

A Jigsaw Puzzle

This is a fun little program that cuts a picture into pieces, splatters them around the screen, then lets the user put them back together by pushing them around with the mouse. Here are the Plain English type definitions:

A puzzle is a thing with a box, a picture and some pieces.
A piece is a thing with a box and a picture.

And here is the main routine:

To run:
Start up.
Create a puzzle from “c:\g4g articles\jigsaw\maga.jpg”.
Display the puzzle.
Show the arrow cursor.
Loop.
Deque an event.
If the event is nil, break.
If the event’s kind is “key down”, break.
If the event’s kind is not “left click”, repeat.
Track the mouse on the puzzle.
Repeat.
Destroy the puzzle.
Shut down.

These are the Plain English routines that create a puzzle:

To create a puzzle from a path:
Allocate memory for the puzzle.
Make the puzzle’s box 8 inches by 8 inches.
Center the puzzle’s box in the screen’s box.
Fetch the puzzle’s picture from the path.
Resize the puzzle’s picture to 8 inches by 8 inches.
Center the puzzle’s picture in the puzzle’s box.
Cut the puzzle into pieces.
Splatter the puzzle’s pieces.

To cut a puzzle into pieces:
Draw the puzzle’s picture.
Put the puzzle’s left-top into a spot.
Loop.
Allocate memory for a piece. Append the piece to the puzzle’s pieces.

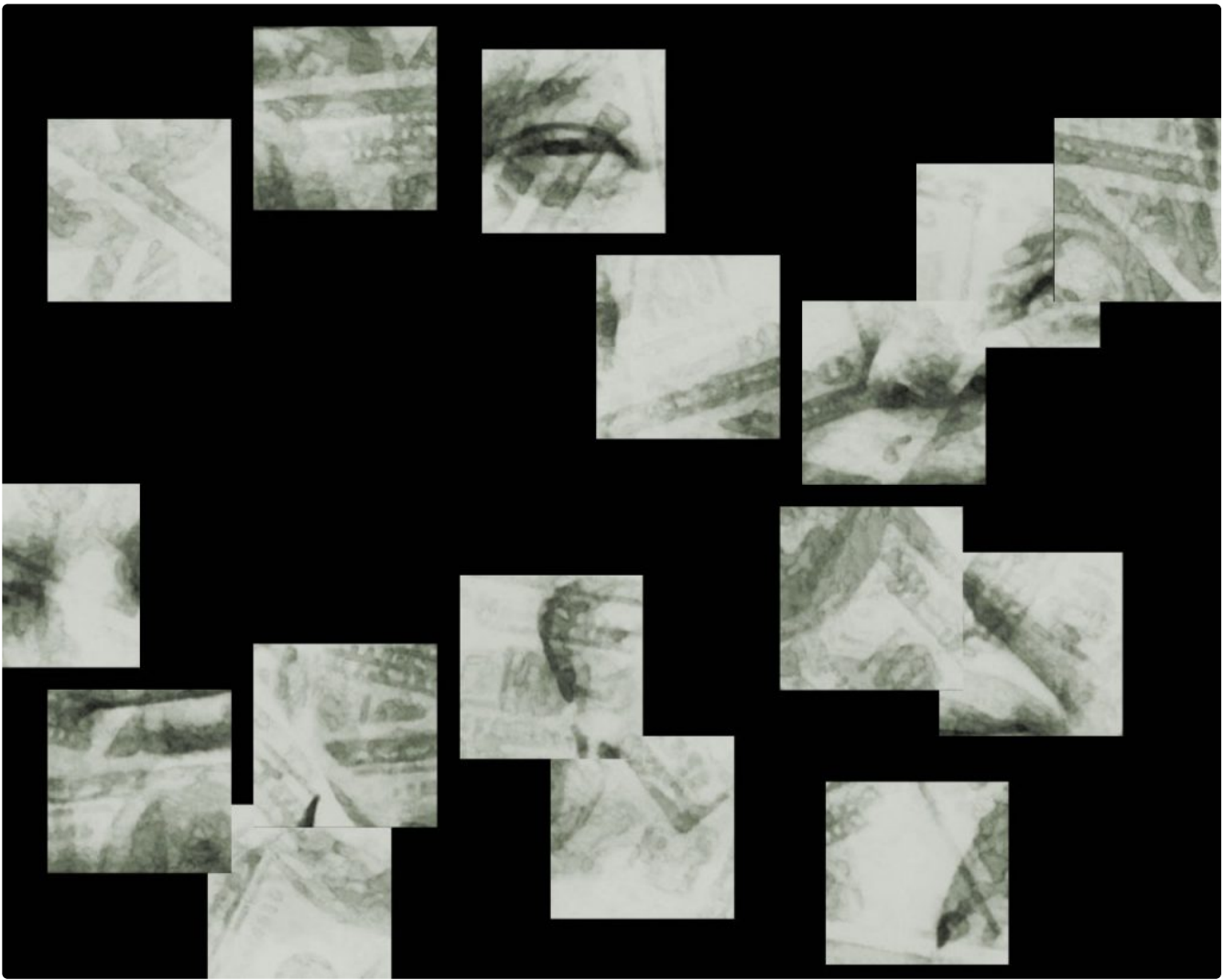
Put the spot and the spot plus 2 inches into the piece's box.
Extract the piece's picture using the piece's box.
Add 2 inches to the spot's left.
If the spot's left is less than the puzzle's right, repeat.
Put the puzzle's left into the spot's left.
Add 2 inches to the spot's top.
If the spot's top is the puzzle's bottom, break.
Repeat.

To splatter some pieces:
Make a box 1 inch smaller than the screen's box.
Loop.
Get a piece from the pieces.
If the piece is nil, break.
Pick a spot anywhere in the box.
Round the spot to the nearest multiple of 1/4 inch.
Center the piece's box on the spot.
Center the piece's picture on the spot.
Repeat.

This is how we draw a puzzle's pieces on the screen:

To display a puzzle:
Clear the screen without refreshing.
Loop.
Get a piece from the puzzle's pieces.
If the piece is nil, break.
Draw the piece's picture.
Repeat.
Refresh the screen.

And this is typical of how the splattered pieces of a puzzle look:



These are the routines we use to track the user's mouse movements:

To track the mouse on a puzzle:

Find a piece of the puzzle given the mouse's spot.

If the piece is nil, exit.

Loop.

If the mouse's left button is up, exit.

Put the mouse's spot into a spot.

Round the spot to the nearest multiple of 1/4 inch.

Center the piece's box on the spot.

Center the piece's picture on the spot.

Display the puzzle.

Repeat.

To find a piece of a puzzle given a spot:

Get the piece from the puzzle's pieces (backwards).

If the piece is nil, exit.

If the spot is not in the piece's box, repeat.

Note that we “snap” the pieces to a 1/4-inch grid so they'll fit together easily. Note also that we search for pieces in the opposite order they are drawn (ie, backwards) so that the front-most piece gets selected first.

This is what the pieces of our sample puzzle look like once they've been properly arranged:



It's a painting I did a while back. I called it "Make America GREEN Again."