Programming Assignment 1 – algorithm

If your intent is to create a file without a LF, how do you create the 16 byte file without an end LF?

```
springfield> cat x.txt
abcdefghijklmnop
springfield> hexdump x.txt
0000000 6261 6463 6665 6867 6a69 6c6b 6e6d 706f
0000010 000a
0000011
```

This ends up with 17 bytes.  You have to do the following:

```
springfield> echo -n "lu" > lu.txt
springfield> hexdump -C lu.txt
00000000  6c 75                                             |lu|
00000002
springfield>
```

ok, let's look at what you should have
Let's take the file lu
That should be:

756c – with a size of 2

```
hexdump -C lu.txt
00000000  6c 75                                             |lu|
00000002
springfield>
```

And our key file is abcdefghijklmop

```
springfield> hexdump -C key.txt
00000000  61 62 63 64 65 66 67 68  69 6a 6b 6c 6d 6e 6f 70  |abcdefghijklmnop|
00000010
```

```
springfield> hexdump -C out.txt
00000000  f1 17 ee e5 ef e7 ec e9  ed eb ea e8 e6 e4 e2 0d  |................|
00000010
springfield>
```

Let's see if it worked.

6c75 0000 0000 0000 0000 0000 0000 0000
becomes
6c75 8181 8181 8181 8181 8181 8181 8181
with padding.

Now let's xor them.
6c75 8181 8181 8181 8181 8181 8181 8181
6162 6364 6566 6768 696a 6b6c 6d6e 6f70

0D17 E2E5 E4E7 E6E9 E8EB EAED ECEF EEF1
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

OK, now for byte swapping. We swap when the byte in the keyfile
is even we do not swap on odd.

The key is above.

Start = 0, end = 15
key = a (61) odd, swap bytes 0 and 15. start=1, end=14
key = b (62) even start = 2
key = c (63) odd, swap bytes 2 and 14. start = 3, end = 13
key = d (64) even – start = 4
key = e (65) odd, swap bytes 4 and 13, start=5, end=12
key = g, swap bytes 6 and 12
key = I, swap bytes 8 and 11

This gives us:
F117 EEE5 EFE7 ECE9 EDEB EAE8 E6E4 E20D

Which matches the expected output
springfield> hexdump -C out.txt
00000000 f1 17 ee e5 ef e7 ec e9 ed eb ea e8 e6 e4 e2 0d |.....|
00000010