

Numéro Sciper	PARTNER SCIPER	Architecture (max = 4.50pts)	Architecture violation comment	encapsulation (max = 3.00pts)	class violation comment
270354	302076	3,75	[A1.5] There should be no presence of GTKmm in project module	3	Good
271829	296759	3,5	[A1.2] Connection with Model sub-system through simulation. h only : why include all the .h in projet.cc ? [A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	RAS. L'encapsulation est bien gérée.
273712	301881	4,5		3	
275336	289067	4,5	-	3	
276861	301291	4,5	Les principes de l'architecture sont parfaitement respectés	3	Warning : counter dans player est une variable static évitable. Tous vos player sont dans un vecteur dans simulation, vecteur qui dispose d'une fonction size(). A corriger
279202	288530	3,75	[A1.4] projet.cc : calls to gtkmm. Simulation.cc : no need to prototype the functios already prototyped in simulation.h	0	[C1] player.h : l35. Map.h : l33. Ball.h : l32. The static arrays outside any functions must only be in the implementation (see forum)
282211	288820	4,5	Call your main module main.cpp and eliminate main.h !!!! You should have used the same architecture as in the project description... having so many more classes makes it a nightmare to correct	3	perfect
282287	283419	3,75	[A4.3] There must be NO function or method call from modules player, ball... (here use of getters from your public tab joueurs etc..) Hormis cela, l'architecture est le point fort de ce rendu. Les relations entre modules sont bien respectées et le Model-View-Controller aussi. Attention cependant à bien passer par simulation pour accéder aux données à afficher (ici vous appelez directement des méthodes qui viennent de classes plus bas dans l'architecture).	0	[C1] ATTENTION, il ne faut jamais mettre d'attributs publics ! Vos 3 tableaux d'entités joueur, ball, obstacle sont donc à mettre en private (et trouver donc une autre solution pour accéder à ces structures dans gui). [C2] Il faut externaliser toutes les méthodes de l'interface. Donc toutes les méthodes set (dans "Ball" par exemple) ne peuvent pas être définies dans la déclaration de la classe (sauf si la définition tient sur une ligne). Attention donc à ne laisser que les prototypes pour le prochain rendu.

282360	289256	4,5	Warning : Prenez l'autorisation de Mr. Boulic pour la multiplication de modules.	3	Nice encapsulation.
282409	284529	4,5		1	[C2] initialize, setReady, setNotReady, Data
282416	287609	4	[A3.1] Tool has to be independant from higher lever OR vous incluez "define.h" et utilisez DIM_MAX Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	Bonne encapsulation ! Continuez
282435	286955	4,5	Bravo, l'architecture du projet respecte bien les attentes pour ce projet. Dans le module tools, renommer la fonction cercle_obstacle pour rester le plus général possible (cercle_carre ?)	3	Très bien.
282515	288166	4,25	[A1.2] : vous ne devez pas inclure player, map et ball dans gui	2,5	[C2] map.h : vous ne devez définir une fonction dans le .h que si c'est un getter, et sur la même ligne que le prototype le cas échéant
282565	284284	3,5	[A1.3][A2.2] : vous ne devez pas inclure ball, map, player dans projet ni dans gui	3	L'encapsulation est bien respectée
282657	287800	4,5		3	
282708	288602	3,75	[A1.5]	3	
282756	286734	3,75	[A1.5]	3	
282836	302583	4,5		3	
282844	288453	3,75	[A1.5]	3	
282988	284213	3,75	[A4.2] graphique.cc use of simulation modules calls. progactu.cc : l3 useless include	3	Good !
283111	284430	3,75	[A4.3]	3	ok
283183	296952	3,75	[A1.5]	3	perfect
283196	284316	4,5	Bonne architecture. Je ne comprends pas pourquoi vous ne modifiez pas directement les attributs de votre classe SimData dans simulation au lieu de passer par une variable statique, mais votre idée fonctionne	3	Les principes d'encapsulation sont bien respectés
283391	289122	4,5		3	
283395	288243	4	[A2.1]	0	[C2]: info_file.h: m_total,m_nbCell,m_nbt,m_count,m_ligne,m_colonne, m_angle, m_pos_x, m_pos_y,

