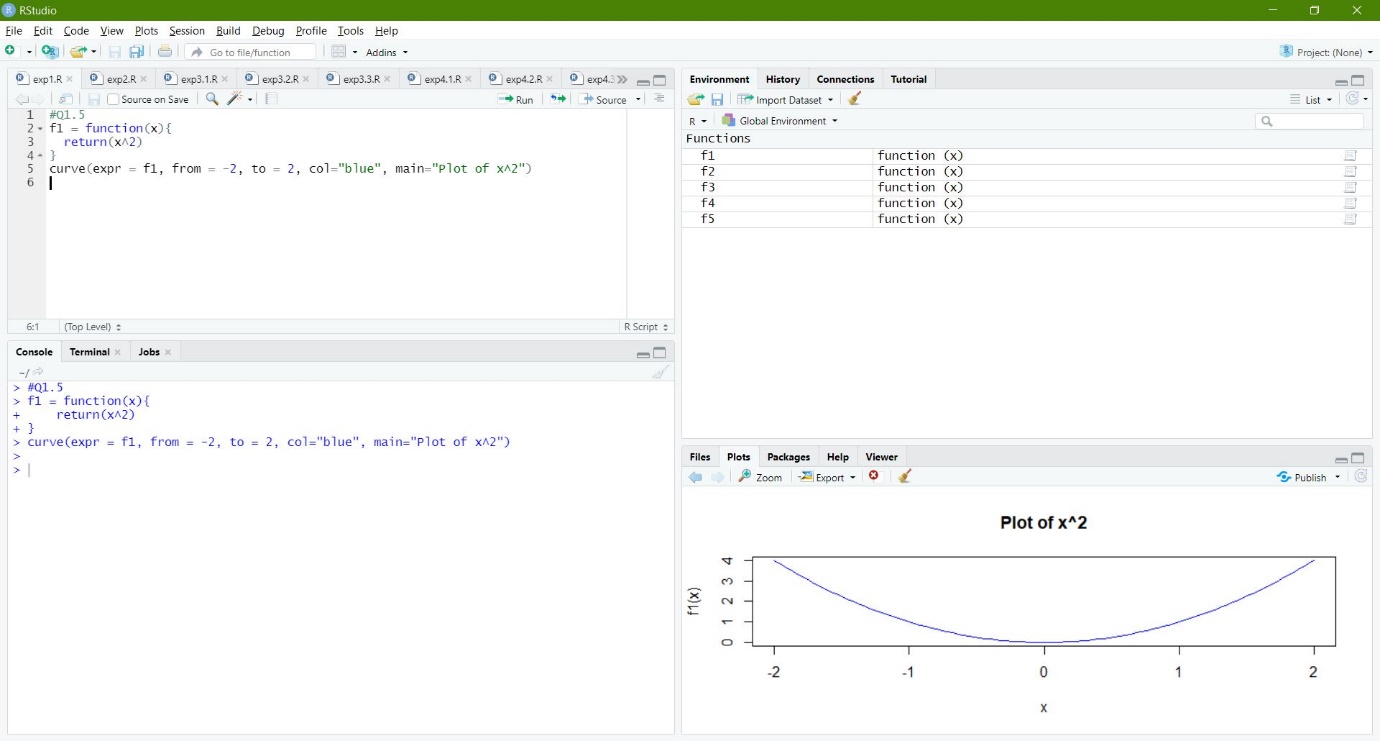












**Q2) Write R code to print a Fibonacci sequence using any of the loop statements:**

A: Code is as follows:

#Q2

# Let Number of terms be 20

nterms = 20

# first two terms

n1 = 0

n2 = 1

count = 2

# check if the number of terms is valid

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

}

}

}

**Output (via Command Window):**

> #Q2

> # Let Number of terms be 20

> nterms = 20

> # first two terms

> n1 = 0

> n2 = 1

> count = 2

> # check if the number of terms is valid

> 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

+ }

+ }

+ }

[1] "Fibonacci sequence:"

