

SCIPER	Coding Style max (4pts)	violation list	violation comment	Global comment
283389	2,00	[L1][P5.2]ville_2.h l34 , l24 , l41 noeud.h l24,l22,l25	dans tes classes private/public ne doit pas etre indednte , si tu casse ta ligne a cause de trop de parametre , n'oublie pas l indendttation. Donnez des noms plus significatif que : const int trois(3) , size_t un(1) . Vous n etes pas consistant avec vos {} , choisissez un style.	Bon travail , on voit que vous avez passe du temps dessus , mais faites attention a "netoyer" ton rendu , les .o et executables sont pas necessaire , les fonctions pour debuguer ne doivent pas etre appelle au moment de rendre le code
287006	2,00	[L1] noeud.cc 141, 149, ville.cc 174-237 [P5] constants.h 31,32,33,37,38	Inconsistent indentation of if-else, several instructions on same line.	As a whole, good project. It's a shame for the architecture violation in projet.cc, encapsulation is always respected. Coding style is pretty good, with the exception of a few indentation and wrapping errors. You should really pay attention to what is considered a magic number and what not to avoid defining meaningless and confuring constants)
288423	4,00		Lorsque vos lignes dépassent les 87 caractères et que vous faites un retour à la ligne, veillez à toujours la rabattre de la même manière. Aérer le code entre vos fonctions pourrait le rendre légèrement plus agréable à la lecture mais dans l'ensemble votre style respecte très bien les conventions.	On peut observer dans votre code des restrictions mises en place et pensées pour la suite du projet (notamment les delete et les const). C'est une très bonne pratique, continuez comme ça! Vous montrez une bonne compréhension des concepts appris avec votre mise en place de l'héritage qui servira plus tard. Faites tout de même attention aux méthodes qui devront être virtuelles ou pas !
288781	3,00	[L1] noeud.cc 25,38,45,54,59,...: more than one instruction on one line / noeud.cc 97,99,172,174,178,...; tools.cc 10: long instructions in if, or instruction on multiple lines not correctly aligned.	The code has a non-unifrom style sometimes there is some space sometimes not and when you have to write an instruction on multiple lines indent correctly with respect to conventions posted on moodle.	Architecture is OK even if you have included error and constant module in projet.cc, but there is a lot to improve: don't try to save lines in the reading function by putting multiple instructions in the same line, you'd better delegate some tasks to sub-functions; getters are not const; you also made an instance of class Ville and then break the encapsulation, you should delclare it in the ville module; you need to externalize the definition of methods of the class; be careful having public attributes is forbidden; and to finish pay attention to indentation rules and conventions posted on moodle.

291348	3,00	[L1] noeud.h 27 ; ville.cc 88,102,116,... ; noeud.cc 68	Several instructions on the same line are not allowed.	As a whole, good project. Architecture is respected throughout. Encapsulation is violated in several cases due to definition of functions in .h, but is otherwise respected. Some indentation mistakes were unfortunately present.
295752	4,00		Good style, coherent throughout the code. You define the variable erreur_potentiel 2 times, once in tools.cc and another in constantes and this value is different each time. Watch out for a magic number at line 71 in noeud.cc.	I see you check for the .txt extension, it is a good initiative. Watch out since in Ubuntu extensions are not required so the check is not required. You should remove all includes that are not useful in your files. Great job otherwise, style is good and simple, architecture is correct and class encapsulation is respected. Consider adding inheritance next. Good job!
295798	2,00	[L1] noeud.cpp 88 ; city.cpp 192-227 ; [L2] all comments	several lines of cout must be aligned to << not to "cout"; several instructions on the same line are not allowed ; closing } is always, for all conventions, aligned with instruction opening { ; all comments are aligned left so that the last char is on col 88, but max col width is 87	As a whole, good project. Encapsulation is respected throughout. Architecture is respected with the only exceptions being error::success and argc in city. Coding style is good, with indentation mistakes mainly concentrated in one function and unfortunately many lines wrap because you set your maximum column width to a wrong value.
295810	4,00	[L1]noeud.cc:128	Bon style en général le code est propre, attention tout de même vous avez deux fonctions qui sont presque à 80 lignes pensez à refactor	Bon rendu et joli style bravo

296202	4,00	OK	Warning:[L1] 22,30 Noeud.h and 14,22 Ville.h: public and private keywords are indeted [L2] 32 Ville.cc: more than 87 char.	Good project overall, the architecture and the coding style are good even if there is few errors for the coding style. Just a few points: you forgot the return 0 in the main; you include a lots of libraries that are not used in files, a library/file has to be included only if it is indispensable (it produces an error when you don't include it); the code is not very spaced out; at last, don't forget to externalize the definition of the method that does not hold on one line.
296232	4,00	Warnings	Warning: [L1] tools.h 5 ; nodes.cc 27,28 Warning: [L2] nodes.cc 171	As a whole, good project. Architecture and Encapsulation are respected throughout. Coding style is good, with only occasional indentation mistakes and wrapping lines (comments also count as wrapping lines). In nodes.cc lines 32,33 are definitely in the wrong place
296253	4,00	ok	Warning [L1] missing tab l.48 tools.h .	Bon rendu, attention projet.cc ne doit connaitre que ville.h car il ne sert qu'à spécifier le mode de fonctionnement du programme. Bonne mise en forme
296351	4,00		conventions respectées et code agréable à lire, continuez comme ça!	très bon rendu globalement, bonne continuation!

296443	4,00		Flawless style. Good job with the your new variable definitions.	I commend your attention to the style, very good job overall. If you correct the small issues mentionned you are in a very good spot. I would also advise you to consider implementing inheritance next for your nodes, this will help you for the graphical part. Good job.
296642	3,00	[L1] ville.cc 117 119,120, 132, 250,252	Break pas indentés, les instructions du case ne sont pas toujours identées de la même manière, même problème pour le if. Deux lignes pas indentées (250,252)	Bon code, clair. Essayez de rester constants sur les identations.
296820	3,00	[L1] noeud.h 27 ; ville.cc 88,102,116,... ; noeud.cc 68	Several instructions on the same line are not allowed.	As a whole, good project. Architecture is respected troughout. Encapsulation is violated in several cases due to definition of functions in .h, but is otherwise respected. Some indentation mistakes were unfortunately present.
296826	4,00	-	Pensez à aligner les affichages de vos cout (par exemple, noeud.cc:118) et à éviter les tabulations inutiles pour faire de l'alignement (par exemple, ville.h:38-47). Eviter les lignes de plus de 87 caractères (pas pénalisé car vous n'en aviez moins que 4). Sinon les entêtes e vos fichiers sont bien faites et le code est facilement parcourable !	Faites attention aux indentations dans la présentation du code (à tous les niveaux), même si vous ne perdez pas de points. Sinon continuez comme ça. Bon travail !

296831	3,00	[L1]noeud.cc,38,41,44,50,53,56,59,... 80,89,95,111,...	très nombreuses incohérences d'indentation.1) une définition de fonction commence à la première colonne sauf si elle est dans un namespace, dans ce cas elle est uniformément décalée à droite, commentaires compris. warning magic numbers noeud.cc:128 (justement il faut utiliser max_link)	Bonne décomposition du code mais le style d'indentation très variable le rend difficile à lire. Choix curieux de faire hériter lien de segment et noeud de cercle ; on dirait plutôt qu'un lien possède un segment ou qu'un noeud possède un cercle.
296853	4,00		Excellent code, parfaitement indenté	très bon rendu, continuez comme ça!
296887	4,00	Warnings	Warning: wrapping line noeud.cpp 201	As a whole, good project. Coding rules are consistently respected and the code is well spaced. Architecture is violated due to definition of class ville in module noeud. Encapsulation is respected throughout. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.
296932	3,00	[L1] noeud.cc 13, 44, 45, 255, 266, ... tools.cc 48-51	Warning: [L2] tools.h 20, 34; several lines of cout must be aligned to << not to "cout"; several instructions on the same line are not allowed	As a whole, good project. Architecture and encapsulation is respected throughout. Coding style is good, and generally also well indented, but there were still too many indentation mistakes.
297104	4,00	-	Pensez à aligner les affichages de vos cout (par exemple, noeud.cc:118) et à éviter les tabulations inutiles pour faire de l'alignement (par exemple, ville.h:38-47). Éviter les lignes de plus de 87 caractères (pas pénalisé car vous n'en aviez moins que 4). Sinon les entêtes e vos fichiers sont bien faites et le code est facilement parcourable !	Faites attention aux indentations dans la présentation du code (à tous les niveaux), même si vous ne perdez pas de points. Sinon continuez comme ça. Bon travail !
297186	3,00	[L1] ville.cc, noeud.cc ; [L2] ligne 22,34..	plusieurs styles d'indentation différents ce qui rend le tout un peu chaotique, warning: quelques lignes sont trop longues	globalement votre code est bien structuré mais il faut faire attention aux contraintes posées par la donnée. Bonne chance pour la suite!

297202	4,00	-	<p>La ligne est trop longue en noeud.cc:16. A part cela, votre code est bien présenté, bravo !</p>	<p>Vous définissez plusieurs fois les règles de compilation dans votre makefile. Sinon, vous pouvez tranquillement commencer le rendu suivant avec votre code !</p>
297280	4,00		<p>Très bon style bravo !</p>	<p>Très bon rendu bravo pour l'effort donné sur l'héritage et l'utilisation plus avancées de concepts de POO.</p>
297284	3,00	[L1] Noeud.cc 38-114, 187	<p>Vos placements d'accolades diffèrent beaucoup d'un endroit à un autre dans le projet. Veillez à rester constants. De plus, il est interdit d'écrire deux lignes sur une seule (par exemple 92). Définissez bien les valeurs que vous utilisez. Un lecteur externe aurait de la peine à comprendre à quoi vous comparez la variable compteur (dans Noeud.cc) corrigez tout cela et votre style sera excellent!</p>	<p>Code dans l'ensemble bon. Essayez d'éviter les includes inutiles par exemple de error.h et de mettre des "using namespace std" dans les .h. Veillez aussi à placer les "ifndef" et "define" tout en haut des fichiers .h</p>

297293	4,00	Warnings	Warning: wrapping line noeud.cpp 201	As a whole, good project. Coding rules are consistently respected and the code is well spaced. Architecture is violated due to definition of class ville in module noeud. Encapsulation is respected throughout. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.
297301	3,00	[L1] : projet.cc : 10, 11, ..., ville.cc: 67, 69, ville.h: 3 à 9, noeud.cc: 12 à 17, noeud.h: 11, 19, 36	Trop de petites erreurs d'indentation, c'est dommage ! Faites très attention au style de votre code, cela impacte tout de suite la visibilité. Vous vous perdez également dans les différentes notations pour les if sur une ligne : parfois avec les accolades, parfois sans, parfois l'instruction sur la ligne suivante, parfois sur la même ligne... Et ne mettez jamais le else sur la même ligne que le if !	Excellent code globalement, vous commettez l'erreur de ne pas vous servir de la classe ville, mais vous vous rendez compte que ce n'est pas viable et prévoyez de le corriger, j'ai donc mis un warning. Vous devrez avoir corrigé le souci pour le rendu 2 ! L'encapsulation est parfaitement respectée, à l'avenir vous pouvez essayer d'imaginer une solution qui se passe complètement de variables statiques, avec le vecteur de Noeud dans ville. Dommage que vous perdiez un malheureux petit point sur l'indentation, sinon votre rendu aurait été parfait ! Courage pour la suite !

297768	4,00		Très bon style bravo !	Très bon rendu bravo pour l'effort donné sur l'heritage et l'utilisation plus avancées de concepts de POO.
297781	3,00	[P5]iofile.cc:41,52,math.cc:96,math.h:30,ville.cc:27	Très bonne présentation mais contrôlez la taille maximale de vos lignes.	Vous créez plus de fichiers que demandés et vous vous compliquez un peu la vie. Vous avez en revanche un makefile minimalise très beau et le projet dans son ensemble est bon !
298006	4,00	ok	warning : le code pourrait etre un peu moin compacte et des commentaires seraient les bienvenus	Great job ! the code is well structured , easy to read and clean.