283419	282287	3,75	[A4.3] There must be NO function or method call from modules player, ball... (here use of getters from your public tab joueurs etc..) Hormis cela, l'architecture est le point fort de ce rendu. Les relations entre modules sont bien respectées et le Model-View-Controller aussi. Attention cependant à bien passer par simulation pour accéder aux données à afficher (ici vous appelez directement des méthodes qui viennent de classes plus bas dans l'architecture).	0	[C1] ATTENTION, il ne faut jamais mettre d'attributs publics ! Vos 3 tableaux d'entités joueur, ball, obstacle sont donc à mettre en private (et trouver donc une autre solution pour accéder à ces structures dans gui). [C2] Il faut externaliser toutes les méthodes de l'interface. Donc toutes les méthodes set (dans "Ball" par exemple) ne peuvent pas être définies dans la déclaration de la classe (sauf si la définition tient sur une ligne). Attention donc à ne laisser que les prototypes pour le prochain rendu.
283442	287811	4,5		3	
283470	286317	3,75	[A1.5]	3	
283616	301494	4,5		3	
283652	303039	4,5		3	
283670	286570	3,75	[A1.5]	1	[C2] Joueur, ~Joueur, Ball, setXpix, ...
283792	287054	4,5	Parfait	3	Niquel
283796	286386	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie.	3	Très bien.
284024	289159	4,5	Good !	3	Good
284194	287056	3,75	[A1.5] There should be no presence of GTKmm in project module	3	Good
284196	299245	3,75	[A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	L'encapsulation de vos classes est très bonne à part ça et la notion d'héritage est particulièrement bien utilisée.
284213	282988	3,75	[A4.2] graphique.cc use of simulation modules calls. progactu.cc : l3 useless include	3	Good !
284284	282565	3,5	[A1.3][A2.2] : vous ne devez pas inclure ball, map, player dans projet ni dans gui	3	L'encapsulation est bien respectée
284316	283196	4,5	Bonne architecture. Je ne comprends pas pourquoi vous ne modifiez pas directement les attributs de votre classe SimData dans simulation au lieu de passer par une variable statique, mais votre idée fonctionne	3	Les principes d'encapsulation sont bien respectés

284402	289234	3,75	[A1.5]	3	
284430	283111	3,75	[A4.3]	3	ok
284529	282409	4,5		1	[C2] initialize, setReady, setNotReady, Data
284564	289285	3,25	[A1.5], [A2.1] player.cc ball.cc	3	static public functions is non-sense
284867	296833	3	[A1.4] projet.cc:use of gtkmm. [A4.2] gui.cc : call to methods of simulation modules	3	ball.h : angle could be private. player.h : same for nbT and count
284888	286950	4,5	Very good !	3	Good
284896	301437	4,5	Si le recours au module lecture peut être justifié, l'ajout d'un module drawing est plus discutable car il ne reste alors quasiment rien dans le module gui. Ceux-ci pourraient être regroupés, d'autant plus qu'ils ont tous les deux la même fonction, à savoir gérer l'affichage de la fenêtre. A part ça votre architecture est correcte.	3	Encapsulation très bien appliquée, bravo.
285416	289276	4,5		3	
286317	283470	3,75	[A1.5]	3	
286386	283796	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie.	3	Très bien.
286570	283670	3,75	[A1.5]	1	[C2] Joueur, ~Joueur, Ball, setXpix, ...
286641	286721	4,5		3	
286721	286641	4,5		3	
286734	282756	3,75	[A1.5]	3	
286830	289281	4,5	Excellent respect de l'architecture	3	Bon respect des principes d'encapsulation
286950	284888	4,5	Very good !	3	Good
286955	282435	4,5	Bravo, l'architecture du projet respecte bien les attentes pour ce projet. Dans le module tools, renommer la fonction cercle_obstacle pour rester le plus général possible (cercle_carre ?)	3	Très bien.
286962	301464	4,25	[A1.1] projet.cc : argc and argv handling is also done by the gui module	3	Good !
287015	296048	4,5		2	[C2]: Player: Getcircle
287038	297846	4,5	You should have used the same architecture as in the project description... having so many more classes makes it a nightmare to correct	3	perfect
287054	283792	4,5	Parfait	3	Niquel

