1. (5 points) Why might it be useful to get in the habit of writing `toString()`, `clone()` and accessors for every class when you first write them?

2. (5 points) Is machine intelligence possible in principle? Explain your answer.

3. (5 points) Why might embodiment (of some stripe) be necessary for learning/intelligence?

4. (5 points) Is the Turing Test a good measure of machine intelligence? Explain.

5. (5 points) What is the formula for the perceptron learning rule?

6. (3 points) Given a 20x20 perceptron where each weight is an `int`, how many different possible states does the weights matrix have?

7. (6 points) Why can't a ptron correctly categorize XOR?
8. (8 points) Draw a network of McCollough-Pitts neurons that can (see previous); include values for wts with \( \theta = 5 \). Make clear why it gets each right with a table, one line per input.

9. (5 points) Write `clone()` for this class (make sure it does a deep copy).

    ```java
    class Foo {
        static final int NUM_WTS = 6;
        int x;
        int[] wts;

        Foo() {
            wts = new int[NUM_WTS];
        }
    }
    ```

10. (6 points) Assume you are implementing rote learning. Write a Learner class with two static methods: `void neverGoHereAgain(Board losingBoard)`, and `void trimLosers(BoardList list)`
11. (8 points) Here’s an interface:

```java
public interface Minimaxible {
    public MinimaxibleList generateLegalNextBoards();
    public boolean gameOver();
    public int staticValue();
    public Move getMove();
    public void setMove(Move m);
    public void setValue(int v);
    public int getValue();
}
```

..and minimax with two deletions `&&` and `&`. Fill in the correct parameters.

class Minimaxer {

    public static Minimaxible bestBoard(Minimaxible currentBoard, int depth) {
        return max(currentBoard, depth);
    }

    private static Minimaxible max (Minimaxible currentBoard, int depth) {
        if (currentBoard.gameOver() || depth == 0) {
            currentBoard.setValue(                          );  <<<<<<<<< 1  
            return currentBoard;
        }
        MinimaxibleList list = currentBoard.generateLegalNextBoards();
        for (Minimaxible nextBoard: list) {
            nextBoard.setValue(                                );  <<<<<<<<< 2  
        }
        return maximumOf(list);
    }

12. (5 points) Assume you have a written a minimax program for some game that is so small you can generate the entire tree. Unfortunately, the first player always loses with best play; thus, if the static evaluator simply returns -100 for a loss and 100 for a win, every move has a value of -100, so your minimaxer just plays at random. How can you modify it so it will select the move that takes the longest to lose with best play?

13. (4 points) What is the law of uphill analysis and downhill synthesis?

14. (4 points) Merker argues that consciousness is a midbrain phenomenon. What are two of his arguments?

   i -
   ii -
15. (6 points) Here are two type 3 (love) vehicles, one with crossed, the other, uncrossed connections; the * is a source the detectors respond to. The detectors inhibit motors on the two wheels (not shown); the more input the detector gets the slower the motor turns. Draw their paths.

```
|        | C |
---|---|
uncrossed | C 
|        | C |
```

```
|        | C |
---|---|
crossed   | C |
|        | C |
```

16. (5 points) Genetic algorithms (GAs) have been characterized as "parallel stochastic Darwinian hill-climbers"; explain what this means.

17. (6 points) What are 3 independent mechanisms that help a GA increase average fitness?
   i -
   ii -
   iii -

18. (4 points) Describe how a GA implements prediction/correction

19. (5 points) What is a "headless chicken" GA?

20. (5 points xtra) What question should have been on this exam that wasn’t? Provide an answer to it!