**Phase 1 – GanttProject**

**Code Smell – Switch Statement**

Uma imagem com texto

Descrição gerada automaticamente

biz.ganttproject.impex.csv.GanttCSVExport

This code snippet is a Switch Statement code smell because it is a very long switch statement that can be handled better using polymorphism.

**Refactoring:** creating an abstract class WriteColumn with subclasses (WriteNameColumn, WriteEmailColumn, etc..) that implement their own *print* method, and creating a **factory** that generates those instances depending on the type provided. This way, the client only needs to call the *print* method once. Here’s the stub of a refactoring:

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente**Code Smell – Long Parameter List**

net.sourceforge.ganttproject.task.algorithm.AlgorithmCollection

Uma imagem com texto

Descrição gerada automaticamenteThis code snippet is a Long Parameter List code smell because it has 8 parameters.

net.sourceforge.ganttproject.task. TaskManagerImpl

As seen in this code snippet from TaskManagerImpl, the arguments of the AlgorithmCollection constructor are just results of method calls to other classes, so we can pass this object as the sole parameter and let AlgorithmCollection get the values that it needs by itself.

**Refactoring:** the AlgorithmCollection constructor should only receive the *taskManager*, *criticalPathAlgorithm* and *scheduler* parameters. It has all it needs to infer the other algorithm objects in its constructor method.