You have to develop a new domain that consists in the extension of the Parity domain you have seen during the lectures about non-relational numerical domains (slides 30-38) with a special element used to represent the singleton (like what happens in the Sign domain). In this context, the even numbers do not represent anymore the element zero. This domain is formalized above.

In addition, you have to implement the Interval domain following the definitions provided during the lectures (56-65) and their Cartesian product equipped with a reduce function used to refine the information contained in one domain with the one tracked by the other one.

The main steps of the project will be to:

1. Implement in Sample a generic interface for non-relational numerical domains following the approach introduced by the lecture (slides 12-17) using the class project.AVPDomain to implement the environment
2. Instantiate it to and by implementing these two analyses
3. Develop the Cartesian product using the project. CartesianAVPDomain class
4. Apply your analyses to a set of case studies.

You can find a skeleton of an implementation of a (completely silly and useless!) domain that shows the main classes and methods you should use in your implementation. If you have questions and/or you need more explanations, send me an email! ([pietro.ferrara@gmail.com](mailto:pietro.ferrara@gmail.com))

At the end, you will have to submit

1. The code of your implementation
2. A report in which you
   1. formalize the arithmetic operators of following the style we have adopted during the lectures (e.g., slide 33) – no more than 2 pages
   2. discuss the results obtained when applying the analysis to the case studies

The main points of evaluation of the project will be

1. The soundness of your analyses
2. The consistence between the formal definitions and the implementation
3. The quality of the code you have developed (therefore, please comment it carefully!)
4. The content of the report

## Material

In the AVP webpage you can find:

1. SampleAVP.jar contains all the libraries you need to compile and run your analysis
2. CaseStudies.scala contains a set of case studies to which you should apply your analysis
3. skeleton.zip contains the Java source code of the skeleton