287056	284194	3,75	[A1.5] There should be no presence of GTKmm in project module	3	Good
287173	301560	4,5		0	[C1]: simulation: nbC,nbP,Player *plrs,nbO,Obstacle *obs,nbB,Ball *bls,double pRad, pSpd, bRad, bSpd, MJ, ML,
287327	287388	4,5	perfect	3	perfect
287388	287327	4,5	perfect	3	perfect
287433	300540	4,5	You should have used the same architecture as in the project description... having so many more classes makes it a nightmare to correct	3	perfect
287459	302017	4,5	Excellente architecture, attention simplement à ne pas inclure directement gtkmm.h. En effet cela inclut tout GTK et rend la compilation trop lente. Ajoutez uniquement les classes nécessaires	3	L'encapsulation est parfaite
287480					
287609	282416	4	[A3.1] Tool has to be independant from higher lever OR vous incluez "define.h" et utilisez DIM_MAX Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	Bonne encapsulation ! Continuez
287725	288407	3,75	[A1.5]	3	perfect
287759	295954	4,5		3	
287800	282657	4,5		3	
287811	283442	4,5		3	
288166	282515	4,25	[A1.2] : vous ne devez pas inclure player, map et ball dans gui	2,5	[C2] map.h : vous ne devez définir une fonction dans le .h que si c'est un getter, et sur la même ligne que le prototype le cas échéant
288243	283395	4	[A2.1]	0	[C2]: info_file.h: m_total,m_nbCell,m_nbt,m_count,m_ligne,m_colonne, m_angle, m_pos_x, m_pos_y,
288252	303011	3,75	[A1.5]projet.cc: Gtkmm ne peut pas être inclus dans projet, l'appel des fonctions de cette librairie externe est réservée au module gui. Warning : Le module tools mériterait d'être étoffé.	3	Bonne encapsulation. Certaines variables statiques auraient pu être évitées : passées en attribut d'une classe par exemple.

288287	299554	4,5	Parfait !	3	Parfait !
288312	296165	4,25	A[1.2] : vous ne devez pas inclure ball, map et player dans projet.	3	Bon travail avec l'encapsulation
288407	287725	3,75	[A1.5]	3	perfect
288453	282844	3,75	[A1.5]	3	
288530	279202	3,75	[A1.4] projet.cc : calls to gtkmm. Simulation.cc : no need to prototype the functios already prototyped in simulation.h	0	[C1] player.h : I35. Map.h : I33. Ball.h : I32. The static arrays outside any functions must only be in the implementation (see forum)
288580	288825	4,25	[A1.1]	3	
288602	282708	3,75	[A1.5]	3	
288820	282211	4,5	Call your main module main.cpp and eliminate main.h !!!! You should have used the same architecture as in the project description... having so many more classes makes it a nightmare to correct	3	perfect
288825	288580	4,25	[A1.1]	3	
288838	289028	3,5	[A1.2], [A1.5]	3	
289028	288838	3,5	[A1.2], [A1.5]	3	
289067	275336	4,5	-	3	
289121	296142	4,5	Attention à ne pas inclure tout gtkmm.h car cela comprend tout GTK et rend la compilation très lente. N'incluez que ce qui est nécessaire. Warning : Gui n'est pas censé utiliser des data-structure du modèle, Player par exemple. Essayez de passer par une sur-classe de Player définie dans tools, comme Circle.	3	Très bonne encapsulation
289122	283391	4,5		3	
289159	284024	4,5	Good !	3	Good
289188	295855	4,5		1	Vous utilisez des tableaux globaux pour les joueurs et les balls
289234	284402	3,75	[A1.5]	3	
289256	282360	4,5	Warning : Prenez l'autorisation de Mr. Boulic pour la multiplication de modules.	3	Nice encapsulation.
289276	285416	4,5		3	
289281	286830	4,5	Excellent respect de l'architecture	3	Bon respect des principes d'encapsulation
289285	284564	3,25	[A1.5], [A2.1] player.cc ball.cc	3	static public functions is non-sense