299248	4,00		code clair, bien indenté	très bon rendu, continuez comme ça!
299274	3,00	[L1] noeud.cc 73 (double indent);tools.cc 25, 59 (missalignment), 51, 85 (double indent); ville.cc 19 (double indent); ville.h missing indentation.	I see you gave attention to the style generally but at some occasions you did not, sorry. Try to stay consistent.	Your code is good. You use booleans to check if it is a Logement or not, this is probably a bad idea, consider using inheritance to implement all types of nodes. This will be specially useful when you begin the graphical part of the project. Otherwise good job!
299290	4,00		code clair et lisible,bon travail	bon travail globalement, attention pour la suite à ne pas utiliser les constantes (dist_min) dans le module tools car c'est pénalisé
299391	3,00	[L1] ville.cc : 62, 148; tools.h : inconsistent indentation of class (12- 32 and 39-51)	Your coding style is very good, the project is clean, well written and pleasant to read. Watch out for the two occurrences in ville.cc and for the inconsistency of the class indentation, you can choose whatever convention fits your style but try to be consistent through the code. Good job!	Excellent work! You may find useful to use inheritance instead of type checking for your Quartiers, especially considering the graphical part of the project.
299450	3,00	[L1] ville.cc 117 119,120, 132, 250,252	Break pas indentés, les instructions du case ne sont pas toujours identées de la même manière, même problème pour le if. Deux lignes pas indentées (250,252)	Bon code, clair. Essayez de rester constants sur les identations.
299579	4,00	ok		programme bien architecturé. l'indentation est très bonne mais le nom de certaines variables et fonctions rendent la lecture plus difficile. merci de rendre seulement le projet comme il est demandé (.cc .h et makefile uniquement)

299585	3,00	[L1] noeud.cc :67,68,176,178,190,191	Style is globally good, good effort considering all magic numbers and good style in general. A shame for the lack of attention in the mentioned lines that were incorrectly indented.	Good code in general. Watch out for what you include and if you really need it. You approached the problem by having your Ville module just read all lines and initialise everything in the Noeud module. This means your Ville module is only a parser and does not represent the Ville itself, since evrything is stored in Noeud. This is fine but maybe not the best logical approach. I also see the you do not use inheritance for your nodes and this could be useful in the next parts of the project. Good job!
299871	4,00		Très bon style bravo !	Très bon rendu bravo pour l'effort donné sur l'heritage et l'utilisation plus avancées de concepts de POO.
299885	4,00		code bien organisé mais très compact par endroits, essayez d'aérer un peu, éviter les les noms de fonctions en e1,e2,e3 et opter pour des noms plus "parlants"	Très bon travail jusque là, à part quelques détails Bonne continuation!

299971	4,00		rien à dire, code impeccable!	excellent dans l'ensemble, continuez comme ça!
300036	3,00	[L1] noeud.cc 126-129, 209-211, 220, tools.h 12-28, 39-44, 53-66	attention aux erreurs d'indentation	bon code avec une bonne architecture. vos méthodes de classes sont pertinentes. cependant, quelques noms de variables pourrait être plus indicatifs.
300038	4,00	[L1]noeud.cc:128	Bon style en général le code est propre, attention tout de meme vous avez deux fonctions qui sont presque à 80 lignes pensez à refactor	Bon rendu et joli style bravo
300048	3,00	[P5] noeud.cc 153, 156, 162, 167	warning : [L1] noeud.cc 225. Bonne indentation mais attention aux magic numbers !	bon code en général mais faites plus attention aux convention de programmation. Faites aussi attention, il ne doit pas y avoir de "using namespace" dans un .h
300086	2,00	[L1] noeud.cpp 88 ; city.cpp 192-227 ; [L2] all comments	several lines of cout must be aligned to << not to "cout"; several instructions on the same line are not allowed ; closing } is always, for all conventions, aligned with instruction opening { ; all comments are aligned left so that the last char is on col 88, but max col width is 87	As a whole, good project. Encapsulation is respected throughout. Architecture is respected with the only exceptions being error::success and argc in city. Coding style is good, with indentation mistakes mainly concentrated in one function and unfortunately many lines wrap beacuse you set your maximum column width to a wrong value.

300089	0,00			
300148	3,00	[L2]noeud.cc:436,446,456,474,475,483,... : 56,474,475,483,...	[L2]noeud.cc:436,446,456,474,475,483,... : Les lignes sont trop longues (max 87 caractères sinon passez à la ligne)	- Le rendu est bon. Bravo pour la bonne encapsulation des classes. Cependant, vous utilisez trois classes pour représenter les noeuds sans utiliser l'héritage, cela amène beaucoup de répétitions dans votre code. Je vous conseille d'utiliser l'héritage et de revoir votre organisation de classes.
300166	4,00		rien à dire, code impeccable!	excellent dans l'ensemble, continuez comme ça!

300247	3,00	[L1] 135,150,162,174 noeud.cc: more than one instruction in one line/ noeud.cc for example line 398 not well aligned,...etc... error done in a lot of different files at a lot of different places and for function parameters, if conditions, return, ...etc...	The indentation rules have been ignored, pay more attention to the conventions posted on moodle and be uniform with your coding style. Warning: [L2] 22 noeud.h; 3 noeud.cc: warning your line has more than 87 char. because of comments. [L2] 297 noeud.cc: the line contains more than 87 char.	Architecture is good, and you have comments in the .h and that's good, but they need to be more concise. There is a lot to improve: you need to externalize your definitions of the method of the class; there is no need to add ";" after each scope! it needs to be done only for a class or a struct scope; there is sometimes more than one instruction on one lines; the parameters of a function that hold in more than one line are not always well aligned as well as in "if" when it have more than one line condition; there is an "if" instruction in the main but it does nothing; you have created a namespace ville and noeud but it is ambiguous as it has the same name (just a first capital letter to distinguish them) as the namespace of the class; to finish you need to spaced out the code because it is sometimes difficult to read.
300259	3,00	[L1]noeud.cc:12,26,55,58,64,74,79...	-Incohérence de style d'accolades à multiples endroits : a travers le code vous mélangez deux styles d'utilisations d'accolades soyez plus cohérent pour le prochain rendu - Warning : n'indentés pas les mots clé private/public	- Très bon rendu bravo, faites attentions à la cohérence du style d'utilisation des accolades.
300299	3,00	[L1] noeud.cc 13, 44, 45, 255, 266, ... tools.cc 48-51	Warning: [L2] tools.h 20, 34; several lines of cout must be aligned to << not to "cout"; several instructions on the same line are not allowed	As a whole, good project. Architecture and encapsulation is respected throughout. Coding style is good, and genrally also well indented, but there were still too many indentation mistakes.
300370	3,00	[L1] 104,121,138,155 ville.cc: more than one instruction on one line.	Warning: constexpr created in noeud.cc and tools.cc but it is in the given constants (no_link); no Ville class; multiple brace style.	Overall the code is well structured. Just few words: don't try to save lines in the reading function by putting multiple instructions in the same line, you'd better delegate some tasks to sub-functions.

300371	2,00	[L1] 22,26,60,61,62,66,67,68 ,72,..... ville.cc: if else style not unique and coherent; 23,169,170,180,191 noeud.cc: long instructions in if not correctly aligned; 10,16 noeud.h, 16 ville.h: private/ public keywords indeted. [P2] 37-139 ville.cc: function of 102 lines	Warning: the class/struct name must have an uppercase first letter. The style is not uniform, pay more attention to conventions that are on moodle.	The code is well spaced out and pleasant to read even if there is sometimes a non- uniform style, the architecture is also good. Just a few comments: you have a lots of getters and they are not const, you have 3 setters in noeud.h but no constructor, and you have an "ifndef, def, endif" in a .cc file that is not supposed to be include in other files.
300454	2,00	[L1] 22,26,60,61,62,66,67,68 ,72,..... ville.cc: if else style not unique and coherent; 23,169,170,180,191 noeud.cc: long instructions in if not correctly aligned; 10,16 noeud.h, 16 ville.h: private/ public keywords indeted. [P2] 37-139 ville.cc: function of 102 lines	Warning: the class/struct name must have an uppercase first letter. The style is not uniform, pay more attention to conventions that are on moodle.	The code is well spaced out and pleasant to read even if there is sometimes a non- uniform style, the architecture is also good. Just a few comments: you have a lots of getters and they are not const, you have 3 setters in noeud.h but no constructor, and you have an "ifndef, def, endif" in a .cc file that is not supposed to be include in other files.
300780	4,00	-	Très bonne présentation. Excellent !	Il n'était pas nécessaire de recréer LINK_VACUUM en define: déjà présent dans constantes.h en constexpr. A part la classe Ville qui semble être inutile, c'est très bien ! bravo !
300840	4,00	Warnings	Warning: [L1] tools.h 5 ; nodes.cc 27,28 Warning: [L2] nodes.cc 171	As a whole, good project. Architecture and Encapsulation are respected throughout. Coding style is good, with only occasional indentation mistakes and wrapping lines (comments also count as wrapping lines). In nodes.cc lines 32,33 are definitely in the wrong place

300891	4,00	-	Vous ne faites aucune faute pour la présentation ! Pensez néanmoins à "aérer" votre code: les getters n'ont pas besoin d'être sur une seule ligne, vous pouvez faire un retour à la ligne après un if ou un else, même pour une instruction, etc.	Améliorez un peu la présentation du code ! A part ça, bravo !
300899	4,00	ok	Warning [L1] missing tab l.48 tools.h .	Bon rendu, attention projet.cc ne doit connaitre que ville.h car il ne sert qu'à spécifier le mode de fonctionnement du programme. Bonne mise en forme
300914	4,00		Good style, coherent throughout the code. You define the variable erreur_potentiel 2 times, once in tools.cc and another in constantes and this value is different each time. Watch out for a magic number at line 71 in noeud.cc.	I see you check for the .txt extension, it is a good initiative. Watch out since in Ubuntu extensions are not required so the check is not required. You should remove all includes that are not useful in your files. Great job otherwise, style is good and simple, architecture is correct and class encapsulation is respected. Consider adding inheritance next. Good job!
300987	3,00	[L1] noeud.cc 25,38,45,54,59,...: more than one instruction on one line / noeud.cc 97,99,172,174,178,...; tools.cc 10: long instructions in if, or instruction on multiple lines not correctly aligned.	The code has a non-uniform style sometimes there is some space sometimes not and when you have to write an instruction on multiple lines indent correctly with respect to conventions posted on moodle.	Architecture is OK even if you have included error and constant module in projet.cc, but there is a lot to improve: don't try to save lines in the reading function by putting multiple instructions in the same line, you'd better delegate some tasks to sub-functions; getters are not const; you also made an instance of class Ville and then break the encapsulation, you should declare it in the ville module; you need to externalize the definition of methods of the class; be careful having public attributes is forbidden; and to finish pay attention to indentation rules and conventions posted on moodle.
301070	4,00	ok	warning : in class public , private/public should not be indented	Great job ! the code is well structured , easy to read and clean , the comments makes it even better.
301271	4,00	OK	Warning:[L1] 22,30 Noeud.h and 14,22 Ville.h: public and private keywords are indented [L2] 32 Ville.cc: more than 87 char.	Good project overall, the architecture and the coding style are good even if there is few errors for the coding style. Just a few points: you forgot the return 0 in the main; you include a lots of libraries that are not used in files, a library/file has to be included only if it is indispensable (it produces an error when you don't include it); the code is not very spaced out; at last, don't forget to externalize the definition of the method that does not hold on one line.
301295	3,00	[P5] missing symbols in read_constr_ini & createObjects (among others).	Warning [L1] you must align the second line of arguments with the first character of the first line arguments.	Pas trop mal mais manque de rigueur attention à votre architecture et vos déclarations. ville doit être retravaillé. bon travail pour les autres modules, la mise en forme est bonne dans l'ensemble.

