

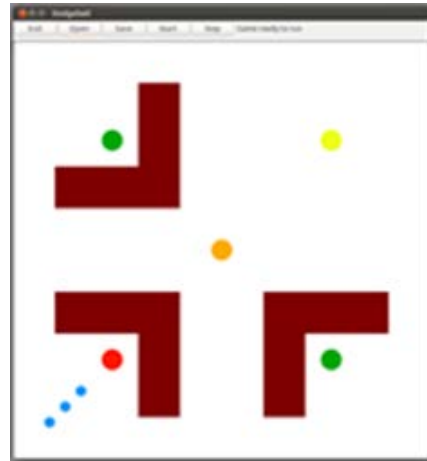
3. Execution tests on the spring VM

Expected drawings

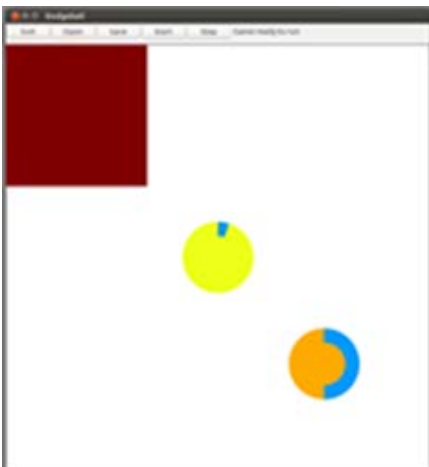
D01



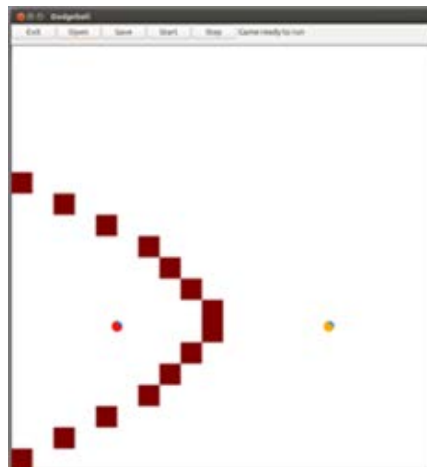
D02



D03



D04



3.4 Column AF: [Comp] => obtaining an executable with make gives 0.5pts
If it does not work at the first trial, remove the 0.5 point.

3.5 Column AG: [CL] Command Line (1.00pt).

Provide D01.txt on the command line as follows:

```
./projet D01.txt
```

Then quit the program with the Exit button. Run it again with:

```
./projet
```

Then use the Open Button to open D01.txt

=> 0.5 pt for each case. Approximate color matching is OK / message is graded at 3.8

Penalty about arcs: -0.25pt max for AG, AJ and AK (-0.75pt in total if all arcs are missing or wrong)

3.6 Column AH: [OSO] Open-Save-Open (1.75pt)

Run the program and use Open button to open D02.txt (0.5pt)

Use the Save button to Save the file as **ddd.txt** (0.75pt)

*Check the source code that the saved file is produced from the data structures and NOT by copying the opened file line by line : report such case as **fraud** and put **Opt** in this column and column AK.*

Use the Open button to read the saved file **ddd.txt** (0.5pt)

If an error message appears in the terminal then put max 0.5 pt in column AH

3.7 Column AI: [SSS] Start,Stop,Step (0.50pt)

Run the program and use Open button to open D02.txt. Check that the use of the buttons Start, Stop, Step produce a short message (stub).

3.8 Column AJ: [ED] Error Detection and message (1.25pt)

Run the program and use Open button to open D03.txt

0,25 for correct drawing and **0.25** for text “**Game ready to run**” next to buttons.

Penalty about arcs: -0.25pt max for **AG, AJ** and **AK** (-0.75pt in total if all arcs are missing or wrong)

Do NOT quit the program ; use the Open Button again for trying to open E01.txt.

The error message should appear in the terminal (pop-up window is ok too) (**0.25pt**)

In addition, the text “**No game to run**” must appear next to the buttons **if the previous context has been erased ; otherwise keep the current message (0.25pt)**.

Drawing: **0.25** for white window, or the previous file is still visible is also OK.

If the program does not handle the error detection, put max 0.5pt in column AJ

3.9 Column AK: [MF] Multiple Files (2.25pt)

Do NOT quit the program while performing this sequence of operation

Run the program and use **Open** button to open **D04.txt** (0.5 pt).

