CSCI 151 Fall 2018
Homework 2

For example, suppose the original file contains "`TEST TEXT`" and your password is "`pass`", and the keyfile contains the values as shown below:

Plain text	T (84)	E (69)	S (83)	T (84)	(32)	T (84)	E (69)	X (88)	T (84)
Password (repeated as needed)	p (112)	a (97)	s (115)	s (115)	p (112)	a (97)	s (115)	s (115)	p (112)
(sum of 2 chars) % 255	196	166	198	199	144	181	184	203	196
Encrypted text (from keyfile)	24	247	37	30	46	232	129	125	24

	144		166		181	182	183	184		196	197	198	199		203	
...	46	...	247	...	232			129	...	24		37	30	...	125	...

Encrypted text	24	247	37	30	46	232	129	125	24
Index from key file	196	166	198	199	144	181	184	203	196
Password (repeated as needed)	p (112)	a (97)	s (115)	s (115)	p (112)	a (97)	s (115)	s (115)	p (112)
(index – pass char) % 255 -> original	T (84)	E (69)	S (83)	T (84)	(32)	T (84)	E (69)	X (88)	T (84)

The encryption and decryption algorithms require that characters from the keyfile must be looked up in 2 different ways:

- know the index – get the character (for encryption)
- know the character – get the index (for decryption)

It is recommended (but not required) that you store the keyfile data in both ways, in 2 character arrays, each of length 254. Since the keyfile contains 254 distinct values, which are 0 to 253, this is possible. Suppose the keyfile was just 8 characters long, containing the values 0 to 7: `43756201`, you would create 2 arrays, like this:

0	1	2	3	4	5	6	7
4	3	7	5	6	2	0	1

0	1	2	3	4	5	6	7
6	7	5	1	0	3	4	2

Newline characters are even more troublesome for this program than the last one, and you must keep them in mind. A filename cannot have a newline at the end; the password should not include a newline either.