

For HW1 we have created a program that uses brute force attack to find the right key. The Brute Force attack will check through 2^{16} key spaces. We checked through `range(0,2^16)` and then changed to a bit vector, used the decryption method given in `DecryptForFun.py` and checked whether the string "Mark Twain" appeared in the file.

```

from cryptBreak import *
if __name__ == '__main__':
    allBit = tuple(range(0, 2**16))
    def keygen(): # generates a key to try
        for keyTry in allBit: #iterates through the set of keys
            plain = cryptBreak('encrypted.txt', keyTry)
            if "Mark Twain" in plain: #tests decoded message for correctness
                print("Encryption Broken!")
                print("Key: ",keyTry)
                print("Message: ",plain)
                FILEOUT = open('decrypted.txt', 'w') # (d)
                FILEOUT.write(plain) # (e)
                FILEOUT.close()
                break

```

```

'''
Homework Number: 1
Name: Sneha Mahapatra
ECN Login: mahapat0
Due Date: 01/23/20
'''

from BitVector import *
PassPhrase = "Hopes and dreams of a million years"
BLOCKSIZE = 16
numbytes = BLOCKSIZE//8

def cryptBreak(ciphertextFile, key):
    FILEIN = open(ciphertextFile) # (J)
    encrypted_bv = BitVector(hexstring=FILEIN.read())
    bv_iv = BitVector(bitlist=[0] * BLOCKSIZE) # (F)
    for i in range(0, len(PassPhrase) // numbytes): # (G)
        textstr = PassPhrase[i * numbytes:(i + 1) * numbytes] # (H)
        bv_iv ^= BitVector(textstring=textstr) # (I)
    key_bv = BitVector(bitlist=[0] * BLOCKSIZE) # (P)
    key_bv = BitVector(intVal=key, size=16)
    msg_decrypted_bv = BitVector(size=0) # (T)
    previous_decrypted_block = bv_iv # (U)
    for i in range(0, len(encrypted_bv) // BLOCKSIZE): # (V)
        bv = encrypted_bv[i * BLOCKSIZE:(i + 1) * BLOCKSIZE] # (W)
        temp = bv.deep_copy() # (X)
        bv ^= previous_decrypted_block # (Y)
        previous_decrypted_block = temp # (Z)
        bv ^= key_bv # (a)
        msg_decrypted_bv += bv # (b)
    outputtext = msg_decrypted_bv.get_text_from_bitvector() # (c)
    return outputtext

```

It is my belief that nearly any invented quotation, played with confidence, stands a good chance to deceive.

- Mark Twain

Encryption Broken!

Key: 25202

Message: It is my belief that nearly any invented quotation, played with confidence, stands a good chance to deceive