[1] 0

[1] 1

[1] 1

[1] 2

[1] 3

[1] 5

[1] 8

[1] 13

[1] 21

[1] 34

[1] 55

[1] 89

[1] 144

[1] 233

[1] 377

[1] 610

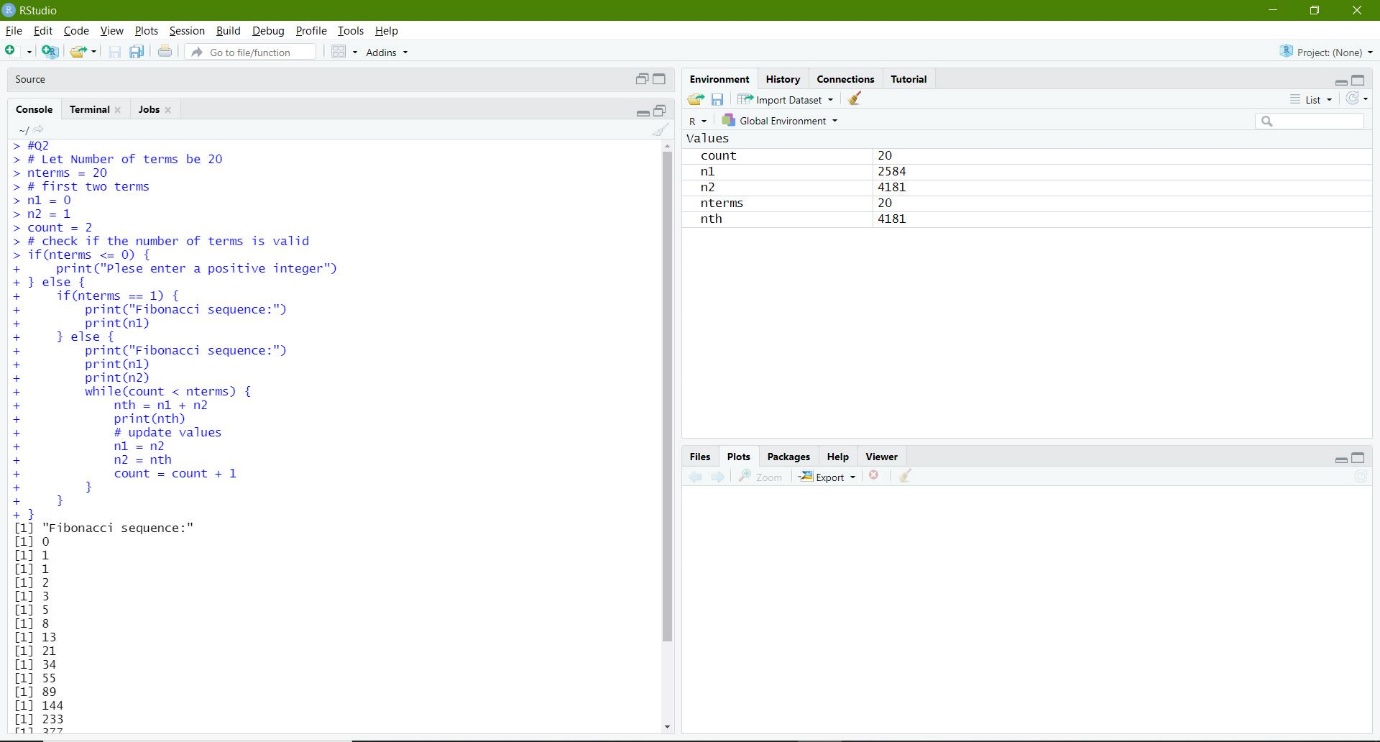
[1] 987

[1] 1597

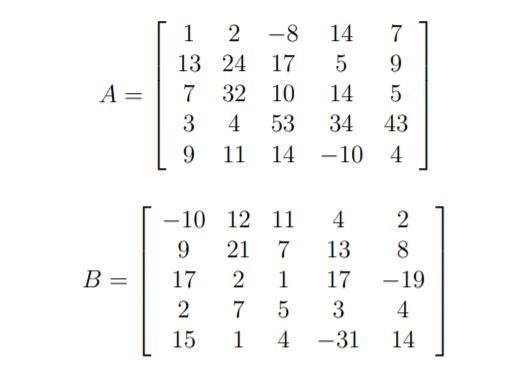
[1] 2584

[1] 4181

**Implementation on R Studio Code (via Command Window):**

****

**Q3) Write R code to find the following if:**

****

**(i) the eigenvalue and eigenvector of A and B.**

**(ii) check whether (AB)−1 = B−1A−1**

**(iii) dimension of 4 ∗ A5 − 5A3 + A2**

**(iv) replace 4th row of A by (5 -4 6 3 2) and 5th column of B by (14 9 43 24 26).**

A: Code is as follows:

