

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import numpy as np

# -----
# Lecture 11 - Gauss Elimination
# (B) Backward Substitution
# -----
#
# - Solve  $U.x = b'$  for  $x$ 
#

A = np.array([[ -3., -5., -5., -2.],
               [  0.,  5.,  8.,  1.],
               [  0.,  0., -7.,  2.],
               [  0.,  0.,  0., -4.]])

b = np.array([6., 3., -7., 0.])
n = len(b)

# Backward Substitution algorithm:
#  $x_i = (b_i - \sum_j (a_{ij} * x_j)) / a_{ii}$ 

x = np.zeros(n)
x[n-1] = b[n-1]/A[n-1,n-1] # do the bottom one first (no sum needed)
for i in range(n-2, -1, -1): # loop over the rows from bottom to top
    xi_sum = 0
    for j in range(i+1, n): # loop over all the known  $x_i$ 's
        xi_sum += A[i,j]*x[j]

    x[i] = (b[i] - xi_sum)/A[i,i]

print('Finished...\n')
print('-'*20)
print('x:')
print(x)
print('residual:')
print(np.linalg.norm(np.dot(A,x) - b))
print('-'*20)

```