301319	4,00	ok	<p>warning : [E16] toutes vos fonctions testErreur n'indiquent pas clairement le but de la fonction ce qui n'est pas très pratique lors de la lecture du code. Pourquoi y a t-il tant de points virugules dans votre code ? [P5] ville.cc 45, vous pourriez utiliser des #define ou constexpr au lieu de ce commentaire</p>	<p>Bon programme avec une architecture pertinente, seulement faites attention aux noms de vos fonctions.</p>
301366	4,00	ok	<p>warning : in class public , private/public should not be indented</p>	<p>Great job ! the code is well structured , easy to read and clean , the comments makes it even better.</p>
301407	4,00	-	<p>Soyez plus consistents dans votre style d'indentation (parfois tab après {, parfois pas, par exemple). Vous avez quelques soucis d'indentation mais pas assez pour être pénalisé. Certaines de vos fautes n'entrent pas aussi dans les guidelines de notation de ce rendu. Contrôlez le document moodle des conventions de programmation !</p>	<p>Vous avez fait un bon rendu, mais il y a divers points (comme la présentation) que vous devriez revoir. Sinon, bravo !</p>
301438	3,00	[L1] All classes	<p>"public", "private", "protected" keywords are not supposed to be indented</p>	<p>You can just put the "using namespace std" at the top of your noeud.cc file instead of repeating it a lot. Very good use of the tools module and inheritance!</p>

301446	4,00	-	Soyez plus consistents dans votre style d'indentation (parfois tab après {, parfois pas, par exemple). Vous avez quelques soucis d'indentation mais pas assez pour être pénalisé. Certaines de vos fautes n'entrent pas aussi dans les guidelines de notation de ce rendu. Contrôlez le document moodle des conventions de programmation !	Vous avez fait un bon rendu, mais il y a divers points (comme la présentation) que vous devriez revoir. Sinon, bravo !
301450	3,00	[L1] ville.cc, noeud.cc ; [L2] ligne 22,34..	plusieurs styles d'indentation différents ce qui rend le tout un peu chaotique, warning: quelques lignes sont trop longues	globalement votre code est bien structuré mais il faut faire attention aux contraintes posées par la donnée. Bonne chance pour la suite!
301482	3,00	[P5] missing symbols in read_constr_ini & createObjects (among others).	Warning [L1] you must align the second line of arguments with the first character of the first line arguments.	Pas trop mal mais manque de rigueur attention à votre architecture et vos déclarations. ville doit être retravaillé. bon travail pour les autres modules, la mise en forme est bonne dans l'ensemble.
301496	4,00		code clair, bien indenté	très bon rendu, continuez comme ça!

301607	2,00	[L1] tools.cc: 21, 22, 38, 45, 63, ville.cc: 25, 219 [P2] ville.cc: decodage_ligne fait 305 lignes.	Vous avez bien compris le principe d'indenter chaque fois qu'un nouveau bloc débute, mais vous utilisez tous les styles d'indentation possible ! Un coup les for ont l'accolade sur la même ligne, un coup sur la suivante, un coup les if sur une seule ligne ont tout sur la même ligne, l'autre tout sur la ligne suivante, et un coup le bloc est ouvert classiquement. Par ailleurs, il convient de ne pas indenter public et private. Par ailleurs, votre fonction decodage_ligne fait 305 lignes au lieu du 80 maximal autorisé. Vous vous servez de cette fonction pour tout tester alors que dans chaque car il y a un endroit plus pertinent pour le faire. Par exemple, tester que l'UID d'un noeud n'est pas l'UID réservé peut se passer dans le constructeur de la classe Noeud	Vous vous donnez beaucoup de mal et ça se voit, malheureusement vous ne vous servez pas assez des principes de l'orienté-objet et perdez de nombreux points pour ça ! Par exemple, plutôt que d'utiliser 4 setters différents pour régler le x et le y des deux noeuds qui composent un lien, vous pouvez avoir une classe Cercle, dans tools, et chaque Noeud possède un Cercle, passé au constructeur, et chaque cercle est défini par son rayon et son centre, qui est une structure Point qui regroupe deux double représentant les coordonnées. Dès que vous avez une structure de données sur laquelle des manipulations doivent être faites, dites vous qu'il faut une classe. Passer cinq variables en argument plutôt qu'un objet qui les regrouperait toutes est une erreur, car suite à cela votre appel de fonction prend 4 lignes au lieu d'une et vous obtenez une fonction de 300 lignes au lieu de 80. Je pense que vous devriez repasser un moment pour assainir le code que vous avez là avant de vous lancer dans le rendu 2, car celui-ci sera très très compliqué en l'état. Courage, c'est beaucoup d'un coup, mais une fois ces soucis réglés vous verrez que tout deviendra limpide et simple d'utilisation !
--------	------	---	---	---

301653	3,00	[P2] ville.cc : automate makes more than 100 lines	Thank you for spacing-out your code, but sometimes it can be too mcuh like in your automate function, making it really too long. Code is very well written, with comments and attention to detail. Very good job	Very good project, architecture is fine and the approach to the problem is good. Well done making the default constructor of node not usable, this shows attention to detail. Very good job. Consider adding inheritance to your project for nodes.
301707	4,00	Ok	Attention toutefois, vous avez une seule erreur d'indentation : votre unique wrapping line devrait avoir son retour indenté	Très beau code dans l'ensemble, agréable à lire et bien travaillé, vous vous êtes donné du mal et ça se voit. Excellent et très classe usage du pattern Singleton pour la classe City. Si c'est volontaire c'est très propre (et si vous avez aucune idée de ce dont je parle, cherchez sur Google Singleton pattern). Seul petit reproche : vous usez de méthodes statiques pour vos classes Node et Link qui sont complètement évitables, essayez de repenser ces usages. Bravo sinon, c'est un très bon départ
301720	3,00	[L1] ville.cc 41-90	Inconsistent indentation of if-else, several instructions on same line	As a whole, good project. It is a shame for the forgotten inclusion in projet.cc. You should really break the lecture function in several smaller function because it is a small mess to read as to now. Architecture and encapsulation is respected throughout. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.
301765	4,00	-	Vous respectez les conventions de programmation mais pensez à un peu plus espacer votre code afin de rendre la lecture plus fluide. Par exemple, la fonction checkNodeLinkSuperposition pourrait avoir plus d'espace pour être plus lisible. De plus utilisez true/false au lieu de 1/0 (tools.cc:23-24)	Traitez de manière unique le cas <code>if(!(line[0] == '#'))</code> et améliorez votre présentation du code, sinon continuez sur cette lancée !

301786	4,00	Ok	Attention toutefois, vous avez une seule erreur d'indentation : votre unique wrapping line devrait avoir son retour indenté	Très beau code dans l'ensemble, agréable à lire et bien travaillé, vous vous êtes donné du mal et ça se voit. Excellent et très classe usage du pattern Singleton pour la classe City. Si c'est volontaire c'est très propre (et si vous avez aucune idée de ce dont je parle, cherchez sur Google Singleton pattern). Seul petit reproche : vous usez de méthodes statiques pour vos classes Node et Link qui sont complètement évitables, essayez de repenser ces usages. Bravo sinon, c'est un très bon départ
301816	3,00	[L1] noeud.cc,38,41,44,50,53,56,59,... 80,89,95,111,...	très nombreuses incohérences d'indentation.1) une définition de fonction commence à la première colonne sauf si elle est dans un namespace, dans ce cas elle est uniformément décalée à droite, commentaires compris. warning magic numbers noeud.cc:128 (justement il faut utiliser max_link)	Bonne décomposition du code mais le style d'indentation très variable le rend difficile à lire. Choix curieux de faire hériter lien de segment et noeud de cercle ; on dirait plutôt qu'un lien possède un segment ou qu'un noeud possède un cercle.
301834	3,00	[L1] (incorrect indentation of wrapping line) -> ville.cc 105, 254 ;ville.h 21, 28; noeud.h 10	Style is Ok, watch out for wrapping lines, you should check conventions on how to do it, sometimes you do it right (noeud.h), but most of the time you do it wrong.	There are a few problems but in general the code is Ok, you are in the right direction if you correct the mentioned problems. I see you defined some variables, this is fine, but you define "True" and "False", which are already defined by default, I do not understand the need for this. You may want to take a better look at how booleans work. Good job nonetheless, you may also consider implementing inheritance for the following stages of the project, good luck!
301837	3,00	[L1] ville.cc 41-90	Inconsistent indentation of if-else, several instructions on same line	As a whole, good project. It is a shame for the forgotten inclusion in projet.cc. You should really break the lecture function in several smaller function because it is a small mess to read as to now. Architecture and encapsulation is respected throughout. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.

301850	3,00	[L1] (incorrect indentation of wrapping line) -> ville.cc 105, 254 ;ville.h 21, 28; noeud.h 10	Style is Ok, watch out for wrapping lines, you should check conventions on how to do it, sometimes you do it right (noeud.h), but most of the time you do it wrong.	There are a few problems but in general the code is Ok, you are in a the right direction if you correct the mentioned problems. I see you defined some variables, this is fine, but you define "True" and "False", which are already defined by default, I do not understand the need for this. You may want to take a better look at how booleans work. Good job nonetheless, you may also consider implementing inheritance for the following stages of the project, good luck!
301863	3,00	[L1] : ville.h: 5, 7, ville.cc: 25, noeud.cc: 54, 96, 114	Trop de petites erreurs d'indentation vous coutent un point, c'est dommage ! Relisez bien les conventions ! Warning : vous utilisez deux styles pour les blocs d'une seule ligne : soit vous faites sans accolade sur la même ligne, soit vous mettez les accolades. Pensez à unifier ! Warning : la variable V dans noeud a un nom très discutable. Pensez à trouver plus parlant !	Bon projet, avec d'excellentes idées pour la manière de faire les calculs. Dommage que vous ne portiez pas plus d'attention à tout ce que peut apporter l'orienté-objet, car votre code serait bien plus agréable et facile d'utilisation pour vous si vous aviez une classe ville correcte, et que les interactions entre ville et noeud étaient un peu mieux utilisées. Par exemple : votre classe ville doit posséder un vecteur qui contient tous ses quartiers et réaliser la lecture, chaque fois qu'elle lit un nouveau Noeud, elle appelle le constructeur de Noeud pour obtenir un nouveau noeud, teste si ce noeud est un conflit avec l'un des noeuds précédemment ajouté dans la ville, et si tout va bien l'ajoute dans le vecteur. Je vous conseille de repasser sur ces parties du code, car bien encapsuler vous rendra la tâche bien plus facile pour les rendus à venir ! Courage !
301873	4,00		warning : [E16] toutes vos fonctions testErreur n'indiquent pas clairement le but de la fonction ce qui n'est pas très pratique lors de la lecture du code. Pourquoi y a t-il tant de points virgules dans votre code ? [P5] ville.cc 45, vous pourriez utiliser des #define ou constexpr au lieu de ce commentaire	Bon programme avec une architecture pertinente, seulement faites attention aux noms de vos fonctions.

301900	3,00	[L1] All classes	"public", "private", "protected" keywords are not supposed to be indented	You can just put the "using namespace std" at the top of your noeud.cc file instead of repeating it a lot. Very good use of the tools module and inheritance!
301927	4,00		Lorsque vos lignes dépassent les 87 caractères et que vous faites un retour à la ligne, veillez à toujours la rabattre de la même manière. Aérer le code entre vos fonctions pourrait le rendre légèrement plus agréable à la lecture mais dans l'ensemble votre style respecte très bien les conventions.	On peut observer dans votre code des restrictions mises en place et pensées pour la suite du projet (notamment les delete et les const). C'est une très bonne pratique, continuez comme ça! Vous montrez une bonne compréhension des concepts appris avec votre mise en place de l'héritage qui servira plus tard. Faites tout de même attention aux méthodes qui devront être virtuelles ou pas !
301940	3,00	[L1] noeud.cc 73 (double indent);tools.cc 25, 59 (missalignment), 51, 85 (double indent); ville.cc 19 (double indent); ville.h missing indentation.	I see you gave attention to the style generally but at some occasions you did not, sorry. Try to stay consistent.	Your code is good. You use booleans to check if it is a Logement or not, this is probably a bad idea, consider using inheritance to implement all types of nodes. This will be specially useful when you begin the graphical part of the project. Otherwise good job!
301950	0,00			
301964	4,00	ok	essayer de garder le même style d'accolade dans tout votre code. excellente indentation le code est clair et agréable à lire, malgré quelques noms de variables qui pourrait être plus parlants	excellent code, très bonne architecture. continuez comme ça
302023	4,00	-	La ligne est trop longue en noeud.cc:16. A part cela, votre code est bien présenté, bravo !	Vous définissez plusieurs fois les règles de compilation dans votre makefile. Sinon, vous pouvez tranquillement commencer le rendu suivant avec votre code !