289304	295808	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie.	3	Principe d'encapsulation parfaitement mis en oeuvre.
289324	289596	3,75	[A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	Bonne encapsulation ! Continuez
289596	289324	3,75	[A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	Bonne encapsulation ! Continuez
290067	300669	3,75	[A1.4][A2.2] : simulation.h ne doit pas voir gui.h. Votre module projet appelle une fonction de simulation qui elle même appelle une fonction de la GUI. C'est une erreur : le module projet doit lui-même déclencher l'affichage de la Gui	3	Warning : définir non pas une mais deux classes dans le gui.cc est un mauvais choix pour la lisibilité. Mais ce choix étant cohérent avec le choix d'architecture pour lequel je vous ai déjà enlevé des points, je ne vous mets qu'un warning. A corriger
290470	296032	4,5	Architecture du projet ok. L'inclusion de gtkmm pourrait être évitée car elle est inutile dans ce module.	0	[C1]ball.h:25, map.h:46, player.h:28, simulation.h:25 : le recours aux variables globales est interdit, car il est contraire au principe d'encapsulation. Seules des variables globales statiques sont autorisées dans l'implémentation d'un module.
295758	296442	4,5		3	
295784	296046	3,75	[A1.5] There should be no presence of GTKmm in project module	3	Good
295808	289304	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie.	3	Principe d'encapsulation parfaitement mis en oeuvre.
295821	299895	4,5		3	
295855	289188	4,5		1	[C1]Vous utilisez des tableaux globaux pour les joueurs et les balls
295890	302360	3,75	[A1.5]	3	
295927	300882	3,75	[A1.4] projet.cc : calls to gtkmm	3	Good !
295954	287759	4,5		3	

296023	296815	4,5	Architecture parfaite	3	Encapsulation parfaite
296026	301418	3,75	[A1.5]	0	[C1] simulation.cc [C2] Player, Ball, Map
296032	290470	4,5	Architecture du projet ok. L'inclusion de gtkmm pourrait être évitée car elle est inutile dans ce module.	0	[C1]ball.h:25, map.h:46, player.h:28, simulation.h:25 : le recours aux variables globales est interdit, car il est contraire au principe d'encapsulation. Seules des variables globales statiques sont autorisées dans l'implémentation d'un module.
296046	295784	3,75	[A1.5] There should be no presence of GTKmm in project module	3	Good
296048	287015	4,5		2	[C2]: Player: Getcircle
296050	297097	4,5	Architecture parfaite	3	Les interfaces et l'encapsulation sont nickels
296057	301225	4,5	Warning : Include gtkmm.h rend la compilation très lente, car vous incluez toute la librairie GTK qui est énorme. N'utilisez que les classes nécessaires	3	Les principes d'encapsulation sont parfaitement respectés
296093	300411	3,75	[A4.3] : vous ne devez pas utiliser Ball, Player et Obstacle depuis Gui. Warning : include gtkmm.h inclut toute la librairie, ce qui est catastrophique pour le temps de compilation. N'incluez que les classes nécessaires	3	Les principes d'encapsulation sont bien respectés
296142	289121	4,5	Attention à ne pas inclure tout gtkmm.h car cela comprend tout GTK et rend la compilation très lente. N'incluez que ce qui est nécessaire. Warning : Gui n'est pas censé utiliser des data-structure du modèle, Player par exemple. Essayez de passer par une sur-classe de Player définie dans tools, comme Circle.	3	Très bonne encapsulation
296165	288312	4,25	A[1.2] : vous ne devez pas inclure ball, map et player dans projet.	3	Bon travail avec l'encapsulation
296169	302659	4,5	Warning : vous devez traiter tous les cas possibles lorsque vous analysez le nombre d'argument passé sur la ligne de commande. Par ailleurs, vous avez beaucoup d'include inutiles, notamment gtkmm.h qui inclut tout GTK et rend la compilation inutilement lente. Mais aussi fstream, string, etc ne sont pas toujours nécessaires.	3	Encapsulation correcte, comme au premier rendu vos fonctions seraient mieux définies dans des classes qu'en dehors.
296235	297075	4,5	Attention to inclusion of Gtkmm in simulation.cc (not penalised only because it was a distraction). Otherwise perfect	3	static public functions is non-sense.

