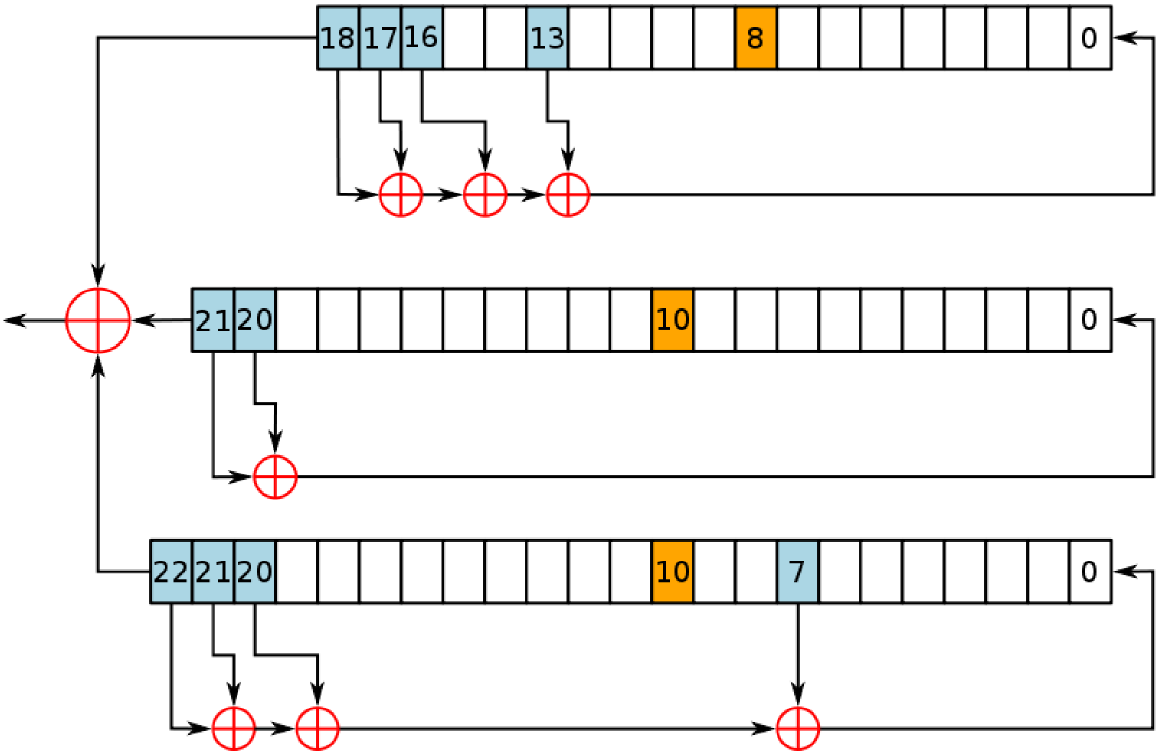
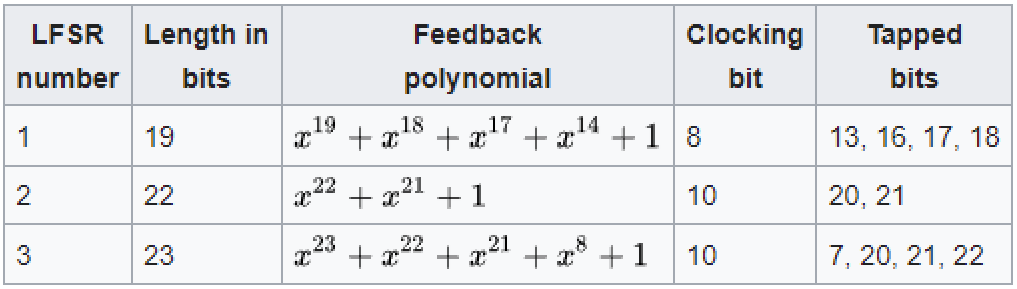
ISM Secure Application Programming

Assignment 1 - Your own A5 PRNG

Implement a Java solution for the A5 PRNG (http://en.wikipedia.org/wiki/A5/1) based on a Linear Feedback Shift Register covered by the Crypto course and described in the next image





LFSR number	Length in bits	Feedback polynomial	Clocking bit	Tapped bits
1	19	$x^{19} + x^{18} + x^{17} + x^{14} + 1$	8	13, 16, 17, 18
2	22	$x^{22} + x^{21} + 1$	10	20, 21
3	23	$x^{23} + x^{22} + x^{21} + x^{8} + 1$	10	7, 20, 21, 22

Source: https://en.wikipedia.org/wiki/A5/1

Functional requirements

1. **The LFSR will be initialized based on an initial 8 char password**. You can choose that value and set it in the source code
2. The LFSR will be used to generate a sequence/byte array **of any size**. For testing you should test it in main with at least 2 different sizes
3. The output should be displayed in hexadecimal
4. The sequence should be used also to generate pseudo random integers. Implement a method that receives a byte array and it will generate the equivalent number of integers (if the input byte array size is not proper throw an exception)
5. The implementation can be based on a byte array, integers or other types, but the method that generates the pseudo random sequence of bytes must have the next signature

   byte[] A5Generator(String password, int sequenceNoBytes) {}

6. You can implement any number of additional functions

Evaluation

1.5pts – the solution contains the required methods (A5Generator() and the byte array to integer function)

2.0 pts – implement the A5 PRNG using bitwise operations

0.5 pts – test it in main on order to prove that it generates a pseudo sequence (display 2 minimum sequences with different sizes)

1 pts – use the first x bytes from the sequence to generate an integer sequence and display it.

-2pts if the solution will generate the initial password in the sequence

-**3pts** if the solution implements the A5 generator but the implementation does not comply with the requirements (the A5Generator function, the byte array to integer function, etc)