302038	3,00	[P5]ville.cc:41,53,57,61,65,70,75,83	<p>Les états de l'automate de lecture sont des entiers de 1 à 7 hardcodé. Il faut créer des constantes pour les définir (les constantes peuvent bien évidemment être des entiers mais le nom permet d'informer au lecteur l'état concerné), Cela rend le code de l'automate très difficile à comprendre en tant que lecteur. + quand vous testez l'erreur "reserved_uid" utilisez des constantes au lieu d'hardcoder 4294967294 (d'ailleurs déjà définie dans constantes.h "no_link") et 770. Mise à part cela le style du code est bon et très lisible alors Bravo !</p>	<p>Bon rendu, malheureusement vous perdez 2 points à cause de petits détails. Faites attention avec le stockage de la liste des noeuds dans un espace non-nommé, je pense que ça risque de vous amener des complications pour les prochaines étapes</p>
302153	3,00	[L1] 104,121,138,155 ville.cc: more than one instruction on one line.	<p>Warning: constexpr created in noeud.cc and tools.cc but it is in the given constants (no_link); no Ville class; multiple brace style.</p>	<p>Overall the code is well structured. Just few words: don't try to save lines in the reading function by putting multiple instructions in the same line, you'd better delegate some tasks to sub-functions.</p>
302297	4,00	ok		<p>programme bien architecturé. l'indentation est très bonne mais le nom de certaines variables et fonctions rendent la lecture plus difficile. merci de rendre seulement le projet comme il est demandé (.cc .h et makefile uniquement)</p>

302394	4,00	ok	warning : le code pourrait etre un peu moins compacte et des commentaires seraient les bienvenus	Great job ! the code is well structured , easy to read and clean.
302397	3,00	[L1] Noeud.cc 38-114, 187	Vos placements d'accolades diffèrent beaucoup d'un endroit à un autre dans le projet. Veuillez à rester constants. De plus, il est interdit d'écrire deux lignes sur une seule (par exemple 92). Définissez bien les valeurs que vous utilisez. Un lecteur externe aurait de la peine à comprendre à quoi vous comparez la variable compteur (dans Noeud.cc) corrigez tout cela et votre style sera excellent!	Code dans l'ensemble bon. Essayez d'éviter les includes inutiles par exemple de error.h et de mettre des "using namespace std" dans les .h. Veuillez aussi à placer les "ifndef" et "define" tout en haut des fichiers .h.
302457	4,00		warning: ville.cc & noeud.cc : functions that are restricted to a module are first only DECLARED in the unnamed namespace at the beginning of the file (O14) ; then they are DEFINED at the end of the file (O16.3)	very clear decomposition of the file reading operation. The code is very pleasant to read owing to good function and variable names, and well-structured code..
302508	4,00		Flawless style. Good job with the your new variable definitions.	I commend your attention to the style, very good job overall. If you correct the small issues mentionned you are in a very good spot. I would also advise you to consider implementing inheritance next for your nodes, this will help you for the graphical part. Good job.

302576	4,00		code bien organisé mais très compact par endroits, essayez d'aérer un peu, éviter les les noms de fonctions en e1,e2,e3 et opter pour des noms plus "parlants"	Très bon travail jusque là, à part quelques détails Bonne continuation!
302680	4,00		Excellent code, parfaitement indenté	très bon rendu, continuez comme ça!
302699	4,00		conventions respectées et code agréable à lire, continuez comme ça!	très bon rendu globalement, bonne continuation!
302808	3,00	[L1] noeud.cc :67,68,176,178,190,191	Style is globally good, good effort considering all magic numbers and good style in general. A shame for the lack of attention in the mentioned lines that were incorrectly indented.	Good code in general. Watch out for what you include and if you really need it. You approached the problem by having your Ville module just read all lines and initialise everything in the Noeud module. This means your Ville module is only a parser and does not represent the Ville itself, since evrything is stored in Noeud. This is fine but maybe not the best logical approach. I also see the you do not use inheritance for your nodes and this could be useful in the next parts of the project. Good job!

302875	3,00	[P5] noeud.cc 153, 156, 162, 167	warning : [L1] noeud.cc 225. Bonne indentation mais attention aux magic numbers !	bon code en général mais faites plus attention aux convention de programmation. Faites aussi attention, il ne doit pas y avoir de "using namespace" dans un .h
302888	2,00	[L1] tools.cc: 21, 22, 38, 45, 63, ville.cc: 25, 219 [P2] ville.cc: decodage_ligne fait 305 lignes.	Vous avez bien compris le principe d'indenter chaque fois qu'un nouveau bloc débute, mais vous utilisez tous les styles d'indentation possible ! Un coup les for ont l'accolade sur la même ligne, un coup sur la suivante, un coup les if sur une seule ligne ont tout sur la même ligne, l'autre tout sur la ligne suivante, et un coup le bloc est ouvert classiquement. Par ailleurs, il convient de ne pas indenter public et private. Par ailleurs, votre fonction decodage_ligne fait 305 lignes au lieu du 80 maximal autorisé. Vous vous servez de cette fonction pour tout tester alors que dans chaque car il y a un endroit plus pertinent pour le faire. Par exemple, tester que l'UID d'un noeud n'est pas l'UID réservé peut se passer dans le constructeur de la classe Noeud	Vous vous donnez beaucoup de mal et ça se voit, malheureusement vous ne vous servez pas assez des principes de l'orienté-objet et perdez de nombreux points pour ça ! Par exemple, plutôt que d'utiliser 4 setters différents pour régler le x et le y des deux noeuds qui composent un lien, vous pouvez avoir une classe Cercle, dans tools, et chaque Noeud possède un Cercle, passé au constructeur, et chaque cercle est défini par son rayon et son centre, qui est une structure Point qui regroupe deux double représentant les coordonnées. Dès que vous avez une structure de données sur laquelle des manipulations doivent être faites, dites vous qu'il faut une classe. Passer cinq variables en argument plutôt qu'un objet qui les regrouperait toutes est une erreur, car suite à cela votre appel de fonction prend 4 lignes au lieu d'une et vous obtenez une fonction de 300 lignes au lieu de 80. Je pense que vous devriez repasser un moment pour assainir le code que vous avez là avant de vous lancer dans le rendu 2, car celui-ci sera très très compliqué en l'état. Courage, c'est beaucoup d'un coup, mais une fois ces soucis réglés vous verrez que tout deviendra limpide et simple d'utilisation !

302910	2,00	[L1][P5.2]ville_2.h l34 , l24 , l41 noeud.h l24,l22,l25	dans tes classes private/public ne doit pas etre indented , si tu casse ta ligne a cause de trop de parametre , n'oublie pas l'indentation. Donnez des noms plus significatif que : const int trois(3) , size_t un(1) . Vous n etes pas consistant avec vos {}, choisissez un style.	Bon travail , on voit que vous avez passe du temps dessus , mais faites attention a "netoyer" ton rendu , les .o et executables sont pas necessaire , les fonctions pour debuguer ne doivent pas etre appelle au moment de rendre le code
302939	4,00	ok	-	Bon projet. Mise en forme claire, juste la structure s2d qui n'a pas un nom explicite.
302953	4,00	ok	public/private ne doivent pas etre indente . les methodes qui ne changent pas les attributs de la classe doivent etre const. warning : vous pouvez initialiser vos structs en 1 ligne comme : " Point p = {0,0} " au lieu de modifier chaque attribut un a un	Bon travail ! vous avez un tres bon code mais faites attention au dernier Warning
303000	4,00		code clair et lisible,bon travail	bon travail globalement, attention pour la suite à ne pas utiliser les constantes (dist_min) dans le module tools car c'est pénalisé
303084	3,00	[L1] ville.cc : 62, 148; tools.h : inconsistent indentation of class (12-32 and 39-51)	Your coding style is very good, the project is clean, well written and pleasant to read. Watch out for the two occurrences in ville.cc and for the inconsistency of the class indentation, you can choose whatever convention fits your style but try to be consistent throughout the code. Good job!	Excellent work! You may find useful to use inheritance instead of type checking for your Quartiers, especially considering the graphical part of the project.
303114	4,00		warning: ville.cc & noeud.cc : functions that are restricted to a module are first only DECLARED in the unnamed namespace at the beginning of the file (O14) ; then they are DEFINED at the end of the file (O16.3)	very clear decomposition of the file reading operation. The code is very pleasant to read owing to good function and variable names, and well-structured code..

303207	4,00	[L1]/[L2] tools.cc 8-9 - Incorrect wrapping line indentation	Very good style, watch out for the mentioned violation but you do not lose points for it.	Careful when using one liner ifs, I advise you to keep these in one line or to add a body block (using {}) otherwise you may forget there are no braces when adding code and have semantical issues that are a pain to find. Otherwise, excellent code, but watchout for the architectural problems. Good job!
--------	------	--	--	---

303227	4,00	[L1]/[L2] tools.cc 8-9 - Incorrect wrapping line indentation	Very good style, watch out for the mentioned violation but you do not lose points for it.	Careful when using one liner ifs, I advise you to keep these in one line or to add a body block (using <code>{}</code>) otherwise you may forget there are no braces when adding code and have semantical issues that are a pain to find. Otherwise, excellent code, but watchout for the architectural problems. Good job!
303778	3,00	[L1] 135,150,162,174 noeud.cc: more than one instruction in one line/ noeud.cc for example line 398 not well aligned,...etc... error done in a lot of different files at a lot of different places and for function parameters, if conditions, return, ...etc...	The indentation rules have been ignored, pay more attention to the conventions posted on moodle and be uniform with your coding style. Warning: [L2] 22 noeud.h; 3 noeud.cc: warning your line has more than 87 char. because of comments. [L2] 297 noeud.cc: the line contains more than 87 char.	Architecture is good, and you have comments in the .h and that's good, but they need to be more concise. There is a lot to improve: you need to externalize your definitions of the method of the class; there is no need to add ";" after each scope! it needs to be done only for a class or a struct scope; there is sometimes more than one instruction on one lines; the parameters of a function that hold in more than one line are not always well aligned as well as in "if" when it have more than one line condition; there is an "if" instruction in the main but it does nothing; you have created a namespace ville and noeud but it is ambiguous as it has the same name (just a first capital letter to distinguish them) as the namespace of the class; to finish you need to spaced out the code because it is sometimes difficult to read.
309996	4,00	ok	warning : [L1] essayez de rester constants dans vos styles d'indentation et d'écriture en général.	très bon programme dans l'ensemble. Il pourrait parfois être un peu plus aéré pour améliorer la lisibilité. Bon usage des #define

310030	4,00	ok	Warning [L1] you must align the second line of arguments with the first character of the first line arguments. Warning [L2] tools.cc 1.5 . Warning [P5] poor choice of name for s2d struct.	Bon rendu, quelques oublis de mise en forme mais bien dans l'ensemble. Faites attention au double fichier avec main et la ligne 1 de tools.h a une faute de frappe (toots.h).
310074	4,00	Warnings	Warning: [L1] tools.cc 25: variable names not aligned across different lines. I would suggest doing several separate declarations. Warning: [L1] noeud.cc 42,51: function parameters not aligned across several lines.	As a whole, good project. Indentation is sometimes wrong, and I personally would prefer a more spaced out code, but as to now it is still ok. Architecture and encapsulation is respected throughout.
310075	3,00	[L1] noeud.cc 23 ville.cc 125, 126, 144	Il est interdit de mettre plusieurs instructions sur la même ligne. Le seul cas accepté étant une instruction de contrôle suivie par l'instruction qui en découle. On évite donc les if/else sur la même ligne et plusieurs affectations sur la même ligne.	Très bon code agréable à lire et clair. Evitez de trop compacter les instructions sur la suite de votre code!
310087	3,00	[L1] noeud.cc 171,179,149, 195, 196, tools.h 9, 14	Attention to the indentations	As a whole, good project. Architecture and encapsulation is respected throughout. Coding style is good, except for several indentation mistakes.
310104	4,00	ok	warning : [L1] essayez de rester constants dans vos styles d'indentation et d'écriture en général.	très bon programme dans l'ensemble. Il pourrait parfois être un peu plus aéré pour améliorer la lisibilité. Bon usage des #define
310106	4,00	ok	Warning [L1] a tabulation for the : part of a constructor is a good practice	Beau rendu. Belle mise en page pas forcément standard mais consistente (le tab après la condition des if est un peu déroutant quoiqu'agréable. Les commentaires sont bien placés et justifiés. Faites par contre attention à l'usage de namespace std c'est toujours un peu dangereux.
310108	0,00			