296372	303028	4	[A2.2] : La gui ne doit pas être incluse dans la simulation. Warning : votre fonction lancerSimu est inutile, et le switch qu'elle contient ne traite pas tous les cas. Il faut y ajouter un default. Warning : include gtkmm.h inclut toute la librairie, ce qui est catastrophique pour le temps de compilation. N'incluez que les classes nécessaires	2	[C1] : gui.h 11, en oubliant le mot clé statique cette variable est globale. Le mot clé const n'a aucune influence : certes vous ne pourrez pas redéfinir ce vector, mais son contenu oui en appelant une méthode push_back() par exemple. Warning : Les constantes doivent être mises en place avec des #define, pas avec des variables statiques constantes. Cela aurait évité votre erreur.
296399	296487	4,5	-	1	[C2] setnbCell, setnbJ, setnbB, setnbO
296442	295758	4,5		3	
296487	296399	4,5	-	1	[C2] setnbCell, setnbJ, setnbB, setnbO
296643	299689	4,5	Warning : Include gtkmm.h rend la compilation très lente, car vous incluez toute la librairie GTK qui est énorme. N'utilisez que les classes nécessaires	3	Bon respect des principes d'encapsulation. Warning : pourquoi avoir des vecteurs statiques dans les modules ET des attributs privés de Simulation qui sont des vecteurs de ball, obstacle et player ?
296644	298386	4,5	Warning : Include gtkmm.h inclut toute la librairie GTK dans votre projet, ce qui rend la compilation très lourde. Evitez, n'incluez que les classes nécessaires. Pensez également à ajouter vos define dans un fichier à part s'ils y en a beaucoup	3	Très bon respect de l'encapsulation
296759	271829	3,5	[A1.2] Connection with Model sub-system through simulation. h only : why include all the .h in projet.cc ? [A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	RAS. L'encapsulation est bien gérée.
296815	296023	4,5	Architecture parfaite	3	Encapsulation parfaite
296816	301033	4,5	Excellente architecture. Un conseil, mettez toujours tous vos prototypes en haut du fichier pour qu'il soit plus clair et plus facilement maintenable (dans gui.cc par exemple).	2,5	Excellente encapsulation. Attention à bien externaliser toutes les méthodes. Seules les méthodes tenant sur une seule ligne sont acceptées. [C2] simulation.cc (Game)

296827	299894	3,75	[A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	RAS. L'encapsulation est bien gérée.
296833	284867	3	[A1.4] projet.cc:use of gtkmm. [A4.2] gui.cc : call to methods of simulation modules	3	ball.h : angle could be private. player.h : same for nbT and count
296867	300131	3,75	[A1.5]	3	perfect
296935	301358	4,5	Warning : I understand what you tried to do, but this for-loop seems to be a bad idea. It is not intuitive for example if I run ./main Erreur Step, but it will "work" ...	2,5	[C2] your GUI module is badly separated, you should not put all in the .cc
296952	283183	3,75	[A1.5]	3	perfect
297075	296235	4,5	Attention to inclusion of Gtkmm in simulation.cc (not penalised only because it was a distraction). Otherwise perfect	3	static public functions is non-sense.
297097	296050	4,5	Architecture parfaite	3	Les interfaces et l'encapsulation sont nickels
297151	301452	2,75	[A1.5], [A2.1], [A2.2]	3	
297176	301998	4,5	Très bien.	3	Principe d'encapsulation bien mis en oeuvre.
297846	287038	4,5	You should have used the same architecture as in the project description... having so many more classes makes it a nightmare to correct	3	perfect
298011	302747	3,75	[A1.5]	1	[C2] Player::flush(), Ball.flush(), Circle::setCentre, Circle::setRayon()
298386	296644	4,5	Warning : Include gtkmm.h inclut toute la librairie GTK dans votre projet, ce qui rend la compilation très lourde. Evitez, n'incluez que les classes nécessaires. Pensez également à ajouter vos define dans un fichier à part s'ils y en a beaucoup	3	Très bon respect de l'encapsulation

