A. Let $X \sim \text{Binom}(8, p)$. In class we figured out the PMF of $Y = (X - 3)^2$. Find $E(Y)$.

B. Let $X \sim \text{Binom}(n, p)$ and $Y \sim \text{Binom}(m, p)$. Compute $E(X + Y)$ in two different ways. (If you can't imagine two ways, then see the hint on the back of this sheet.)

In class, we are about to compute $E(X)$ for $X \sim \text{HGeom}(m, b, n)$. The argument is a bit confusing. The purpose of this next exercise is to independently check the case where $n = 2$.

C. Let $X \sim \text{HGeom}(m, b, 2)$. Compute $E(X)$ manually — without using the general result, which we don't know yet. Simplify as much as possible.

When data are stored in a computer or transmitted between computers, they are expressed as sequences of bits, with each bit being 0 or 1. For example, the seven-bit sequence 1000010 could express the number 66 or the letter B, depending on the context. (Don't worry how.)

Unfortunately, when data are stored or transmitted, there is a chance for errors to creep into them. For example, a small electrical disturbance might cause the fifth bit above to flip from 0 to 1, changing the sequence to 1000110, meaning 70 or F.

The simplest way to detect such errors is to append a parity bit. The parity bit is chosen to be either 0 or 1, so that the entire bit sequence has an even number of 1s. In the running example, the seven-bit sequence 1000010 has an even number of 1s, so I append a 0 to make the eight-bit sequence 10000100, which still has an even number of 1s. Then I send the eight-bit sequence to you. If the fifth bit gets flipped during storage or transmission, so the sequence becomes 10001100, then there are no longer evenly many 1s, so you know that an error has occurred, and you ask me to re-send the message.

As far as error detection goes, parity bits are quite primitive. One of their flaws is that they cannot detect whether two errors have happened. For example, if the fourth and fifth bits get flipped, so the sequence becomes 10011100, then there is still an even number of 1s, so you have no reason to believe that any errors have occurred.

To clarify, we am not interested in the case where a single bit flips twice. For example, suppose that I send you 10000100. During transmission, the sequence changes to 10001100 and then back to 10000100. So you receive 10000100, which is the correct bit sequence. In this case, we say that no error has occurred.

D. Suppose that I store or send a seven-bit sequence with a parity bit, for a total of eight bits. Errors can happen in any of these eight bits, independently of each other. The rate of error in any single bit is 0.001. When you retrieve or receive the eight-bit sequence, it has even parity. What is the probability that the sequence is erroneous?

Hint for B: One way is linearity of expectation. Another way is based on the story of what the binomial distribution means.