310128	3,00	[L1]tools.cc:159;167, [L1]ville.cc:208-210,230-232	A part quelques oublis, le code très bien présenté. Grande uniformité dans la présentation et facile à lire ! Bravo !	Vous avez fait un excellent rendu. Le code est bien présenté et semble (sans exécution) optimal. Vous utilisez try/catch correctement. Bravo à vous !
--------	------	---	---	---

310261	3,00	[P5] ville.cc: 50 à 55, noeud.cc: 116 à 119, 127	<p>Vous utilisez trop de magic number ! Pourquoi return 1 plutôt que true pour un booléen, et vous utilisez sans cesse des tableaux de deux ou trois éléments sans que l'on comprenne pourquoi. Remplacez ces magic numbers par des symboles ou des constexpr (par exemple si j'ai un tableau coords de deux Points qui représente des coordonnées x et y, je fais les define pour pouvoir écrire coords[X] et coords[Y] plutôt que coords[0] et coords[1]) Warning : vous êtes inconsistant dans les styles d'indentation. Les if sur une ligne sont traités de deux manières différentes (noeud.cc: 116, tools.cc: 27), les for également (ville.cc: 57 et 58), et vous oubliez d'indenter les listes d'initialisations (noeud.cc: 23 à 36) A corriger pour le rendu 2 !</p>	<p>Plutôt bon projet dans le style et agréable à lire, malheureusement vous perdez pas mal de points car vous ne respectez pas les principes d'encapsulation. Cela risque de vous complexifier la tâche pour les rendus suivants : Ville doit être la seule à agir sur les vecteurs qui représente ses quartiers ! Elle lit le fichier, récupère les infos pour appeler un constructeur de Noeud, teste si ce noeud est en erreur avec l'un de ceux précédemment ajouté, puis l'ajoute si tout va bien. Pourquoi vouloir déléguer ces traitements à Noeud ? Si vous arrivez à corriger tout ça, vous serez en première ligne pour obtenir une note parfaite aux deux prochains rendus ! Courage !</p>
310271	4,00	Ok	Ok	<p>Excellent travail, ça pète ! De l'héritage aux petits oignons, des classes pour tout, une parfaite maîtrise de l'orienté-objet et du C++, j'ai pas grand chose à dire. Normalement vous devriez avoir besoin d'une autorisation pour démultiplier le nombre de fichiers, toutefois vous avez fait ça habilement et tous les fichiers du module tools respectent leur principe "bas niveau"... Pour être vraiment nickel, il faudrait éviter toute fonction hors-classe : vous pouvez le faire ! Votre fonction removeNode ne sert qu'à ce que la Ville puisse supprimer un noeud, ce qu'elle peut aisément faire en le retirant de son vecteur, le destructeur de Noeud peut se charger de prévenir les voisins pour une déletion parfaite des liens, et les fonctions de tools peuvent devenir des méthodes des classes correspondantes. A vous de voir ! Encore bravo en tout cas, je suis impressionné</p>
310276	3,00	[L1] Noeud.h 13, 27 Noeud.cc 69-75	<p>Style très bon dans l'ensemble. Informations à noter; on n'indente pas les mots clés public/private et votre fonction dans "Noeud.cc" possède des indentations de longueur différente des autres.</p>	<p>Code bien pensé pour correspondre à la suite du projet. Ajouter une classe ville serait une option à considérer pour rester dans une approche plus orientée objet. Je vous rend aussi attentifs sur le fait que le mode de lancement "console" n'est propre qu'à la version "demo" du projet et n'est pas demandé.</p>

310279	4,00	ok	<p>warning : vous pouvez initialiser vos structs en 1 ligne comme : " S2D origine = {0,0} " au lieu de modifier chaque attribut un a un</p>	<p>Tres bon travail ! le code est agrable a lire , prenez en compte le dernier warning pour ameliorer votre code.</p>
310295	4,00		<p>Bon respect de l'indentation hormis quelques détails: - noeud.h les mots clé public des sous classes ne devraient pas être indentés - tools.cc:43 la ligne fait 89 caractères attention - ville.cc:26 la ligne est trop longue et l'usage des accolades inconsistents, dans ce cas la vous pouvez juste passer à la ligne - Attention à l'automate qui est à la limite par rapport aux nombres de lignes, p-e définissez des fonctions pour chaque état</p>	<p>Dans l'ensemble très bon rendu et le code est agréable à lire Bravo ! Faites attention tout de mêmes aux petits détails d'indentation et longueurs de lignes</p>
310349	3,00	<p>[L1] : noeuds.cc: 22, 269, noeuds.h: 31, tools.cc: 19, ville.h: 25 et suivantes,</p>	<p>Plusieurs petites erreurs d'indentation, ou des points virgules après des accolades qui ne servent à rien. Warning : chaîner des for et des if sans accolades c'est globalement une mauvaise idée si le dernier maillon de la chaîne doit avoir plusieurs lignes. Votre fonction e05() est un excellent exemple du problème que ça pose</p>	<p>Votre code n'est pas désagréable à lire et on sent les efforts que vous y faites, malheureusement vous restez ancrés dans les principes de programmation séquentielle et tordez l'orienté objet dans tous les sens pour qu'il soit le plus séquentiel possible. Vos classes ne servent qu'à contenir des données, vous n'utilisez pas du tout les principes d'encapsulation ni n'essayez de faire en sorte que vos classes représentent quelque chose. Une ville est représentée par ses quartiers, un noeud par son type, sa position et sa taille ! Cela n'a donc aucun sens de mettre le vecteur de noeud ailleurs que dans ville, ni de laisser les noeuds gérer la construction de la ville (en lisant le fichier). Je vous conseille vivement de repasser sur votre code, mettre dans ville ce qui devrait y être et utiliser les principes de POO, car sans cela le rendu 2 risque d'être un enfer. Courage, votre code est bon et vos idées bonnes, il n'y a plus qu'à les mettre en forme !</p>

310409	4,00	Warnings	Warning: [L1] ville.cc 20,112 ; ville.h 27	As a whole, good project. Architecture and Encapsulation are respected throughout. Coding style is good, with only occasional indentaion mistakes. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.
310410	4,00	ok	warning : [P5] attention aux magic numbers (36 projet.cc, 93, 95 ville.cc) ; [L1] essayez de rester constants dans votre style d'indentation	bon programme, les noms des variables et constexpr sont pertinents.
310447	2,00	[P5]noeud.cc:19,25,28,30,35,...;ville.cc:52,67,82,... [L1]noeud.cc:16-52,57-161;ville.cc:22	<p>Vous pouvez utiliser true et false au lieu de 1 et 0 comme valeur de traitement. if(... == 1) peut s'écrire en cas de return de bool if(...) !</p> <p>N'utilisez pas sans constante la valeur 4294967295 (il y a déjà une constexpr: no_link et faites un cast ou créer un constexpr pour cette valeur)</p>	Corriger les quelques soucis que vous avez avec les conventions de programmations, notamment utilisez plus de symboles pour éviter les magic number et faites attention à la double indentation. Du point de vue fonctionnel, continuez comme cela !

310572	3,00	[L1] : noeuds.cc: 22, 269, noeuds.h: 31, tools.cc: 19, ville.h: 25 et suivantes,	Plusieurs petites erreurs d'indentation, des points virgules après des accolades qui ne servent à rien. Warning : chaîner des for et des if sans accolades c'est globalement une mauvaise idée si le dernier maillon de la chaîne doit avoir plusieurs lignes. Votre fonction e05() est un excellent exemple du problème que ça pose	Votre code n'est pas désagréable à lire et on sent les efforts que vous y faites, malheureusement vous restez ancrés dans les principes de programmation séquentielle et tordez l'orienté objet dans tous les sens pour qu'il soit le plus séquentiel possible. Vos classes ne servent qu'à contenir des données, vous n'utilisez pas du tout les principes d'encapsulation ni n'essayez de faire en sorte que vos classes représentent quelque chose. Une ville est représentée par ses quartiers, un noeud par son type, sa position et sa taille ! Cela n'a donc aucun sens de mettre le vecteur de noeud ailleurs que dans ville, ni de laisser les noeuds gérer la construction de la ville (en lisant le fichier). Je vous conseille vivement de repasser sur votre code, mettre dans ville ce qui devrait y être et utiliser les principes de POO, car sans cela le rendu 2 risque d'être un enfer. Courage, votre code est bon et vos idées bonnes, il n'y a plus qu'à les mettre en forme !
310573	4,00		Votre style est bon. Warning on remarque une légère discontinuité dans vos placements d'accolades, essayez de rendre le style uniforme au maximum.	Il serait bon de spécifier dans vos prototypes des noms de paramètres. Un lecteur de vos interfaces aurait de la peine à déterminer quel type correspond à quoi. Mis à part cela votre code est assez bien pensé et préparé pour la suite.
310576	4,00	ok	Warning [L1] a tabulation for the : part of a constructor is a good practice	Beau rendu. Belle mise en page pas forcément standard mais consistente (le tab après la condition des if est un peu déroutant quoiqu'agréable. Les commentaires sont bien placés et justifiés. Faites par contre attention à l'usage de namespace std c'est toujours un peu dangereux.
310608	3,00	[P5]ville.cc:77,78,81,82, 85,86,89,...	Utilisez des enums pour éviter les magic numbers ! Vous pouvez aussi écrire vos getters sur d'avantages de lignes. Les autres règles des conventions de programmation sont respectées !	C'est un bon projet mais pensez à corriger les divers points énumérés aux sections précédentes.

310616	3,00	[P5] noeud.h : 26, 27, 35, 36, 37, 38	Vos noms de fonctions doivent être bien plus parlants ! Vous déclarez en apparence quatre fois la même fonction en changeant juste son numéro, il faut que l'on comprenne ce qu'elle fait rien qu'en lisant sa signature Warning : vous devez utiliser le même style de majuscule dans tout le fichier : soit camelCase, soit snake_case. La fonction addnode ne rentre dans aucune des deux cases, à corriger	Très bon projet dans l'ensemble, félicitations ! Dommage que vous fassiez cette petite erreur, mais elle sera aisément corrigée. Pensez également à retirer les <code>ifndef</code> des <code>.cc</code> , car ces macros de protection ne sont utiles que pour les <code>.h</code> et éviter les inclusions infinies. Vous n'êtes pas loin d'un rendu parfait, et vous avez parfaitement compris le principe de l'orienté-objet, continuez !
310635	4,00		conventions respectées et code agréable à lire, continuez comme ça!	attention pour la suite à ne pas utiliser les constantes (<code>dist_min</code>) dans le module <code>tools</code> car c'est pénalisé, sinon très bon rendu!
310672	4,00	OK	Warning: [L2] 157 noeud.cc: line contains more than 87 char.	Great delegation of tasks to sub_functions ! The architecture is ok and the coding style is nice and uniform. Few comments: "lecture_fichier" module is very dependent on "ville" module and then it is not necessary to add this module; also in this module there is a struct and then be very careful because it means that all the "attributes" are here in "public".
310674	3,00	[L1] Noeud.h 13, 27 Noeud.cc 69-75	Style très bon dans l'ensemble. Informations à noter; on n'indent pas les mots clés <code>public/private</code> et votre fonction dans "Noeud.cc" possède des indentations de longueur différente des autres.	Code bien pensé pour correspondre à la suite du projet. Ajouter une classe <code>ville</code> serait une option à considérer pour rester dans une approche plus orientée objet. Je vous rend aussi attentifs sur le fait que le mode de lancement "console" n'est propre qu'à la version "demo" du projet et n'est pas demandé.
310741	4,00	Ok	Warning : vous utilisez autant le <code>snake_case</code> que le <code>camelCase</code> pour vos majuscules, vous arrivez même à utiliser les deux en une ligne (noeud.h : 23) décidez-vous et utilisez l'un ou l'autre. A corriger	Excellent projet, code très clair et lisible, quelques petites erreurs sans conséquences à droite à gauche. Je dois vous avouer que je comprenais pas l'intérêt de vos <code>#ifndef</code> dans le code. J'ai fini par piger et j'ai bien ri : <code>ifndef</code> laisse le code visible si la variable n'est PAS définie. Donc tout aurait du planter chez vous, sauf que vous faites <code>#ifndef rendu1</code> et <code>#define RENDU1</code> , et du coup ça passe car <code>rendu1</code> n'est bien pas défini ! Vous auriez du utiliser <code>#ifdef</code> pour votre idée (pas mauvaise au demeurant). Par ailleurs vous pouvez mettre le <code>#define</code> ailleurs que dans une fonction : ce n'est pas du code C++ à proprement parler, ça va en début de fichier. Par ailleurs, vous n'avez pas de classe <code>Cercle</code> , or elle sera vraiment pratique pour tout ce qui est dessin de GUI, je vous conseille d'en faire une. Excellent sinon, et bravo !