299245	284196	3,75	[A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	L'encapsulation de vos classes est très bonne à part ça et la notion d'héritage est particulièrement bien utilisée.
299479	299496	4,5	-	1	[C2] Player, Ball, Cercle, Point, ...
299496	299479	4,5	-	1	[C2] Player, Ball, Cercle, Point, ...
299554	288287	4,5	Parfait !	3	Parfait !
299561	300845	3,75	[A1.4] projet.cc : calls to gtkmm	0	[C1] simulation.h : l21 you create a public version of player. Structs are only allowed in tools.
299689	296643	4,5	Warning : Include gtkmm.h rend la compilation très lente, car vous incluez toute la librairie GTK qui est énorme. N'utilisez que les classes nécessaires	3	Bon respect des principes d'encapsulation. Warning : pourquoi avoir des vecteur statiques dans les modules ET des attributs privés de Simulation qui sont des vecteurs de ball, obstacle et player ?
299724	301959	4,5	Good !	3	Good
299882	301272	3,75	[A1.5]	3	
299892	300594	4,25	[A1.2] : vous devez inclure simulation.h, votre code ne compile que parce que gui.h inclut simulation.h	2,5	[C2] player.h : vous ne devez pas définir la liste d'initialisation de votre constructeur dans le .h
299894	296827	3,75	[A1.5] There must be NO GTKmm function call in projet. Hormis cette étourderie, l'architecture est très bien respectée. Vous semblez avoir compris les enjeux de la modularisation.	3	RAS. L'encapsulation est bien gérée.
299895	295821	4,5		3	
300131	296867	3,75	[A1.5]	3	perfect
300253	302703	4,5	perfect	3	static public functions is non-sense
300411	296093	3,75	[A4.3] : vous ne devez pas utiliser Ball, Player et Obstacle depuis Gui. Warning : include gtkmm.h inclut toute la librairie, ce qui est catastrophique pour le temps de compilation. N'incluez que les classes nécessaires	3	Les principes d'encapsulation sont bien respectés

300443	301030	4,5		2	[C1]: gui, m_area
300456	301829	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie. Le module circle parait inutile : la classe circle aurait pu être ajoutée au module tools.	3	Principe d'encapsulation très bien appliqué.
300540	287433	4,5	You should have used the same architecture as in the project description... having so many more classes makes it a nightmare to correct	3	perfect
300558	302896	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie.	3	Encapsulation très bien respectée.
300559	302674	3,75	[A1.5] There should be no presence of GTKmm in project module.	3	[Comment] I don't understand how semantically Cercle and Carre are a Point. It might be necessary to comment this choice of conception.
300594	299892	4,25	[A1.2] : vous devez inclure simulation.h, votre code ne compile que parce que gui.h inclut simulation.h	2,5	[C2] player.h : vous ne devez pas définir la liste d'initialisation de votre constructeur dans le .h
300615	301581	3,25	[A1.5], [A3.1]tool.h should not include gtkmm	3	
300620	301641	4,5		3	
300669	290067	3,75	[A1.4][A2.2] : simulation.h ne doit pas voir gui.h. Votre module projet appelle une fonction de simulation qui elle même appelle une fonction de la GUI. C'est une erreur : le module projet doit lui-même déclencher l'affichage de la Gui	3	Warning : définir non pas une mais deux classes dans le gui.cc est un mauvais choix pour la lisibilité. Mais ce choix étant cohérent avec le choix d'architecture pour lequel je vous ai déjà enlevé des points, je ne vous mets qu'un warning. A corriger
300803	301708	4	[A3.1] tools.cc : l20 : Game::getCell()	3	Good !
300845	299561	3,75	[A1.4] projet.cc : calls to gtkmm	0	[C1] simulation.h : l21 you create a public version of player. Structs are only allowed in tools.
300882	295927	3,75	[A1.4] projet.cc : calls to gtkmm	3	Good !
301001	302551	4,5	projet.cc : useless gtkmm includes. projet.h : can be removed	3	Good !
301030	300443	4,5		2	[C1]: gui, m_area

