Explanation for Problem 1:

For encrypt for image I first modify the encryption function from hw4, I put the key generated outside of the encrypt function and into ctr_aes_image image. This way it can greatly decrease the run time for encrypt the image. I then start the ctr_aes_image function by reading the image file, and then encrypt the iv, which is the initialization vector, and then xor with encryted iv then write it to file. After this, I updated the iv by adding one to its integer and then do the same process all over again.

Explanation for Problem 2:

For problem two I basically followed the instructions in lecture 10 on page 36 to implement the function. I also first generate the round key at the beginning of the function, and then I encrypt the dt input and xor with the v0, which is the initialization vector. Then I encrypt the output before again to get the Rj. I then write it to the file and this became the first generated number sequence. This Rj then xor with the first encrypted dt, then got encrypted again to get the vj+1.

Encrypted image from problem 1

the 5 pseudo-random numbers generated from problem 2:

331374527193731622526773163027689011175

262633037080229609278739248627548891 87

621388110439928640615094882415799550 8

317525806849049200816126045738729418009

240080400546264647934751409092776671804