Peter Maitland, Christopher Pauly, Matus Mikus, Petr Sramek

JOOSE lab group ??????

Task 1:

We have decided that the given design is not the most suitable one as it does not support easy extendibility and the complexity

of Board classes is too great. Thus, in our design, created using Strategy design pattern, MVC design pattern and Observer pattern,

we moved the fly method into a MoveBehaviour interface which then specifies the movements of either Bird or Grain. Then each Piece

implements Runnable and Observer so that we can run it in separate thread and update it with the status of other pieces.

Since Board has all the pieces it needs to be observable to allow one piece to know about all the others. Following these changes

we used Board as a Model which holds the data and we separated the View into a separate class called DisplayBoard. This way we

separated Model from a View as much as Java allowed us to.

Task 2:

<img>

Task 5:

Our design allowed us to add a new behaviour easily since we have an interface for that. So only by creating new class implementing

that interface and adding an appropriate button we extended the behaviour. This was mainly thanks to usage of Strategy design pattern.