def like\_a\_gauss(mat):

"""

Changes mat into Reduced Row-Echelon Form.

"""

# Let's do forward step first.

# at the end of this for loop, the matrix is in Row-Echelon format.

for i in range(min(len(mat), len(mat[0]))):

# every iteration, ignore one more row and column

for r in range(i, len(mat)):

# find the first row with a nonzero entry in first column

zero\_row = mat[r][i] == 0

if zero\_row:

continue

# swap current row with first row

mat[i], mat[r] = mat[r], mat[i]

# add multiples of the new first row to lower rows so lower

# entries of first column is zero

first\_row\_first\_col = mat[i][i]

for rr in range(i + 1, len(mat)):

this\_row\_first = mat[rr][i]

scalarMultiple = -1 \* this\_row\_first / first\_row\_first\_col

for cc in range(i, len(mat[0])):

mat[rr][cc] += mat[i][cc] \* scalarMultiple

break

# Now reduce

for i in range(min(len(mat), len(mat[0])) - 1, -1, -1):

# divide last non-zero row by first non-zero entry

first\_elem\_col = -1

first\_elem = -1

for c in range(len(mat[0])):

if mat[i][c] == 0:

continue

if first\_elem\_col == -1:

first\_elem\_col = c

first\_elem = mat[i][c]

mat[i][c] /= first\_elem

# add multiples of this row so all numbers above the leading 1 is zero

for r in range(i):

this\_row\_above = mat[r][first\_elem\_col]

scalarMultiple = -1 \* this\_row\_above

for cc in range(len(mat[0])):

mat[r][cc] += mat[i][cc] \* scalarMultiple

# disregard this row and continue

print(mat)

#Enter values of matrix

n = input("Enter dimension of matrix: ")

ma = []

for r in range (0,(int)(n),1):

mn = []

for c in range(0,(int)(n),1):

x = input("Enter value of matrix A at [%(r)i][%(c)i]: " %{'r':r, 'c':c})

mn.append((int)(x))

ma.append(mn)

mb = []

print("\n ")

for r in range (0,(int)(n),1):

mn = []

x = input("Enter value of matrix B at [%(r)i][0]: " %{'r':r})

mn.append((int)(x))

mb.append(mn)

print("\n")

print ("Matrix A:")

print (ma)

print ("Matrix B:")

print (mb)

print ("\n")

#Matrix multiplication

mx = []

for r in range (0,(int)(n),1):

mn = []

x = 0

for c in range(0,(int)(n),1):

x+=ma[r][c]\*mb[c][0]

mn.append(x)

mx.append(mn)

print ("Matrix X:")

print (mx)

#Reduced Row Echelon Form

mr = ma

for r in range (0,(int)(n),1):

mr[r].append(mb[r][0])

print ("\nMatrix R:")

print (mr)

print("rref: ")

like\_a\_gauss(mr)