```python
# Python - Program for matrix inversion using Jacobi Iteration Method.

#---Imports----------------
import numpy as np
from scipy.linalg import solve

#---Inputs-----------------
A = eval(input("Enter the matrix A : "))              # As np.array([[a11, a12], [a21, a22]])
B = eval(input("Enter the matrix B : "))              # As [b1, b2]
X = eval(input("Enter initial guess X : "))           # As [x1, x2]
n = eval(input("Enter the number of iteration : "))   # Integer input.

#-----Calculation----------
D = np.diag(A)
R = A - np.diagflat(D)                                # R = L+U = Lower + Upper Triangular matrix

for i in range(n):
  X = (B - np.dot(R,X))/D

#------Output--------------
print()
print("Solution using Jacobi’s Iteration Method : ", X)
print()
print("Solution using Solve Syntax : ", solve(A,B))
```