```
# Fall 2018
# Lab 2

# Provided encryption method from text
def substitutionEncrypt(plainText,key):
        alphabet = "abcdefghijklmnopqrstuvwxyz "
        plainText = plainText.lower()
        cipherText = ""
        for ch in plainText:
            idx = alphabet.find(ch)
            cipherText = cipherText + key[idx]
        return cipherText

# Written decryption method
def substitutionDecrypt(cipherText, key):
    alphabet = "abcdefghijklmnopqrstuvwxyz "
    cipherText = cipherText.lower()
    plainText = ""

    # Loop through each character in cipherText
    # Find position in key string
    # Concatentate parallel position in alphabet
    for ch in cipherText:
        idx = key.find(ch)
        plainText = plainText + alphabet[idx]

    return plainText

# Simple Test
originalMessage = "the quick brown fox"
testKey = "zyxwvutsrqponmlkjihgfedcba "
cipher = substitutionEncrypt(originalMessage, testKey)
plain = substitutionDecrypt(cipher, testKey) # Test decrypt function
print("Original:  ", originalMessage)
print("Ciphertext:", cipher)
print("Decrypted: ", plain)
```

Results

```
Original:   the quick brown fox
Ciphertext: gsv jfrxp yildm ulc
Decrypted:  the quick brown fox
```