Analysis: Punched Cards

This problem required us to print out ASCII pictures of punched cards of different sizes. Although the problem and solution are relatively straightforward, we need to know how to read from stdin and write to stdout. Examples of this can be found in the "Coding" section of the FAQ.

Because the dimensions of the punched cards we need to print are so small, one option would be to store all the ASCII pictures in an array and just print the requested cards. However, we can make use of nested <u>for-loops</u> to generate the art for any sized punched card.

We can use a loop to print out the punched card one line at a time and use another loop to print the characters in each line. The odd-numbered (1-indexed) lines alternate between (+) and (-), and the other lines alternate between (+) and (.). We can check the line number and column number (mod 2) to see which character we should print. The only exception is when both the line number and column number are ≤ 2 . In this case, we always print a period (.).

As some of you may have already discovered, for this year's qualification round, you could submit solutions in *Punched Card Python*. The following is a solution to *Punched Cards* written using *Punched Card Python*:

