Assignment 2: Random Linear Fountain Codes

The last assignment is split into two parts. The first part should simulate the sender of a data fountain, in the second part the receiver is added in addition.

1) Simulation of the sender side: The user should be allowed to enter a string. This string of 8-bit characters is interpreted as a long bit vector resp. it is the data to be transmitted. Group this bit vector into a matrix. The width of the matrix should be 8, the height should correspond to the number of characters the user entered. Thus, you see the bit-representation of one character per line in your matrix. Also display this matrix to the user. Now the program should draw bits from a random bit source and produce a stream of packets from the user input. Store a sufficiently large number of packets from this fountain. As you generate packets, the random bit vector and the resulting packet should be displayed to the user as a bit-string.

The first part of the assignment is due till January 9, 2009 sharp (no extension).
Assignment 2: Random Linear Fountain Codes

2) Given that your user has entered \( n \) characters, the above application should have stored \( n \) packets from the fountain. Now, try to reconstruct the original string. After each iteration you should display the state of your program in a meaningful way so that a user can see what you are doing. If you encounter a zero-only line in your transform matrix you need to draw another packet (and random bit vector) from your fountain and continue to reconstruct the original data until you succeed. How often can you cope with \( n \) packets, how often do you need 1, 2 or more additional packets?

The second part of the assignment is due till January 23, 2009.

Your application can be a text-mode only application. Write in Java, C or C++. Except for the required character in- and output you should do all programming by yourself. Your application need not be optimal but self-made. You may work in groups of at most 4 people. Keep your program as simple as possible. You can store bits as integers or alike to you convenience. Go for static arrays rather than dynamic data structures if helpful.

If working properly, the first part scores 40% of this assignment, the second part scores 60%.
Assignment 2: Random Linear Fountain Codes

3) What is the average number of additional packets for different values for $n$? 
   (again, simulate for $n=8, 16, 64$)

Extend the application from the last exercise:

- It should be possible to define a (short) message.
- This message should be XORed with random vectors generating a sequence of packets.
- After a sufficient number of packets have been generated, the original message should be reconstructed.