Save as eee.txt (0.25 pt)

Open D01.txt / D01 image should replace D04 image (0.25 pt)

Open eee.txt / D04 image should be visible again (0.25)

Open D03.txt / D03 image should be visible (0.25pt)

Save as fff.txt (0.25 pt)

Open D02.txt / D02 image should replace D03 image (0.25 pt)

Open fff.txt / D03 image should be visible again (0.25)

Penalty about arcs: -0.25pt max for **AG, AJ** and **AK** (-0.75pt in total if all arcs are missing or wrong)

4. Rendu2 ARCHITECTURE EVALUATION: check these three points

The spreadsheet column Y shows the default maximum of **4,5** points for ARCHITECTURE.

[A1] Architecture features to check for module projet:

Nb points (max=2pts)	Module role / separation of functionalities
[A1.1] 0,25	handle the command line arguments (argc and argv)
[A1.2] 0,25	connection with the Model sub-system with simulation.h only .
[A1.3] 0,50	There must be NO data structure of Player, Ball, Map, Obstacle in this module
[A1.4] 0,25	connection with gui through gui.h ; a Gui data structure is allowed
[A1.5] 0,75	There must be NO GTKmm function call in projet

- Although not requested, a Simulation data structure is allowed

[A2] Architecture features to check for the Model sub-system:

Nb points	Module role / separation of functionalities
[A2.1] 0,50	simulation.h must NOT be included in the lower-level modules (same as rendu1)
[A2.2] 0,50	There must be NO dependency to GTKmm in any Model module

[A3] Architecture features to check for module tool:

Nb points	Module role / separation of functionalities (same as rendu1)
[A3.1] 0,50	The module tool has to be independent from higher level modules as described on p6 “contrainte sémantique importante” of the project description

[A4] Architecture features to check for module gui:

Nb points	Module role / separation of functionalities
[A4.1] 0,25	connection with the Model sub-system with simulation.h only but simulation.h can include other interfaces for its own class needs. OK to include tools.h
[A4.2] 0,00	Although not requested, a Simulation data structure is allowed
[A4.3] 0,75	There must be NO function or method call from modules Player,Ball,Map,Obstacle

=> **Remove the number of point indicated for each feature that is not achieved.**

In the spreadsheets column Z architecture violation_comment, note down the corresponding **code(s)** : e.g. [A1.1], [A2.1], [A3.1], [A4.3] etc...

5. CLASS ENCAPSULATION / MODULARIZATION: (same as rendu1 except C2 penalty)

The spreadsheet column AA shows the **default maximum of 3 points** for **CLASS**.

[C1] Encapsulation violation : using any global variable or making any attribute public is strictly forbidden in any modules, including static attributes (no problem for methods and static methods).

It is allowed to have static variables in the implementation of a module (indicate a warning if there are too many of them).

No problem for struct model made visible in tools.h by the tools module.

=> **Remove 1 point per public attribute or global variable** (max 3pt).

[C2] Externalization of methods' definition : whenever a module interface shows a class interface, it should contain only method prototypes. The method definition must be externalized in the module implementation.

The only *accepted exception* of method definition in the class interface are the **getters** methods that fits onto the same line as the function prototype.

=> **Remove 0.5 point per interface that is not correctly externalized** (max 2 pt).

The total of removed points from C1 and C2 is maximum 3 pts.

In the spreadsheet column AB `class_violation_comment`, note down the corresponding **code [C1],[C2]** together with the **interface name** and the **public attribute name**. Indicate that it must be corrected in future assignments.

6. CODING STYLE (less criteria than rendu1 to spare time for execution tests)

The spreadsheet column AC shows the default maximum of 3 points for STYLE

[L1] Indentation rules have been ignored **more than 4 times** ; read carefully [the conventions](#) before considering this penalty because we accept some variants. Please note that we don't indent the public/private keywords in class declaration. Indicate only a **warning** if the whole code is consistent in the use of multiple brace styles (e.g. two styles are used but always in the same way, for the same control instructions)

[L2] There are **more than 4 wrapping line** in the code (more than 87 char); Indicate only a warning if 4 wrapping lines or less.

[P2] Apart from two functions of max 80 lines, all function size must not exceed 40 lines (+tolerance of 4 lines) with geany (with the default font size).

=> **remove 1 point per coding style criteria that is violated**

In the spreadsheet column R violation_list, note down the **code** representing the violated criteria followed by the file name and the **line numbers** it occurs. For instance **[L2] gui.cc 57,65,80-84** means that this set of lines are violating the wrapping criteria in gui.cc. If the same type of violation occurs more than 5 times, you mention briefly how much larger the problem is in the violation comment column S

Keep the violation_list alphabetically sorted and separate each entry by a comma.