Decrypted Message -> “Time is an illusion. Lunchtime doubly so.
- Douglas Adams”

Recovered Key -> 0111001101110100 (binary); 29556 (decimal)

Code Summary -> Using the passphrase “Hopes and dreams of a million years” and an encryption key size of 16 bits, I first reduced the passphrase down to a bit array for use as an initialization vector, which is needed to XOR with the first block of the text. I loaded in the file with the encrypted message and created a hexstring from it, then differentially XORd each block over range [0, # of blocks in message]. I then retrieve the text from the bit vector and close the file before returning.

My main statement uses a for loop that iterates through values 0 to 65535, which is the max int value for a key space of 16 bits. It then creates a bitvector with the value of the iterative variable and a size of 16 bits. If “Douglas Adams” is in the decrypted message, it prints the message and key, then breaks.

CITATION -> code snippets from DecryptForFun.py (Lecture 2 code) were used in my implementation