301033	296816	4,5	<p>Excellente architecture.</p> <p>Un conseil, mettez toujours tous vos prototypes en haut du fichier pour qu'il soit plus clair et plus facilement maintenable (dans gui.cc par exemple).</p>	2,5	<p>Excellente encapsulation.</p> <p>Attention à bien externaliser toutes les méthodes. Seules les méthodes tenant sur une seule ligne sont acceptées.</p> <p>[C2] simulation.cc (Game)</p>
301225	296057	4,5	<p>Warning : Include gtkmm.h rend la compilation très lente, car vous incluez toute la librairie GTK qui est énorme.</p> <p>N'utilisez que les classes nécessaires</p>	3	Les principes d'encapsulation sont parfaitement respectés
301265	301403	4,25	<p>Architecture très propre et bien modularisée.</p> <p>Attention à bien respecter la figure 9 et faire la connexion entre "projet" et le modèle via simulation.h (et non pas gui). Votre solution fonctionne mais est relativement faible car rend le projet très dépendant de Gtkmm/Gui.</p>	3	RAS. L'encapsulation est bien gérée.
301272	299882	3,75	[A1.5]	3	
301291	276861	4,5	Les principes de l'architecture sont parfaitement respectés	3	Warning : counter dans player est une variable static évitable. Tous vos player sont dans un vecteur dans simulation, vecteur qui dispose d'une fonction size(). A corriger
301358	296935	4,5	<p>Warning : I understand what you tried to do, but this for-loop seems to be a bad idea. It is not intuitive for example if I run ./main Erreur Step, but it will "work" ...</p>	2,5	[C2] your GUI module is badly separated, you should not put all in the .cc
301403	301265	4,25	<p>Architecture très propre et bien modularisée.</p> <p>Attention à bien respecter la figure 9 et faire la connexion entre "projet" et le modèle via simulation.h (et non pas gui). Votre solution fonctionne mais est relativement faible car rend le projet très dépendant de Gtkmm/Gui.</p>	3	RAS. L'encapsulation est bien gérée.
301411	302276	3,75	[A1.5]	1	[C2] MyArea, ~MyArea, set_rempli, ~Ball, ~Player, ~Obstacle
301418	296026	3,75	[A1.5]	0	[C1] simulation.cc [C2] Player, Ball, Map
301437	284896	4,5	<p>Si le recours au module lecture peut être justifié, l'ajout d'un module drawing est plus discutable car il ne reste alors quasiment rien dans le module gui. Ceux-ci pourraient être regroupés, d'autant plus qu'ils ont tous les deux la même fonction, à savoir gérer l'affichage de la fenêtre. A part ça votre architecture est correcte.</p>	3	Encapsulation très bien appliquée, bravo.

301452	297151	2,75	[A1.5], [A2.1], [A2.2]	3	
301464	286962	4,25	[A1.1] projet.cc : argc and argv handling is also done by the gui module	3	Good !
301494	283616	4,5		3	
301560	287173	4,5		0	[C1]: simulation: nbC,nbP,Player *plrs,nbO,Obstacle *obs,nbB,Ball *bls,double pRad, pSpd, bRad, bSpd, MJ, ML,
301581	300615	3,25	[A1.5], [A3.1]tool.h should not include gtkmm	3	
301595	302759	3,75	[A1.4] projet.cc : calls to gtkmm	3	Good !
301633	302909	3,75	[A1.5]	3	
301641	300620	4,5		3	
301708	300803	4	[A3.1] tools.cc : l20 : Game::getCell()	3	Good !
301750	302031	3,75	[A1.4] projet.cc : calls to gtkmm	3	Good !
301766	302289	4,5		3	
301829	300456	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie. Le module circle paraît inutile : la classe circle aurait pu être ajoutée au module tools.	3	Principe d'encapsulation très bien appliqué.
301836	303153	4,5		3	
301881	273712	4,5		3	
301949	302565	4,5		3	
301956	301994	4,5	Good !	3	Warning : strange duplication of Player, Map and Ball in Simulation.h ... You need to discuss this choice of conception with Ronan before the final submit.
301959	299724	4,5	Good !	3	Good
301994	301956	4,5	Good !	3	Warning : strange duplication of Player, Map and Ball in Simulation.h ... You need to discuss this choice of conception with Ronan before the final submit.
301998	297176	4,5	Très bien.	3	Principe d'encapsulation bien mis en oeuvre.
302017	287459	4,5	Excellente architecture, attention simplement à ne pas inclure directement gtkmm.h. En effet cela inclut tout GTK et rend la compilation trop lente. Ajoutez uniquement les classes nécessaires	3	L'encapsulation est parfaite
302031	301750	3,75	[A1.4] projet.cc : calls to gtkmm	3	Good !
302076	270354	3,75	[A1.5] There should be no presence of GTKmm in project module	3	Good

