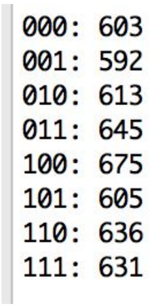
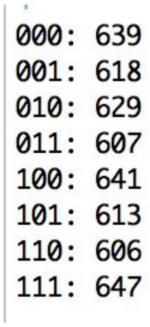
CS 6058 project 1 report

Zheng Liu

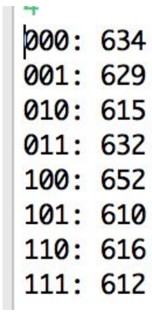
1. For the distribution of the key, since each digit is randomly generated between 1 and 0, the following is a couple of run for the 5000 keys.



```
000: 603
001: 592
010: 613
011: 645
100: 675
101: 605
110: 636
111: 631
```

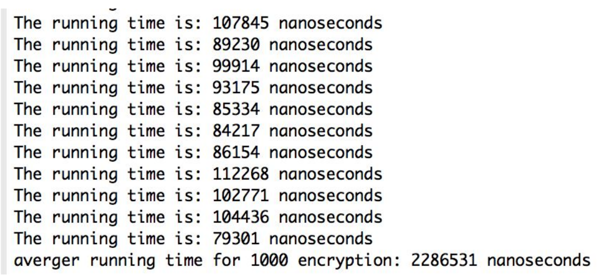


```
000: 639
001: 618
010: 629
011: 607
100: 641
101: 613
110: 606
111: 647
```

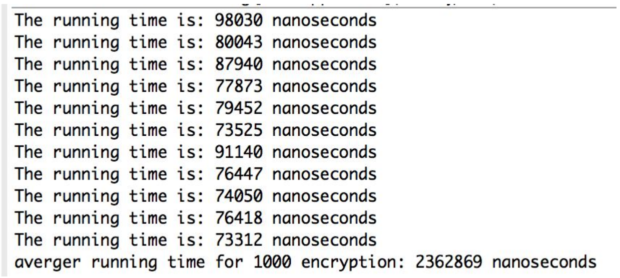


```
000: 634
001: 629
010: 615
011: 632
100: 652
101: 610
110: 616
111: 612
```

Since the key is 3 digits long which means that each key has ½^8 chance, and ⅛ of 5000 is around 625, from the screenshot, we can see that the real world is around 625 for each key, which means that the key is uniformly distributed.



```
The running time is: 107845 nanoseconds
The running time is: 89230 nanoseconds
The running time is: 99914 nanoseconds
The running time is: 93175 nanoseconds
The running time is: 85334 nanoseconds
The running time is: 84217 nanoseconds
The running time is: 86154 nanoseconds
The running time is: 112268 nanoseconds
The running time is: 102771 nanoseconds
The running time is: 104436 nanoseconds
The running time is: 79301 nanoseconds
averger running time for 1000 encryption: 2286531 nanoseconds
```



```
The running time is: 98030 nanoseconds
The running time is: 80043 nanoseconds
The running time is: 87940 nanoseconds
The running time is: 77873 nanoseconds
The running time is: 79452 nanoseconds
The running time is: 73525 nanoseconds
The running time is: 91140 nanoseconds
The running time is: 76447 nanoseconds
The running time is: 74050 nanoseconds
The running time is: 76418 nanoseconds
The running time is: 73312 nanoseconds
averger running time for 1000 encryption: 2362869 nanoseconds
```

After run 2000 encryption, we can see that the average running time for the encryption is around 1300000 nanoseconds.

2. I developed that program in Mac OS with eclipse and Java, and I have tested the program under window. There are three components: key plaintext and ciphertext, the relation among them is XOR, for example,
key XOR ciphertext = plaintext
Key XOR plaintext = ciphertext
To make this work, I convert plain text to binary number by use ascii code and convert decimal number to binary number.
For generating new keys, for each digit of the key there are only two possible opinion which are 1 and 0, so that a simple for loop is enough for random generating the new key for a given length.