Object Oriented Programming – 2017/2018 – 2nd Semester
Self-evaluation form

Group: **30**   Oral discussion date: **30/05/2018**   Penalization (days): **0**

Number: **78366**   Name: **Catarina França Martins Nabais**   Expected mark: **17**

Number: **78119**   Name: **Filipe Baptista**   Expected mark: **17**

Number: **78270**   Name: **Kyrylo Yefimenko**   Expected mark: **19**

Number: **78419**   Name: **Madalena França Martins Nabais**   Expected mark: **18**

Please fill the following form relative to the **implementation** of the project:

General aspects:
How do you classify the UML tool used (identify it)? **Visual Paradigm** ☐ Good ■ Fair ☐ Bad
Does your application use any external library, besides that provided within JDK?   ■ No ☐ Yes (which ones?):
How many packages does your application have? ☐ 1   ☐ 2   ■ ≥ 3: **6**
How many interfaces does your application have? ■ 1   ☐ 2   ☐ ≥ 3:
Is your application extensible to further developments? ☐ Yes   ☐ No   ■ Partialy
Does your application have at least one polymorphic invocation?   ☐ No   ■ Yes (methods?): **execute() , addToPec()**
How many times the `instanceof` operator is used in your application (really count them)? **2**   In which methods? **equals()**
Which XML parser is used to parse the input file? **SAX**
Have external libraries been required? ■ No ☐ Yes (which ones?):
Do you provide a DTD? ■ Yes   ☐ No   When parsing, is XML validated against it? ■ Yes   ☐ No
Concerning visibility of the fields, check visibilities that are used in the code: ■ Public   ■ Private   ☐ Package   ■ Protected
Concerning visibility of the methods, check visibilities that are used in the code: ■ Public   ■ Private   ■ Package   ☐ Protected
Concerning visibility of the classes, check visibilities that are used in the code: ■ Public   ☐ Package
Does your application contain any static field? ■ Yes (how many?): **11**   ☐ No
Does your application contain any static method? ■ Yes (how many?): **1**   ☐ No
Does your application contain any user defined exceptions? ☐ Yes (how many?):   ■ No

Simulation problem:
Data structure of the events (PEC): **ArrayList**   From `java.util`? ☐ No ■ Yes
Is it ordered? ☐ No   ■ Yes, with a: ☐ Comparable   ☐ Comparator   ■ Other
Are all events implemented as described in the project description and the FAQ?
Death: ■ Yes   ☐ With faults   ☐ Not implemented
Reproduction: ■ Yes   ☐ With faults   ☐ Not implemented
Move: ■ Yes   ☐ With faults   ☐ Not implemented
Are observations implemented as events? ■ Yes   ☐ No   All 20 at once in the PEC? ■ Yes   ☐ No
Data structure of the individuals: **Array List**   From `java.util`? ☐ No ■ Yes
Is it ordered? ■ No   ☐ Yes, with a: ☐ Comparable   ☐ Comparator   ☐ Other
Data structure for the grid: **HashMap**   From `java.util`? ☐ No ■ Yes
Are epidemics implemented as described in the project description? ■ Yes   ☐ With faults   ☐ No
Are the best 5 individuals stored in memory? ☐ Yes ■ No, they are calculated only when needed ☐ Other
To decide which individuals survive epidemics, is a random number generated per each? ■ Yes   ☐ No
Are nonsurvivors cleaned from memory? ■ Yes ☐ No (why):
Is the best path always found when you run the xml five provided in the Project webpage? ☐ Yes   ☐ No *

*(Encontramos sempre o melhor caminho do individuo na população, embora esse caminho possa não corresponder com o melhor caminho encontrado em toda a grelha. Isto é, há testes em que o melhor caminho dá custo 13 e esse é de facto o melhor caminho entre os indivíduos da população, embora o melhor caminho na grelha tenha custo 12).

Global evaluation:

What was the degree of participation of each element in the group? (% should sum 100%)?
Num__78366__ : __25__ % Num__78119__ : __25__ % Num__78270__ : __25__ % Num__78419__ : __25__ %

In the extent of your perception of the developed work, fill the following tables:

Project documentation	Yes	No
Is the project correctly documented through comments in the source code?	■	☐
Was the javadoc tool used to build the documentation of the developed packages?	■	☐
Is it complete, with:		
- overview of packages?	■	☐
- summary of classes, interfaces and exceptions?	■	☐
- brief description of classes, interfaces and exceptions?	■	☐
- summary of fields, constructors and methods?	■	☐
- detail of fields, constructors and methods?	■	☐

Project compilation	Yes	No
Does the project compile without errors?	■	☐
Does the project compile without warnings?	■	☐
If the answer is no, are all these warnings unchecked warnings?	☐	☐

Running	Yes	No	With faults
Is the jar file runnable from the shell?	■	☐	☐
Does the project read correctly the parameters?	■	☐	☐
Does the project run with the input given in the project webpage?	■	☐	☐
Does the project generate any supplementary information (status, debug, etc)?	☐	■	☐

Development environment used? ■ Linux   ☐ Windows   ☐ Unix   ☐ Mac/OS

Java version used: __8__

Was the final program tested in the laboratory workstations? ☐ Yes   ■ No

The following table is to be filled by the **professor**:

Report	Yes/Good	No/Bad	Incomplete/Fair
Cover identifies the course, authors and group number	☐	☐	☐
Goals of the work are very succinct but clearly stated	☐	☐	☐
Intelligibility of the document	☐	☐	☐
Structure of the document	☐	☐	☐
Clear/concise justification of main data structures used	☐	☐	☐
OO solution (extensibility, polymorphism, etc.)	☐	☐	☐
Critical evaluation of the application performance	☐	☐	☐
Description of functionalities beyond requested ones	☐	☐	☐
Conclusions	☐	☐	☐