302276	301411	3,75	[A1.5]	1	[C2] MyArea, ~MyArea, set_rempli, ~Ball, ~Player, ~Obstacle
302289	301766	4,5		3	
302327	302584	4,5	L'architecture est bien respectée. Attention à éviter d'inclure tout GTK en utilisant gtkmm.h, car cela rend la compilation très lente. N'incluez que les classes de GTK dont vous avez besoin.	3	Warning : quasiment toutes les fonctions que vous laissez en dehors des classes devraient être à l'intérieur pour faire de l'orienté-objet correct. Ce choix que vous faites impose un grand nombre de variable statique : bref, vous restez dans un esprit très séquentiel et ajoutez des classes à droite à gauche pour "faire orienté-objet". C'est dommage car votre code en devient nécessairement moins lisible et plus compliqué à produire pour vous
302360	295890	3,75	[A1.5]	3	
302551	301001	4,5	projet.cc : useless gtkmm includes. projet.h : can be removed	3	Good !
302565	301949	4,5		3	
302583	282836	4,5		3	
302584	302327	4,5	L'architecture est bien respectée. Attention à éviter d'inclure tout GTK en utilisant gtkmm.h, car cela rend la compilation très lente. N'incluez que les classes de GTK dont vous avez besoin.	3	Warning : quasiment toutes les fonctions que vous laissez en dehors des classes devraient être à l'intérieur pour faire de l'orienté-objet correct. Ce choix que vous faites impose un grand nombre de variable statique : bref, vous restez dans un esprit très séquentiel et ajoutez des classes à droite à gauche pour "faire orienté-objet". C'est dommage car votre code en devient nécessairement moins lisible et plus compliqué à produire pour vous
302659	296169	4,5	Warning : vous devez traiter tous les cas possibles lorsque vous analysez le nombre d'argument passé sur la ligne de commande. Par ailleurs, vous avez beaucoup d'include inutiles, notamment gtkmm.h qui inclut tout GTK et rend la compilation inutilement lente. Mais aussi fstream, string, etc ne sont pas toujours nécessaires.	3	Encapsulation correcte, comme au premier rendu vos fonctions seraient mieux définies dans des classes qu'en dehors.
302674	300559	3,75	[A1.5] There should be no presence of GTKmm in project module.	3	[Comment] I don't understand how semantically Cercle and Carre are a Point. It might be necessary to comment this choice of conception.
302703	300253	4,5	perfect	3	static public functions is non-sense
302747	298011	3,75	[A1.5]	1	[C2] Player::flush(), Ball.flush(), Circle::setCentre, Circle::setRayon()
302759	301595	3,75	[A1.4] projet.cc : calls to gtkmm	3	Good !

302896	300558	3,75	[A1.5] projet.cc : seul le module gui est autorisé à utiliser la librairie externe gtkmm, le module projet ne peut pas appeler de fonction de cette librairie.	3	Encapsulation très bien respectée.
302909	301633	3,75	[A1.5]	3	
303011	288252	3,75	[A1.5]projet.cc: Gtkmm ne peut pas être inclus dans projet, l'appel des fonctions de cette librairie externe est réservée au module gui. Warning : Le module tools mériterait d'être étoffé.	3	Bonne encapsulation. Certaines variables statiques auraient pu être évitées : passées en attribut d'une classe par exemple.
303028	296372	4	[A2.2] : La gui ne doit pas être incluse dans la simulation. Warning : votre fonction lancerSimu est inutile, et le switch qu'elle contient ne traite pas tous les cas. Il faut y ajouter un default. Warning : include gtkmm.h inclut toute la librairie, ce qui est catastrophique pour le temps de compilation. N'incluez que les classes nécessaires	2	[C1] : gui.h 11, en oubliant le mot clé statique cette variable est globale. Le mot clé const n'a aucune influence : certes vous ne pourrez pas redéfinir ce vector, mais son contenu oui en appelant une méthode push_back() par exemple. Warning : Les constantes doivent être mises en place avec des #define, pas avec des variables statiques constantes. Cela aurait évité votre erreur.
303039	283652	4,5		3	
303153	301836	4,5		3	
305538	270354	3,75	[A1.5] There should be no presence of GTKmm in project module	3	Good