310786	2,00	[P5]noeud.cc:19,25,28,30,35,...;ville.cc:52,67,82,... [L1]noeud.cc:16-52,57-161;ville.cc:22	<p>Vous pouvez utiliser true et false au lieu de 1 et 0 comme valeur de traitement. if(... == 1) peut s'écrire en cas de return de bool if(...) !</p> <p>N'utilisez pas sans constante la valeur 4294967295 (il y a déjà une constexpr: no_link et faites un cast ou créer un constexpr pour cette valeur)</p>	<p>Corriger les quelques soucis que vous avez avec les conventions de programmations, notamment utilisez plus de symboles pour éviter les magic number et faites attention à la double indentation. Du point de vue fonctionnel, continuez comme cela !</p>
310832	3,00	[L1]: noeud.cc: 81, 140, ville.cc: 46, 85	<p>Indentations oubliées et if else sur la même ligne (inconsistant avec d'autre if else sur une seule ligne dans le même code). Par ailleurs, vous avez fait le choix d'aligner vos wrapping lines avec ce qu'il y a au-dessus, et c'est un choix dommageable. Cela crée des cas comme dans noeud.cc : 102 où 4 lignes supplémentaires sont requises. En général, 2 indentations suffises</p>	<p>Bon projet dans l'ensemble mais vous passez à côté de ce qui fait la force de l'orienté objet. Toutes vos classes ne servent qu'à regrouper des données et y accéder (donc sont équivalentes à des struct), car toutes les fonctions qui devraient être des méthodes de ces classes sont définies en-dehors. Cela crée des cas absurdes, comme votre fonction createur_neoud qui est littéralement un constructeur de la classe Noeud, sauf qu'en plus elle doit avoir en argument le vecteur de Noeud entier ! Je vous conseille fortement avant de vous atteler au prochain rendu d'essayer de corriger ça, vous ne devriez avoir des fonctions en dehors de classe que dans tools et éventuellement la fonction de lecture de ville. Courage pour la suite !</p>

310847	4,00	Ok	Warning : vous utilisez autant le snake_case que le camelCase pour vos majuscules, vous arrivez même à utiliser les deux en une ligne (noeud.h : 23) décidez-vous et utilisez l'un ou l'autre. A corriger	Excellent projet, code très clair et lisible, quelques petites erreurs sans conséquences à droite à gauche. Je dois vous avouer que je comprenais pas l'intérêt de vos #ifndef dans le code. J'ai fini par piger et j'ai bien ri : ifndef laisse le code visible si la variable n'est PAS définie. Donc tout aurait du planter chez vous, sauf que vous faites #ifndef rendu1 et #define RENDU1, et du coup ça passe car rendu1 n'est bien pas défini ! Vous auriez du utiliser #ifdef pour votre idée (pas mauvaise au demeurant). Par ailleurs vous pouvez mettre le #define ailleurs que dans une fonction : ce n'est pas du code C++ à proprement parler, ça va en début de fichier. Par ailleurs, vous n'avez pas de classe Cercle, or elle sera vraiment pratique pour tout ce qui est dessin de GUI, je vous conseille d'en faire une. Excellent sinon, et bravo !
310863	3,00	[L1] Dans toutes vos classes, ville.cc 36-91	Les mots clefs "public " et "private" ne doivent pas être indentés, on ne met pas deux instructions if/else sur la même ligne. Attention à indenter tous vos cases et les instructions de la même manière. Pensez à utiliser les mots-clés true et false à la place de 1 et 0 pour gagner en clareté.	Code bon dans l'ensemble. Attention, ajoutez un commentaire dans votre module Connexion pour en définir l'utilité et la raison pour laquelle vous avez choisi de l'ajouter, d'un point de vue externe il semble complexifier votre code sans justification.
310978	4,00	ok	-	Bon rendu, attention à l'encapsulation dans noeud sinon c'est bien
310986	4,00	ok	Warning [L1] you must align the second line of arguments with the first character of the first line arguments. Warning [L2] tools.cc l.5 . Warning [P5] poor choice of name for s2d struct.	Bon rendu, quelques oublis de mise en forme mais bien dans l'ensemble. Faites attention au double fichier avec main et la ligne 1 de tools.h a une faute de frappe (toots.h).
311013	4,00	ok	Warning : comments should not be avoided	Great job ! the code is well structured , easy to read and clean.

311043	2,00	[L1] ville.h; the class is indented entirely/ 59-132 ville.cc; in the switch, the case are not indented/ 57-58 tools.cc; long instructions in if not correctly aligned/ noeud.h; public and private keywords are indented and the methods/attributes of the class are indented 2 times/ 38-39,44-45 noeud.cc; the instructions in the functions are indented 2 times, etc.... [P2] 48-133 ville.cc: function of 85 lines	The indentation rules have been ignored, pay more attention to the conventions posted on moodle and be uniform with your coding style. Warning: [L2] 31 ville.h: more than 87 char.	Overall the architecture and the structure of the project is good, but some improvements need to be done on the coding style mainly. Few comments: the getters should be const; non-uniform calling for bool sometimes you call "true" 1 and sometimes true and same thing for "false"; tools.h is included 2 times in noeud.h; you made an ifstream that you don't use in the main; you also made an instance of class Ville and then break the encapsulation, you should declare it in the ville module; and to finish, the programming conventions has been ignored.
311105	3,00	[L1] noeud.cc 171,179,149, 195, 196, tools.h 9, 14	Attention to the indentations	As a whole, good project. Architecture and encapsulation is respected throughout. Coding style is good, except for several indentation mistakes.
311112	4,00	ok	essayer de garder le même style d'accolade dans tout votre code. excellente indentation le code est clair et agréable à lire, malgré quelques noms de variables qui pourrait être plus parlants	excellent code, très bonne architecture. continuez comme ça
311155	4,00		code clair et lisible, bon travail	bon travail dans l'ensemble, faites plus attention aux contraintes de la donnée

311236	3,00	[L1] : ville.h: 5, 7, ville.cc: 25, noeud.cc: 54, 96, 114	Trop de petites erreurs d'indentation vous coutent un point, c'est dommage ! Relisez bien les conventions ! Warning : vous utilisez deux styles pour les blocs d'une seule ligne : soit vous faites sans accolade sur la même ligne, soit vous mettez les accolades. Pensez à unifier ! Warning : la variable V dans noeud a un nom très discutable. Pensez à trouver plus parlant !	Bon projet, avec d'excellentes idées pour la manière de faire les calculs. Dommage que vous ne portiez pas plus d'attention à tout ce que peut apporter l'orienté-objet, car votre code serait bien plus agréable et facile d'utilisation pour vous si vous aviez une classe ville correcte, et que les interactions entre ville et noeud étaient un peu mieux utilisées. Par exemple : votre classe ville doit posséder un vecteur qui contient tous ses quartiers et réaliser la lecture, chaque fois qu'elle lit un nouveau Noeud, elle appelle le constructeur de Noeud pour obtenir un nouveau noeud, teste si ce noeud est un conflit avec l'un des noeuds précédemment ajouté dans la ville, et si tout va bien l'ajoute dans le vecteur. Je vous conseille de repasser sur ces parties du code, car bien encapsuler vous rendra la tâche bien plus facile pour les rendus à venir ! Courage !
311252	3,00	[L1] -city.cc l.24,26 - node.cc l.70,75,134,136 -tool.cc l.17	you have wierd line returns in city.cc. You have an inconsistant tab policy with the : part of function init in multiple areas (tool.cc l.17 or node.cc same line for exemple). You also need to take care of wraping line wierd stuff (node.cc l.70) and align second line arguments start with first character of the first line arguments (node.cc l.75). Warning[L1] be consistent with switch in one you have lines btw cases in another not.	Architecture de modules correcte. Faites attention à déclarer correctement le contenu de vos modules. Mise en forme inconsistente à laquelle il vous faudra porter attention. Faites aussi gaffe aux quelques }; qui ne sont pas nécessaires
311385	4,00	ok	warning : vous pouvez initialiser vos structs en 1 ligne comme : " Point p = {0,0} " au lieu de modifier chaque attribut un à un. Aussi arrêtez de créer des variables temporaires inutile comme : "double a; a= 0 ; return a;" vous pouvez directement mettre "return 0;"	Bon travail ! vous avez un très bon code mais faites attention au dernier Warning

311390	2,00	tout vos includes [L1] , et {} [P5.2] ville.cc l17 ,l15 , noeud.cc	mauvaise utilisation des #ifndef , pas besoin pour chaque inclusion ; etrange style d indentation , il est difficile de s y retrouver ex : lecture de ville.cc . Airez votre code !!! , ex create_link . Public/Private ne doivent pas etre indente. Donnez des nom plus indicatif que i,j,t1,t2,w	Tres bon programme et structure du code ! faites attention au style , et surtout a vos #ifndef
311574	3,00	[P5] city.cc 208, 300 test.cc 26-29, 36	warning : [L1] common_tools.cc 56	Vous devez complètement revoir l'architecture de votre code ! Vous avez beaucoup trop de modules essayer de revoir votre architecture à partir de celle conseillée dans la donnée. A part ça les conventions sont très bien respectées. Attention tout de même aux magic numbers.
311644	2,00	[L1] noeud.cc getters setters [P2] noeud.cc 105-202	deux indentations, fonction dépassant 80 lignes	Vos interfaces sont très claires et sobres ça montre que vous avez bien compris comment les utiliser! Externaliser un peu votre stockage de noeuds vous facilitera probablement la vie pour la suite.
311730	3,00	[P5] noeud.h : 26, 27, 35, 36, 37, 38	Vos noms de fonctions doivent être bien plus parlants ! Vous déclarez en apparence quatre fois la même fonction en changeant juste son numéro, il faut que l'on comprenne ce qu'elle fait rien qu'en lisant sa signature Warning : vous devez utilisez le même style de majuscule dans tout le fichier : soit camelCase, soit snake_case. La fonction addingnode ne rentre dans aucune des deux cases, à corriger	Très bon projet dans l'ensemble, félicitations ! Dommage que vous fassiez cette petite erreur, mais elle sera aisément corrigée. Pensez également à retirer les ifndef endif des .cc, car ces macros de protection ne sont utiles que pour les .h et éviter les inclusions infinies. Vous n'êtes pas loin d'un rendu parfait, et vous avez parfaitement compris le principe de l'orienté-objet, continuez !
311913	4,00	ok	warning : [L1] 248-250 ville.cc ; [P5] 57, 70, 83 ville.cc	excellente indentation, le code est clair et aéré. attention aux include dans projet.cc. certains retours à la ligne pour éviter le wrapping sont inutiles.
311916	4,00		Excellent code, parfaitement indenté	excellent dans l'ensemble, attention aux contraintes de la donnée, notamment l'architecture

311970	3,00	[L1] : projet.cc : 10, 11, ..., ville.cc: 67, 69, ville.h: 3 à 9, noeud.cc: 12 à 17, noeud.h: 11, 19, 36	Trop de petites erreurs d'indentation, c'est dommage ! Faites très attention au style de votre code, cela impacte tout de suite la visibilité. Vous vous perdez également dans les différentes notations pour les if sur une ligne : parfois avec les accolades, parfois sans, parfois l'instruction sur la ligne suivante, parfois sur la même ligne... Et ne mettez jamais le else sur la même ligne que le if !	Excellent code globalement, vous commettez l'erreur de ne pas vous servir de la classe ville, mais vous vous rendez compte que ce n'est pas viable et prévoyez de le corriger, j'ai donc mis un warning. Vous devrez avoir corrigé le souci pour le rendu 2 ! L'encapsulation est parfaitement respectée, à l'avenir vous pouvez essayer d'imaginer une solution qui se passe complètement de variables statiques, avec le vecteur de Noeud dans ville. Dommage que vous perdiez un malheureux petit point sur l'indentation, sinon votre rendu aurait été parfait ! Courage pour la suite !
311980	2,00	[L1] noeud.cc getters setters [P2] noeud.cc 105-202	deux indentations, fonction dépassant 80 lignes	Vos interfaces sont très claires et sobres ça montre que vous avez bien compris comment les utiliser! Externaliser un peu votre stockage de noeuds vous facilitera probablement la vie pour la suite.

