**Project 5**

**Question 1 (FSK 9.1). Investigate the random number generators built into your favorite programming language. Would you use these random number generators for cryptographic purposes?**

Python provides a PRNG (Psuedo Random Number Generator) within the module random. The algorithm behind this module is the Mersenne Twister Generator which is widely used among many other programming languages. Although this is a popularly used PRNG for common applications, it is not acceptable for cryptographic use. It is not cryptographically secure because after observing a sufficient amount of iterations it is possible to predict future iterations.

Python also provides hardware random number generators or (TRNG) which generate random numbers based on unpredictable natural phenomena and then introduced to the computing system. Two good choice for CSPRNG’s(cryptographically secure Psuedo number generator) would be the secrets module and the os.urandom module(provided by the random module). They work on the entropy available from the system which is harvested from the hardware.

For CSPRNG, you can use a PRNG(deterministic) with a truly random seed, but the period must be large enough for it to be feasibly secure. This is efficient and practical if the PRNG’s algorithm is acceptable for cryptographic use. [Yarrow and Fortuna]

**Question 2 (FSK 9.4). What are the advantages of using a PRNG over an RNG? What are the advantages of using a RNG over a PRNG?**

The advantage of using a PRNG over a RNG(TRNG) is that it is more efficient and the results are reproducible, given the same seed value. The advantage of using a TRNG is that the numbers are non-deterministic (truly random), which would be ideal in situations such as key generations.

**Question 3 (FSK 10.2). Compute 13635 + 16060 + 8190 + 21363 (mod 29101) in two ways and verify the equivalence: by reducing modulo 29101 after each addition and by computing the entire sum first and then reducing modulo 29101.**

Method 1 –

13635 + 16060 + 8190 + 21363 = 59248 mod 29101 = **1046**

Method 2-

13635 mod 29101 = 13635

13635 +16060 mod 29101 = 594

594 +8190 mod 29101 = 8784

8784 + 21363 mod 29101 = **1046**

**Question 4 (FSK 10.4). Is {1, 3, 4} a subgroup of the multiplicative group of integers modulo 7? Justify your answer.**

No, it is not because 3\*4 mod 7 = 5 which is not part of the subgroup.

**Question 5 (FSK 10.5). Use the GCD algorithm to compute the GCD of 91261 and 117035.**

117035 = 91261 \* 1 + 25774

91261 = 25774 \*3 + 13939

25774 = 13939 \* 1 + 11835

13939 = 11835 \* 1 + 2104

11835 = 2104 \* 5 + 1315

2104 = 1315 \* 1 + 789

1315 = 789 \* 1 + 526

789 = 526 \* 1 +263

526 = 263 \* 2 + 0

GCD is 263

**Question 6 (FSK 10.6). Use the extended GCD algorithm to compute the inverse of 74 modulo the prime 167.**

74\*x = 1 mod 167

x = 74-1 mod 167

167 = 74(2) + 19

74 = 19(3) +17

19 = 17(1) +2

17 = 2(8) +1

1 = 17-8(2)

= 17-8(19-17) = **17(9) -19(8)**

= (74-19(3))\*9 – 19(8) = **74(9) – 19(35)**

**=** 74(9) – (167-74(2))\*35 = **74(79) + 167(-35)**

\*167(-35) mod 167 = 0\*

1 = 74(79) + 0 mod 167

1 = 74(79) mod 167

**79 is inverse of 74!**

**Question 7. Compute 27573 (mod 569) using Fermat’s Little Theorem.**

573 = 568 + 5

(27568) \* 275 mod 569

1 \* 275 mod 569 = 434 mod 569