#Q3

A = c(1,13,7,3,9,2,24,32,4,11,-8,17,10,53,14,14,5,14,34,-10,7,9,5,43,4)

B = c(-10,9,17,2,15,12,21,2,7,1,11,7,1,5,4,4,13,17,3,-31,2,8,-19,4,14)

dim(A) = c(5,5)

dim(B) = c(5,5)

A

B

#Q3.1

eigen(A)

eigen(B)

#Q3.2

x = (A\*B)^-1

y = A^-1 \* B^-1

x

y

#Therefore it is verified as X == Y

#Q3.3

z = (4 \* (A^5)) - (5\*(A^3)) + (A^2)

z

#dimensions of Z is 5 x 5 matrix

#Q3.4

A

B

A\_new = c(5,-4,6,3,2)

B\_new = c(14,9,43,24,26)

A[4,] = A\_new

B[,5] = B\_new

A

B**Output (via Command Window):**

> #Q3

> A = c(1,13,7,3,9,2,24,32,4,11,-8,17,10,53,14,14,5,14,34,-10,7,9,5,43,4)

> B = c(-10,9,17,2,15,12,21,2,7,1,11,7,1,5,4,4,13,17,3,-31,2,8,-19,4,14)

> dim(A) = c(5,5)

> dim(B) = c(5,5)

> A

[,1] [,2] [,3] [,4] [,5]

[1,] 1 2 -8 14 7

[2,] 13 24 17 5 9

[3,] 7 32 10 14 5

[4,] 3 4 53 34 43

[5,] 9 11 14 -10 4

> B

[,1] [,2] [,3] [,4] [,5]

[1,] -10 12 11 4 2

[2,] 9 21 7 13 8

[3,] 17 2 1 17 -19

[4,] 2 7 5 3 4

[5,] 15 1 4 -31 14

> #Q3.1

> eigen(A)

eigen() decomposition

$values

[1] 65.237708+ 0.00000i 4.280246+13.39402i 4.280246-13.39402i -9.199137+ 0.00000i

[5] 8.400937+ 0.00000i

$vectors

[,1] [,2] [,3] [,4]

[1,] 0.1431104+0i 0.3873298-0.3300622i 0.3873298+0.3300622i -0.66347869+0i

[2,] 0.3293292+0i -0.2176621+0.0211566i -0.2176621-0.0211566i 0.31679845+0i

[3,] 0.4232249+0i -0.0387338-0.1563271i -0.0387338+0.1563271i -0.38676369+0i

[4,] 0.8308060+0i 0.6035959+0.0000000i 0.6035959+0.0000000i -0.05963256+0i

[5,] 0.0412769+0i -0.3762133+0.4017552i -0.3762133-0.4017552i 0.55343705+0i

[,5]

[1,] 0.1076409+0i

[2,] -0.1919504+0i

[3,] 0.2908784+0i

[4,] 0.6045964+0i

[5,] -0.7081112+0i

> eigen(B)

eigen() decomposition

$values

[1] 32.70316+ 0.00000i -21.55109+ 0.00000i 6.55190+10.80064i 6.55190-10.80064i

[5] 4.74412+ 0.00000i

$vectors

[,1] [,2] [,3] [,4]

[1,] -0.3526381+0i 0.69384832+0i 0.13355646-0.06567600i 0.13355646+0.06567600i

[2,] -0.7934667+0i -0.04667258+0i -0.34811884+0.14419030i -0.34811884-0.14419030i

[3,] -0.4111692+0i -0.70032419+0i 0.71282735+0.00000000i 0.71282735+0.00000000i

[4,] -0.2743534+0i 0.11737968+0i -0.08512726-0.08243612i -0.08512726+0.08243612i

[5,] 0.0415566+0i -0.11029158+0i -0.20160469-0.52255347i -0.20160469+0.52255347i

[,5]

[1,] -0.14957716+0i

[2,] 0.59194943+0i

[3,] -0.76844062+0i

[4,] -0.17544646+0i

[5,] -0.07707206+0i

> #Q3.2

> x = (A\*B)^-1

> y = A^-1 \* B^-1

> x

[,1] [,2] [,3] [,4] [,5]