312066	2,00	<p>[L1] ville.h; the class is indented entirely/ 59-132 ville.cc; in the switch, the case are not indented/ 57-58 tools.cc; long instructions in if not correctly aligned/ noeud.h; public and private keywords are indented and the methods/attributes of the class are indented 2 times/ 38-39,44-45 noeud.cc; the instructions in the functions are indented 2 times, etc.... [P2] 48-133 ville.cc: function of 85 lines</p>	<p>The indentation rules have been ignored, pay more attention to the conventions posted on moodle and be uniform with your coding style. Warning: [L2] 31 ville.h: more than 87 char.</p>	<p>Overall the architecture and the structure of the project is good, but some improvements need to be done on the coding style mainly. Few comments: the getters should be const; non-uniform calling for bool sometimes you call "true" 1 and sometimes true and same thing for "false"; tools.h is included 2 times in noeud.h; you made an ifstream that you don't use in the main; you also made an instance of class Ville and then break the encapsulation, you should declare it in the ville module; and to finish, the programming conventions has been ignored.</p>
312078	4,00		<p>Le style est très bon et vous utilisez correctement les constantes bravo</p>	<p>Bravo pour le rendu presque parfait. Attention juste a la fonction main qui doit seulement traiter les arguments du programme puis directement appeller une fonction du module ville (sans instancier de classe ville) + deplacer votre constructeur de noeud dans le fichier noeud.cc</p>

312106	4,00	ok	warning : vous pouvez initialiser vos structs en 1 ligne comme : " S2D origine = {0,0} " au lieu de modifier chaque attribut un a un	Great job ! the code is well structured , easy to read and clean.
312120	4,00	OK	Warning [L1] 22 node.h: long instructions in if not correctly aligned; Warning also if-else style is not always uniform.	The code is very nice structured (the namespace tools is a good idea)! There is only a few things to say: the if-else instruction could be more uniform; and "NodeManager" and "City" are the same conceptual thing then you don't need to separate this in 2 different class.
312121	2,00	[L1] noeud.cc 141, 149, ville.cc 174-237 [P5] constants.h 31,32,33,37,38	Inconsistent indentation of if-else, several instructions on same line.	As a whole, good project. It's a shame for the architecture violation in projet.cc, encapsulation is always respected. Coding style is pretty good, with the exception of a few indentation and wrapping errors. You should really pay attention to what is considered a magic number and what not to avoid defining meaningless and confusing constants)
312280	4,00	ok	warning : [L1] 248-250 ville.cc ; [P5] 57, 70, 83 ville.cc	excellente indentation, le code est clair et aéré. attention aux include dans projet.cc. certains retours à la ligne pour éviter le wrapping sont inutiles.
312282	3,00	[L1]: noeud.cc: 81, 140, ville.cc: 46, 85	Indentations oubliées et if else sur la même ligne (inconsistant avec d'autre if else sur une seule ligne dans le même code). Par ailleurs, vous avez fait le choix d'aligner vos wrapping lines avec ce qu'il y a au-dessus, et c'est un choix dommageable. Cela crée des cas comme dans noeud.cc : 102 où 4 lignes supplémentaires sont requises. En général, 2 indentations suffises	Bon projet dans l'ensemble mais vous passez à côté de ce qui fait la force de l'orienté objet. Toutes vos classes ne servent qu'à regrouper des données et y accéder (donc sont équivalentes à des struct), car toutes les fonctions qui devraient être des méthodes de ces classes sont définies en-dehors. Cela crée des cas absurdes, comme votre fonction createur_neoud qui est littéralement un constructeur de la classe Noeud, sauf qu'en plus elle doit avoir en argument le vecteur de Noeud entier ! Je vous conseille fortement avant de vous atteler au prochain rendu d'essayer de corriger ça, vous ne devriez avoir des fonctions en dehors de classe que dans tools et éventuellement la fonction de lecture de ville. Courage pour la suite !
312296	4,00	ok	warning : vous pouvez initialiser vos structs en 1 ligne comme : " S2D origine = {0,0} " au lieu de modifier chaque attribut un a un	Great job ! the code is well structured , easy to read and clean.
312367	4,00		Excellent code, parfaitement indenté	excellent dans l'ensemble, attention aux contraintes de la donnée, notamment l'architecture
312453	3,00	[L1] noeud.cc fonction switch	il faut aussi indenter les "case"	Ce serait une bonne pratique de définir les 0 et 1 que vous utilisez dans vos array de position comme X et Y afin de gagner en clareté. Votre code est très soigné et bien pensé dans l'ensemble!
312460	4,00	ok	-	Bon rendu, attention à l'encapsulation dans noeud sinon c'est bien

312577	3,00	[P5] ville.cc: 50 à 55, noeud.cc: 116 à 119, 127	<p>Vous utilisez trop de magic number ! Pourquoi return 1 plutôt que true pour un booléen, et vous utilisez sans cesse des tableaux de deux ou trois éléments sans que l'on comprenne pourquoi. Remplacez ces magic numbers par des symboles ou des constexpr (par exemple si j'ai un tableau coords de deux Points qui représente des coordonnées x et y, je fais les define pour pouvoir écrire coords[X] et coords[Y] plutôt que coords[0] et coords[1]) Warning : vous êtes inconsistant dans les styles d'indentation. Les if sur une ligne sont traités de deux manières différentes (noeud.cc: 116, tools.cc: 27), les for également (ville.cc: 57 et 58), et vous oubliez d'indenter les listes d'initialisations (noeud.cc: 23 à 36) A corriger pour le rendu 2 !</p>	<p>Plutôt bon projet dans le style et agréable à lire, malheureusement vous perdez pas mal de points car vous ne respectez pas les principes d'encapsulation. Cela risque de vous complexifier la tâche pour les rendus suivants : Ville doit être la seule à agir sur les vecteurs qui représente ses quartiers ! Elle lit le fichier, récupère les infos pour appeler un constructeur de Noeud, teste si ce noeud est en erreur avec l'un de ceux précédemment ajouté, puis l'ajoute si tout va bien. Pourquoi vouloir déléguer ces traitements à Noeud ? Si vous arrivez à corriger tout ça, vous serez en première ligne pour obtenir une note parfaite aux deux prochains rendus ! Courage !</p>
312620	3,00	[L1] tout noeud.cc	<p>vous ne respectez pas les indentations , le contenu entre {} d une fonction doit etre indente warning : vous pouvez initialiser vos structs en 1 ligne comme : " S2D origine = {0,0} " au lieu de modifier chaque attribut un a un</p>	<p>Bon travail , faites attention a votre indentation la prochaine fois</p>
312648	3,00	[L2] tools.h : 18, 93, 123, 124, 127, 173, 184, 201. tools.cc : 43	<p>Watch out for your line alignment when avoiding wrapping lines, it is very inconsistent, please check the conventions on this.</p>	<p>Very clean code, congratulations! Watch out for the mentioned problems in the style and architecture but generally the code is good. Maybe consider adding inheritance for your nodes in the future. Good job!</p>
312656	4,00	ok	-	<p>Bon rendu, attention au namespace std qui peut jouer des tours, belle mise en forme</p>

312737	4,00	-	<p>Vous respectez les conventions de programmation mais pensez à un peu plus espacer votre code afin de rendre la lecture plus fluide. Par exemple, la fonction checkNodeLinkSuperposition pourrait avoir plus d'espace pour être plus lisible. De plus utilisez true/false au lieu de 1/0 (tools.cc:23-24)</p>	<p>Traitez de manière unique le cas <code>if(!(line[0] == '#'))</code> et améliorez votre présentation du code, sinon continuez sur cette lancée !</p>
312757	2,00	<p>[L1]noeud.h:9,12,20,27. ville.h:8,11 [L1]noeud.cc:43-52 [P5]noeud.cc:101,104,119,134</p>	<p>- les mots clé private/public ne doivent pas être indentés - Attention noeud.cc:43-52 : le corp de la fonction est indenté deux fois - les états sont défini en tant qu'entiers hardcodé et devraient être définis en tant que constantes afin qu'on puisse comprendre le code</p> <p>Warning Style :la fonction "collisionPassage" est difficile à lire(noeud.cc:58-67), le calcul des projections peut se faire entierement dans le module tools tant que vous ne faites pas allusion aux modules noeud/ville.</p>	<p>Bon rendu, cependant il faut revoir un peu le style ainsi que votre encapsulation</p>
312854	4,00	Warnings	<p>Warning: [L1] ville.cc 20,112 ; ville.h 27</p>	<p>As a whole, good project. Architecture and Encapsulation are respected throughout. Coding style is good, with only occasional indentaion mistakes. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.</p>
312920	4,00	ok	<p>warning : vous pouvez initialiser vos structs en 1 ligne comme : " Point p = {0,0} " au lieu de modifier chaque attribut un a un. Aussi arretez de creer des variables temporaires inutile comme : "double a; a= 0 ; return a;" vous pouvez directement mettre "return 0;"</p>	<p>Bon travail ! vous avez un tres bon code mais faites attention au dernier Warning</p>
312936	4,00		<p>Très bon style attention cependant : [P5]projet.cc:11 :n'utilisez pas de constantes du style (ONE 1) cela est inutile.</p>	<p>Bravo pour votre rendu parfait, l'architecture est bien respectée, vous utilisez les classes correctement en respectant l'encapsulation. Attention tout de meme au petit détail sur l'utilisation des constantes.</p>

312983	3,00	[P5] noeud.cc 37, 54-56	excellent style attention [L2] : plusieurs instructions sur la même ligne noeud.cc 54-56.	très bon code malgré des petites fautes de style. poursuivez dans cette voie.
313001	4,00	ok	Warning [P5] char values names not explicit (ville.cc l.71-72 & noeud.cc l.23)	Bon rendu, manque parfois un peu de clarté ((les commentaires c'est bien) mais agréable à lire, belle rigueur de mise en forme. Des namespaces pour certaines fonction serait appréciés pour un peu de clarté supplémentaire.
313218	4,00		Warning : noeud.cc: 25, 26, ville.cc: 43, vous utilisez des magic numbers. Vous devriez définir un symboles ou une constexpr pour éviter cela, afin d'écrire content[POSITION_X], par exemple, plutôt que content[2] qui n'a pas de sens pour le lecteur	Très bon projet dans l'ensemble, code très agréable à lire et vous avez le bon goût de ne commettre que des demi-impairs qui ne vous coutent (pour le moment) pas de point. Notamment pour la rupture d'encapsulation, j'ai épluché votre code, et à aucun endroit vous n'agissez directement sur le vecteur ensemble depuis noeud, vous ne faites que le lire pour détecter les erreurs. Vous n'auriez pas dû vous acharner à faire tous les tests dans Noeud : certains doivent être faits dans Ville, et cela vous évitera d'avoir à donner accès à tous les noeuds de la ville à une classe qui ne le doit pas. Corrigez toutes ses remarques, et votre projet sera parfait ! Faites par ailleurs attention au français, choisissez votre camp définitivement, et n'utilisez pas le capslock pour un type (MODE), contentez vous des #define. Encore bravo pour l'ensemble !
313312	4,00	OK	Warning: [P5] 8,9 tools.h; 8 ville.h: poor choice of symbol name, be careful the names could be ambiguous.	The code is commented in the .h files and it's very appreciable ! It's also very well spaced out and pleasant to read. There is just few comments to say: you need to externalize the definition of the constructor if it does not hold on one line; be careful with the names you give to symbol; and for if-else style when you have simple instruction and write it on the same line, you don't need to put 1 tab or even more between the end of the "if" parenthese and the instruction.
313318	3,00	[P5] noeud.cc 37, 54-56	excellent style attention [L2] : plusieurs instructions sur la même ligne noeud.cc 54-56.	très bon code malgré des petites fautes de style. poursuivez dans cette voie.
313415	3,00	[L1] noeud.cc 23 ville.cc 125, 126, 144	Il est interdit de mettre plusieurs instructions sur la même ligne. Le seul cas accepté étant une instruction de contrôle suivie par l'instruction qui en découle. On évite donc les if/else sur la même ligne et plusieurs affectations sur la même ligne.	Très bon code agréable à lire et clair. Evitez de trop compacter les instructions sur la suite de votre code!
313705	4,00	-	Très bonne présentation. Excellent !	Il n'était pas nécessaire de recréer LINK_VACUUM en define: déjà présent dans constantes.h en constexpr. A part la classe Ville qui semble être inutile, c'est très bien ! bravo !

