Problem 3: Diagram a correlating branch predictor that uses 3 bits of global information, 5 bits of local information, 3-bit saturating counters. Include the dimensions of all the machine data structures. Diagram the state machine. Give brief pseudo code for how this operates.

Index a 256 x 3-bit RAM using the low 5 bits of the PC concatenated with 3 bits from the branch history shift register. The shift register has as input the branch direction. The book draws this as a two dimensional table. Same thing.

On decode, combine the 8 bits as above, index the prediction table, read the value and predict based on the most significant bit.

Retain prediction table index into the execute stage. At that point, update the table entry according to the saturating counter state-transition-diagram. Write the result to the table. Shift the taken/not-taken bit into the shift history.

There are two reasonable state machine. Both have state encodings going from 000 at the bottom to 111 at the top. The one on the right provides some hysteresis. You don't get stuck toggling between 011 and 100. Even a branch that alternates between taken and not-taken will get predicted right about half the time.