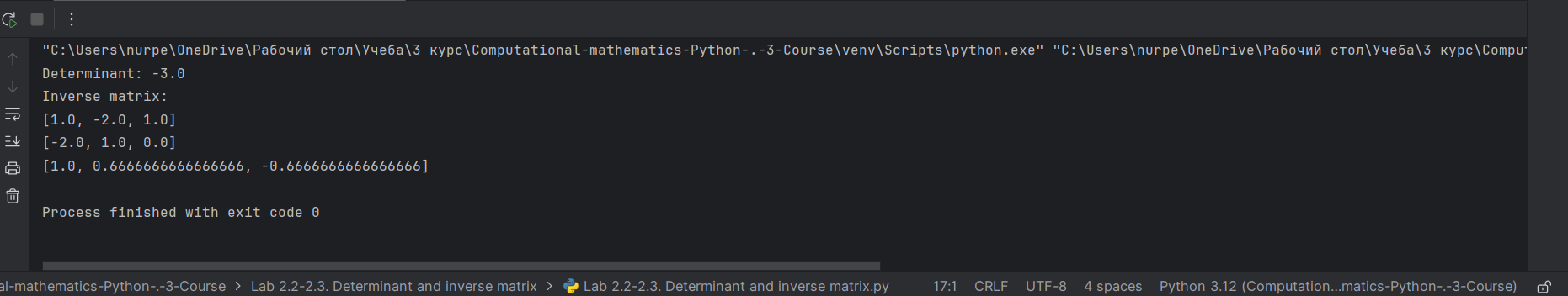
IT3-2208 Zhenisov Zhanserik

1. def determinant\_and\_inverse(matrix):  
 n = len(matrix)  
 det = 1  
  
 augmented\_matrix = [row[:] + [1 if i == j else 0 for j in range(n)] for i, row in enumerate(matrix)]  
  
 for i in range(n):  
 if augmented\_matrix[i][i] == 0:  
 for k in range(i + 1, n):  
 if augmented\_matrix[k][i] != 0:  
 augmented\_matrix[i], augmented\_matrix[k] = augmented\_matrix[k], augmented\_matrix[i]  
 det \*= -1  
 break  
 else:  
 print("Matrix is singular and does not have an inverse.")  
 return None, 0  
  
 pivot = augmented\_matrix[i][i]  
 det \*= pivot  
  
 for j in range(2 \* n):  
 augmented\_matrix[i][j] /= pivot  
  
 for k in range(n):  
 if k != i:  
 factor = augmented\_matrix[k][i]  
 for j in range(2 \* n):  
 augmented\_matrix[k][j] -= factor \* augmented\_matrix[i][j]  
  
 inverse\_matrix = [row[n:] for row in augmented\_matrix]  
  
 return inverse\_matrix, det  
  
  
Matrix = [  
 [2, 2, 3],  
 [4, 5, 6],  
 [7, 8, 9]  
]  
  
inverse, det = determinant\_and\_inverse(Matrix)  
  
print("Determinant:", det)  
if inverse:  
 print("Inverse matrix:")  
 for row in inverse:  
 print(row)

2. 

3. 