P. Sloth-out (500 points)

Surprise task!

Sloths like to play breakout but they are too lazy and tired (after several hours of programming!) to interact with such a wild game, so to please them you'd better design a game level for passive players where the ball can bounce around between blocks for as long as possible before falling off the bottom.

Rules

The playing field is 40 pixels wide and 60 pixels tall. The ball is 1 pixel by 1 pixel, and moves pixel by pixel (no fractions). Breakable blocks are in the top 40x40 pixels of the field. Each block is 4x4 pixels large, so there is space for 10x10 blocks.

The bottom left pixel of the field has coordinates `X=0, Y=0`, the top right pixel has `X=39, Y=59`. The initial position of the ball is `X=10, Y=0`.

The ball has a velocity of two components: `VX, VY`. Both are either -1 or 1. The initial velocity of the ball is `VX=1, VY=1`.

The ball moves iteration by iteration. The first iteration has number 1.

In odd iterations (including the first one), the ball tries to move horizontally: `NX := X + VX`. If the new position would cross the edge (`NX < 0` or `NX > 39`) then the ball doesn't move, and `VX` is inverted (`VX := 0 - VX`). If the new position would intersect a block, then the block is destroyed, the ball doesn't move, and `VX` is inverted. Otherwise, the ball moves: `X := NX`.

In even iterations, the ball tries to move vertically: `NY := Y + VY`. If the new position would cross the top edge (`NY > 59`), the ball doesn't move, and `VY` is inverted. If the new position would cross the bottom edge (`NY < 0`), the game ends. If the new position would intersect a block, then the block is destroyed, the ball doesn't move, and `VY` is inverted. Otherwise, the ball moves: `Y := NY`.

Output format

Submit a design for a breakout level in a text file of 10 lines, each line 10 characters long. Blocks are marked by '1' characters, lack of blocks are '0' characters. The first line specifies the topmost row of blocks (`Y = 56 .. 59`), from left to right.

Scoring

This is a scaled problem. The team who submits a level in which the ball lasts the most iterations gets **500 points**; other teams are scaled relatively to the best submission: `500*(TEAM/BEST)`^{2}. Note that there is a **2 minutes delay** before you can upload another submission!