CS 355, Fall, 2019, Project 2

Write a program in Java to simulate a Linear Feedback Shift Register.

See `www.cs.purdue.edu/homes/ssw/cs355/x07.pdf` for definitions.

Let the register have n bits. Assume $n < 100$. Let the bits in the register be numbered r_0, r_1, …, r_{n-1} from left to right. Let the tap bits be numbered t_0, t_1, …, t_{n-1} from left to right. The bits in the register shift one bit position to the right whenever a key bit is needed. The bit that emerges from the right end is the next key bit. The new bit at the left end is the XOR of several bits in the register before the shift, with the tap bits t_i that are 1 telling which bits r_i are included in the XOR. This means that the new bit for the left end is

$$\bigoplus_{i=0}^{n-1} r_i t_i = \sum_{i=0}^{n-1} r_i t_i \bmod 2,$$

where $\oplus$ means XOR (sum modulo 2) and the multiplication $r_i t_i$ is modulo 2, which is the same as a logical AND of bits.

The input to your program will be (1) an integer n with $0 < n < 100$, (2) a string of n bits representing the initial contents, (3) a string of n bits representing the tap vector. and (4) a bit string representing the plaintext or ciphertext. The 4 items are separated by newline characters, that is, each of the four input items is on its own line. (The length of the bit string in (4) will be < 200 bits.)

Your program should simulate the LFSR to produce a key stream of the same length as the plaintext or ciphertext. Then it should write (1) the key stream (for debugging and grading purposes) and (2) the XOR of the key stream and the plaintext or ciphertext, which represents the ciphertext or plaintext.

Example:

If the input to your program is:

```
4
1100
0011
10101111001001111011
```

then your program should write:

```
00110101111000100110
10011010110001011101
```

exactly as shown because Vocareum grades by comparing character strings.

Name your program lfsr.java. Submit your program to Vocareum by 11:59 PM on the due date. It will be compiled and run ten times with ten different secret input sets, each worth 10 points.