Pledge: *I pledge my honor that I have abided by the Stevens Honor System.* - **Eric Altenburg**

Problem 3: Crypt-analyze this!

The ciphertexts and their corresponding plaintexts are as follows:

c_1	2d0a0612061b0944000d161f0c1746430c0f0952181b004c1311080b4e07494852
p_1	Testing testing can you read this
c_2	200a054626550d051a48170e041d011a001b470204061309020005164e15484f44
p_2	Yep I can read you perfectly fine
c_3	3818101500180b441b06004b11104c064f1e0616411d064c161b1b04071d460101
p_3	Awesome one time pad is working
c_4	200e0c4618104e071506450604124443091b09520e125522081f061c4e1d4e5601
p_4	Yay we can make fun of Nikos now
c_5	304f1d091f104e0a1b48161f101d440d1b4e04130f5407090010491b061a520101
p_5	I hope no student can read this
c_6	2d0714124f020111180c450900595016061a02520419170d1306081c1d1a4f4601
p_6	That would be quite embarrassing
c_7	351a160d061917443b3c354b0c0a01130a1c01170200191541070c0c1b01440101
p_7	Luckily OTP is perfectly secure
c_8	3d0611081b55200d1f07164b161858431b0602000454020d1254084f0d12554249
p_8	Didnt Nikos say there was a catch
c_9	340e0c040a550c1100482c4b0110450d1b4e1713185414181511071b071c4f0101
p_9	Maybe but I didnt pay attention
c_{10}	2e0a5515071a1b081048170e04154d1a4f020e0115111b4c151b492107184e5201
p_{10}	We should really listen to Nikos
c_{11}	370e1d4618104e05060d450f0a104f044f080e1c04540205151c061a1a5349484c
p_{11}	Nah we are doing fine without him

To successfully decipher this, I used crib dragging which has been implemented many times by other programmers online. Here is the website I used specifically: https://toolbox.lotusfa.com/crib_drag/. This website requires that I input 2 ciphertexts and then some crib words. The latter is rather difficult without the context of the ciphertexts and relies on some probability with you knowing which words to use for the crib; however, given this circumstance, Alice and Bob were likely talking about Nikos since he thinks they were planning behind his back. So with this information, ”Nikos” was the initial crib I used.

However, the first two plaintexts do not contain ”Nikos” so keeping c_1, I cycled through the other ciphertexts until I found that c_4’s plaintext p_4 contained ”Nikos” in it. In the plaintext p_1 ” Nikos ” mapped to ”u read” and so my next crib phrase was ”you read” and this gave me ”of Nikos” in p_4. Then using ” fun of Nikos ” I was able to continually build upon each of the two plain texts until I arrived at their full messages as seen above.

After decoding p_1 and p_4, I then used this XOR calculator http://xor.pw/# to obtain the key. I changed the input 1 to be ASCII (base 256), input 2 to be Hex (base 16), and the output to be ASCII (base 256). With p_1 as input 1 and c_1 as input 2, my output was $k = youfoundthekey!congratulations!!!$. Using k as my input 1, I then changed the input 2 to be the various ciphertexts c_2, c_3, and c_5 through

c_{11} to which I found their respective plaintexts. I did not need to run the key through c_4 because I already found it's respective plaintext.