Problem 1. Let $P$ be a set of 4 points: $A = (1, 2), B = (2, 1), C = (0, 1)$ and $D = (1, 0)$ where $A, B$ have color red, while $C, D$ have color blue. We want to find a plane that separates the red points from the blue points.

1. Convert the problem to $\mathbb{R}^3$ so that it can be solved by the Perceptron algorithm. Give the resulting dataset $P'$.

2. Execute Perceptron on $P'$. Give the equation of the plane that is maintained by the algorithm at the end of each iteration.

3. Convert the plane output by Perceptron back to the original 2d space to obtain a separation plane on $P$.

Problem 2. Let $P$ be a set of multidimensional points where each point is colored in either red or blue. We want to design an algorithm to achieve the following purpose:

- Either return a separation plane;
- Or declare that $P$ has no separation planes with a margin at least $\gamma$ (recall that the margin of a separation plane $\pi$ equals the minimum of the distances from the points of $P$ to $\pi$).

Note that your algorithm must still work even if $P$ is not linearly separable.

Problem 3. Describe how to solve the classification problem in Problem 1 by way of linear programming.

Problem 4. Prove that the reduction from linear classification to linear programming explained in the lecture is correct.

Problem 5. The figure below shows the boundary lines of 5 half-planes. Consider the execution of the linear programming algorithm we discussed in the class on these 5 half-planes. Recall that the algorithm starts by randomly permuting the boundary lines, and assume that the resulting permutation is exactly $\ell_1, \ell_2, ..., \ell_5$. The algorithm then processes $\ell_i$ in the $i$-th round, for $i = 1, ..., 5$, and at any moment maintains a point $p$ as the current answer. Explain which point $p$ is at the end of each round, starting from $i = 2$. 

```
\ell_1
\ell_2
\ell_3
\ell_4
\ell_5
```