[1,] -0.100000000 0.041666667 -0.011363636 0.017857143 0.071428571

[2,] 0.008547009 0.001984127 0.008403361 0.015384615 0.013888889

[3,] 0.008403361 0.015625000 0.100000000 0.004201681 -0.010526316

[4,] 0.166666667 0.035714286 0.003773585 0.009803922 0.005813953

[5,] 0.007407407 0.090909091 0.017857143 0.003225806 0.017857143

> y

[,1] [,2] [,3] [,4] [,5]

[1,] -0.100000000 0.041666667 -0.011363636 0.017857143 0.071428571

[2,] 0.008547009 0.001984127 0.008403361 0.015384615 0.013888889

[3,] 0.008403361 0.015625000 0.100000000 0.004201681 -0.010526316

[4,] 0.166666667 0.035714286 0.003773585 0.009803922 0.005813953

[5,] 0.007407407 0.090909091 0.017857143 0.003225806 0.017857143

> #Therefore it is verified as X == Y

> #Q3.3

> z = (4 \* (A^5)) - (5\*(A^3)) + (A^2)

> z

[,1] [,2] [,3] [,4] [,5]

[1,] 0 92 -128448 2137772 65562

[2,] 1474356 31781952 5655152 11900 232632

[3,] 65562 134054912 395100 2137772 11900

[4,] 846 3792 1672040396 181546332 587638086

[5,] 232632 637670 2137772 -394900 3792

> #dimensions of Z is 5 x 5 matrix

> #Q3.4

> A

[,1] [,2] [,3] [,4] [,5]

[1,] 1 2 -8 14 7

[2,] 13 24 17 5 9

[3,] 7 32 10 14 5

[4,] 3 4 53 34 43

[5,] 9 11 14 -10 4

> B

[,1] [,2] [,3] [,4] [,5]

[1,] -10 12 11 4 2

[2,] 9 21 7 13 8

[3,] 17 2 1 17 -19

[4,] 2 7 5 3 4

[5,] 15 1 4 -31 14

> A\_new = c(5,-4,6,3,2)

> B\_new = c(14,9,43,24,26)

> A[4,] = A\_new

> B[,5] = B\_new

> A

[,1] [,2] [,3] [,4] [,5]

[1,] 1 2 -8 14 7

[2,] 13 24 17 5 9

[3,] 7 32 10 14 5

[4,] 5 -4 6 3 2

[5,] 9 11 14 -10 4

> B

[,1] [,2] [,3] [,4] [,5]

[1,] -10 12 11 4 14

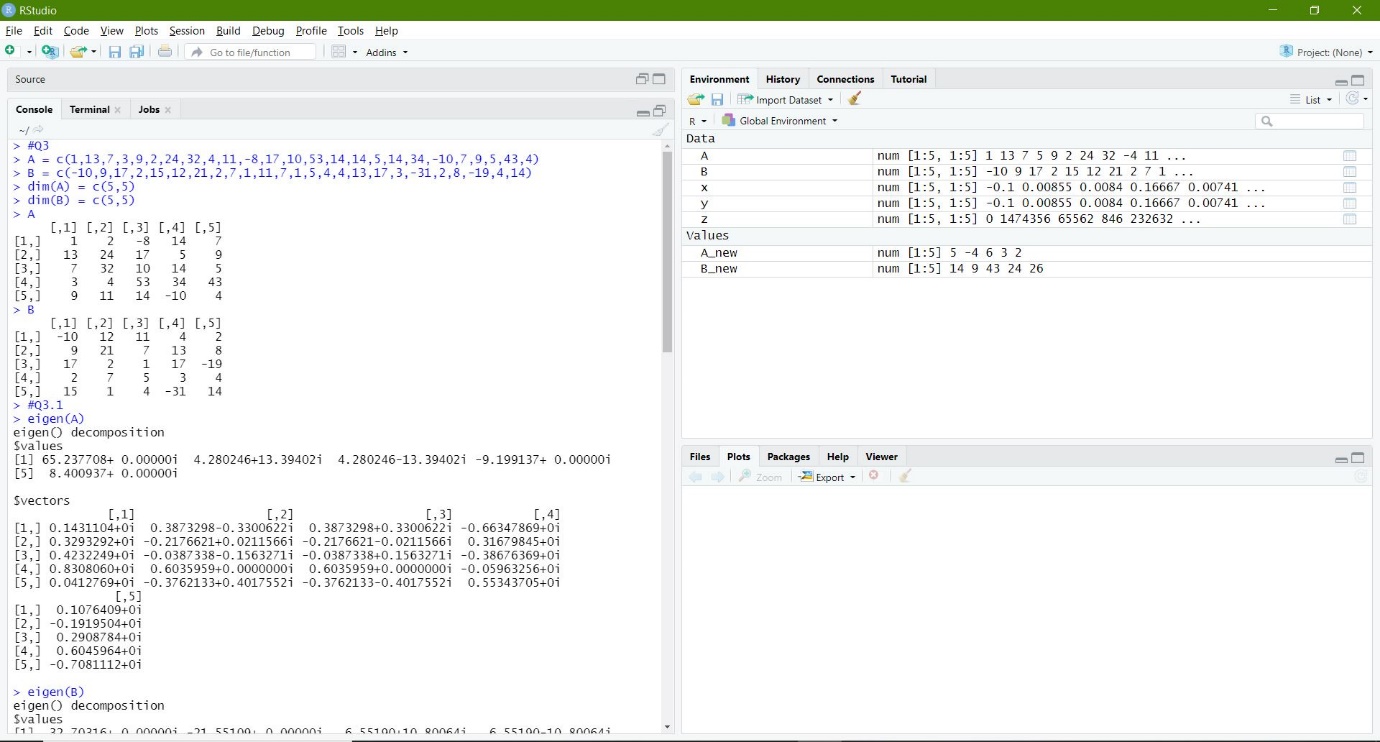
[2,] 9 21 7 13 9

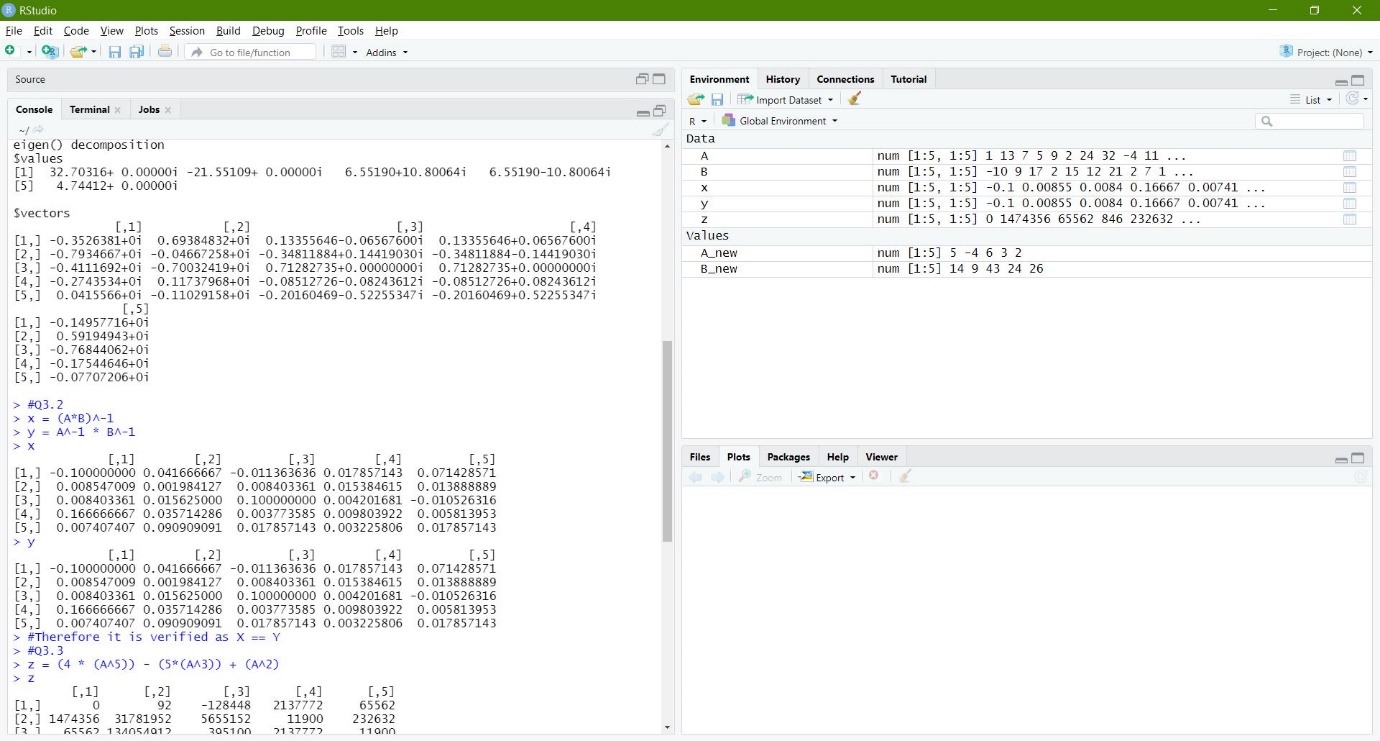
[3,] 17 2 1 17 43

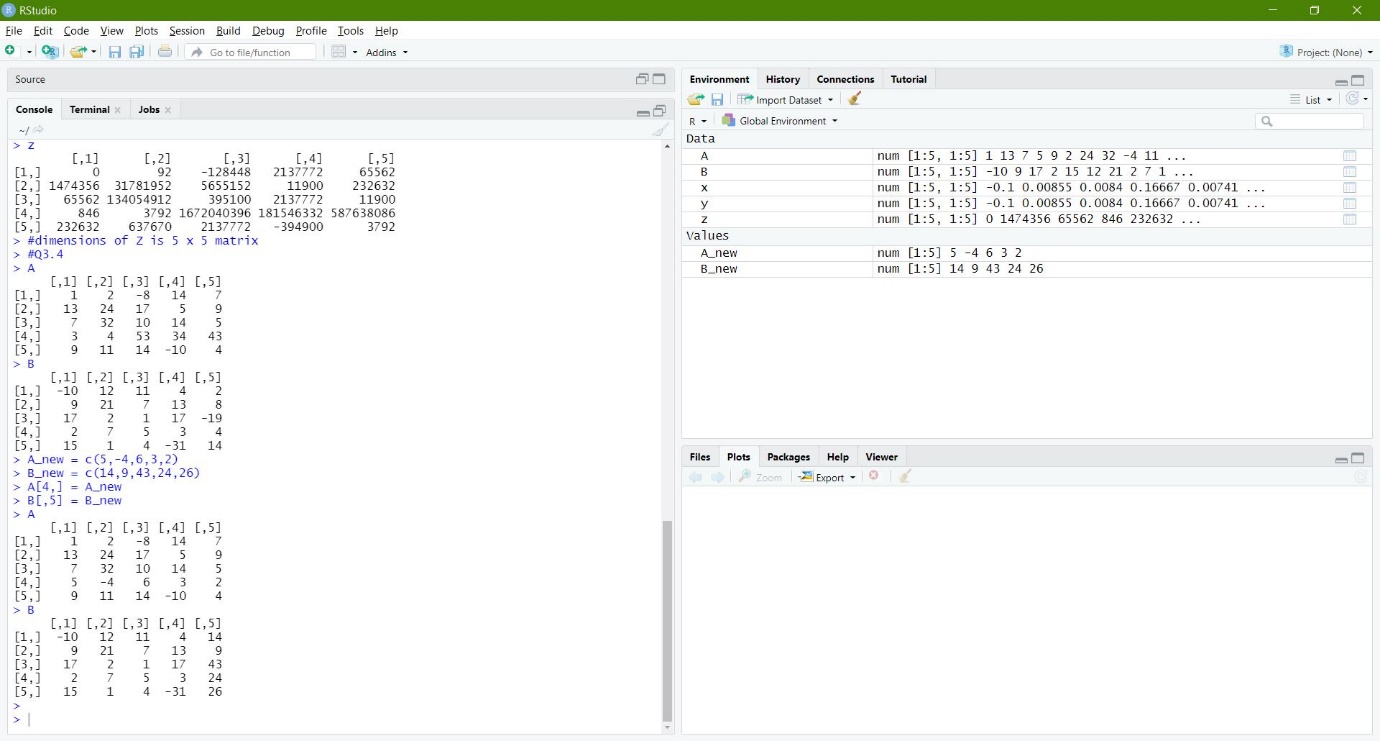
[4,] 2 7 5 3 24

[5,] 15 1 4 -31 26

**Implementation on R Studio Code (via Command Window):**

****

****

****

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_