314353	3,00	[L2] tools.h : 18, 93, 123, 124, 127, 173, 184, 201. tools.cc : 43	Watch out for your line alignment when avoiding wrapping lines, it is very inconsistent, please check the conventions on this.	Very clean code, congratulations! Watch out for the mentioned problems in the style and architecture but generally the code is good. Maybe consider adding inheritance for your nodes in the future. Good job!
314377	3,00	[L1] noeud.cc fonction switch	il faut aussi indenter les "case"	Ce serait une bonne pratique de définir les 0 et 1 que vous utilisez dans vos array de position comme X et Y afin de gagner en clareté. Votre code est très soigné et bien pensé dans l'ensemble!
314443	3,00	[P5]ville.cc:41,53,57,61,65,70,75,83	<p>Les états de l'automate de lecture sont des entiers de 1 à 7 hardcodé. Il faut créer des constantes pour les définir (les constantes peuvent bien évidemment être des entiers mais le nom permet d'informer au lecteur l'état concerné), Cela rend le code de l'automate très difficile à comprendre en tant que lecteur. + quand vous testez l'erreur "reserved_uid" utilisez des constantes au lieu d'hardcoder 4294967294 (d'ailleurs déjà définie dans constantes.h "no_link") et 770. Mise à part cela le style du code est bon et très lisible alors Bravo !</p>	Bon rendu, malheureusement vous perdez 2 points à cause de petits détails. Faites attention avec le stockage de la liste des noeuds dans un espace non-nommé, je pense que ça risque de vous amener des complications pour les prochaines étapes

314456	4,00	OK	Warning: [P5] 8,9 tools.h; 8 ville.h: poor choice of symbol name, be careful the names could be ambiguous.	The code is commented in the .h files and it's very appreciable ! It's also very well spaced out and pleasant to read. There is just few comments to say: you need to externalize the definition of the constructor if it does not hold on one line; be careful with the names you give to symbol; and for if-else style when you have simple instruction and write it on the same line, you don't need to put 1 tab or even more between the end of the "if" parenthese and the instruction.
314470	3,00	[L1] ville.cc 45-107	Inconsistent indentation of if-else, several instructions on same line. Warning: wrapping parameters of a function should be aligned to the first letter of the first parameter, not to the parentheses. Warning: you often use many useless intermediate variables, for instance just to store a sum that you are going to use once.	The code has room for improvement. Avoid using too many intermaediate variables that can potentially lead to less clear code (not too many, not too little, there is a little gray area in between and that's what you should aim for). Pay attention to your indentations. It is prohibited to define functions in the .h . Architecture respected with the exception of error.h included in projet.cc). I did not understand why you have the fichier variable in the main. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.
314472	3,00	[L1] noeud.cc l42, erreur() [P5.2] tools.cc l11,12,13	les variables : int un(1) , int deux(2) ... sont inutile. Attention a l indentation a vos switch . Warning : remplacez vos " Int a; a = 2; return a"; par "return 2".	Bon travail dans l ensemble , malgres les petites fautes d implemetations.
314477	3,00	[L1] Dans toutes vos classes, ville.cc 36-91	Les mots clefs "public " et "private" ne doivent pas être indentés, on ne met pas deux instructions if/else sur la même ligne. Attention à indenter tous vos cases et les instructions de la même manière. Pensez à utiliser les mots-clés true et false à la place de 1 et 0 pour gagner en clareté.	Code bon dans l'ensemble. Attention, ajoutez un commentaire dans votre module Connexion pour en définir l'utilité et la raison pour laquelle vous avez choisi de l'ajouter, d'un point de vue externe il semble complexifier votre code sans justification.

314483	4,00	noeud.cc wrapping line indentation : 104	Excellent style, good job.	Excellent job with your code, I see a lot of attention to detail. I see you use char for typing, consider implementing inheritance for nodes, it will surely be useful for the next stages of the project. Good job!
314495	4,00	ok	-	Bon rendu, attention au namespace std qui peut jouer des tours, belle mise en forme
314504	4,00		Votre style est bon. Warning on remarque une légère discontinuité dans vos placements d'accolades, essayez de rendre le style uniforme au maximum.	Il serait bon de spécifier dans vos prototypes des noms de paramètres. Un lecteur de vos interfaces aurait de la peine à déterminer quel type correspond à quoi. Mis à part cela votre code est assez bien pensé et préparé pour la suite.
314754	4,00		Le style est très bon et vous utilisez correctement les constantes bravo	Bravo pour le rendu presque parfait. Attention juste a la fonction main qui doit seulement traiter les arguments du programme puis directement appeller une fonction du module ville (sans instancier de classe ville) + deplacer votre constructeur de noeud dans le fichier noeud.cc
314886	4,00		conventions respectées et code agréable à lire, continuez comme ça!	attention pour la suite à ne pas utiliser les constantes (dist_min) dans le module tools car c'est pénalisé, sinon très bon rendu!

314939	4,00		<p>Bon respect de l'indentation hormis quelques détails:</p> <ul style="list-style-type: none"> - noeud.h les mots clé public des sous classes ne devraient pas être indentés - tools.cc:43 la ligne fait 89 caractères attention - ville.cc:26 la ligne est trop longue et l'usage des accolades inconsistents, dans ce cas la vous pouvez juste passer à la ligne - Attention à l'automate qui est à la limite par rapport aux nombres de lignes, p-e définissez des fonctions pour chaque état 	<p>Dans l'ensemble très bon rendu et le code est agréable à lire Bravo ! Faites attention tout de mêmes aux petits détails d'indentation et longueurs de lignes</p>
314962	4,00	-	<p>Vous ne faites aucune faute pour la présentation ! Pensez néanmoins à "aérer" votre code: les getters n'ont pas besoin d'être sur une seule ligne, vous pouvez faire un retour à la ligne après un if ou un else, même pour une instruction, etc.</p>	<p>Améliorez un peu la présentation du code ! A part ça, bravo !</p>
315014	4,00	OK	<p>Warning [L1] 22 noeud.h: long instructions in if not correctly aligned; Warning also if-else style is not always uniform.</p>	<p>The code is very nice structured (the namespace tools is a good idea)! There is only a few things to say: the if-else instruction could be more uniform; and "NodeManager" and "City" are the same conceptual thing then you don't need to separate this in 2 different class.</p>
315019	4,00	noeud.cc wrapping line indentation : 104	<p>Excellent style, good job.</p>	<p>Excellent job with your code, I see a lot of attention to detail. I see you use char for typing, consider implementing inheritance for nodes, it will surely be useful for the next stages of the project. Good job!</p>
315065	4,00	Warnings	<p>Warning: [L1] tools.cc 25: variable names not aligned across different lines. I would suggest doing several separate declarations. Warning: [L1] noeud.cc 42,51: function parameters not aligned across several lines.L1</p>	<p>As a whole, good project. Indentation is sometimes wrong, and I personally would prefer a more spaced out code, but as to now it is still ok. Architecture and encapsulation is respected throughout.</p>

315120	4,00	OK	Warning: [L2] 157 noeud.cc: line contains more than 87 char.	Great delegation of tasks to sub_functions ! The architecture is ok and the coding style is nice and uniform. Few comments: "lecture_fichier" module is very dependent on "ville" module and then it is not necessary to add this module; also in this module there is a struct and then be very careful because it means that all the "attributes" are here in "public".
315192	4,00	ok	warning : vous pouvez initialiser vos structs en 1 ligne comme : " S2D origine = {0,0} " au lieu de modifier chaque attribut un a un	Tres bon travail ! le code est agrable a lire , prenez en compte le dernier warning pour ameliorer votre code.
315273	4,00	ok	warning : [P5] 19, 134 ville.cc	très bonne architecture, le code est parfaitement indenté, continuez comme ça.
315334	3,00	[L1] ville.cc 45-107	Inconsistent indentation of if-else, several instructions on same line. Warning: wrapping parameters of a function should be aligned to the first letter of the first parameter, not to the parentheses. Warning: you often use many useless intermediate variables, for instance just to store a sum that you are going to use once.	The code has room for improvement. Avoid using too many intermediate variables that can potentially lead to less clear code (not too many, not too little, there is a little gray area in between and that's what you should aim for). Pay attention to your indentations. It is prohibited to define functions in the .h . Architecture respected with the exception of error.h included in projet.cc). I did not understand why you have the fichier variable in the main. Just a warning for future reference: "using namespace" should not be used in .h as it will affect all .cpp including said .h and this can cause huge compilation issues.
315408	3,00	[L1] tout noeud.cc	vous ne respectez pas les indentations , le contenu entre {} d une fonction doit etre indente warning : vous pouvez initialiser vos structs en 1 ligne comme : " S2D origine = {0,0} " au lieu de modifier chaque attribut un a un	Bon travail , faites attention a votre indentation la prochaine fois

315430	3,00	[L1]tools.cc:159;167, [L1]ville.cc:208-210,230-232	A part quelques oublis, le code très bien présenté. Grande uniformité dans la présentation et facile à lire ! Bravo !	Vous avez fait un excellent rendu. Le code est bien présenté et semble (sans exécution) optimal. Vous utilisez try/catch correctement. Bravo à vous !
315536	4,00	Ok	Warning [L1] some spaces are not consistent	Très belle mise en forme, bien segmentée. Attention au using namespace std généralisé.
315548	3,00	[P2] ville.cc : automate makes more than 100 lines	Thank you for spacing-out your code, but sometimes it can be too mcuh like in your automate function, making it really too long. Code is very well written, with comments and attention to detail. Very good job	Very good project, architecture is fine and the approach to the problem is good. Well done making the default constructor of node not usable, this shows attention to detail. Very good job. Consider adding inheritance to your project for nodes.

315567	3,00	[L1] noeud.cc 126-129, 209-211, 220, tools.h 12-28, 39-44, 53-66	attention aux erreurs d'indentation	bon code avec une bonne architecture. vos méthodes de classes sont pertinentes. cependant, quelques noms de variables pourrait être plus indicatifs.
315587	3,00	[L1] noeud.cc 142, erreur() [P5.2] tools.cc 111,12,13	les variables : int un(1) , int deux(2) ... sont inutile. Attention a l'intentation a vos switch . Warning : remplacez vos " Int a; a = 2; return a"; par "return 2".	Bon travail dans l'ensemble , malgres les petites fautes d'implemetations.
315651	4,00	Ok	Ok	Excellent travail, ça pète ! De l'héritage aux petits oignons, des classes pour tout, une parfaite maîtrise de l'orienté-objet et du C++, j'ai pas grand chose à dire. Normalement vous devriez avoir besoin d'une autorisation pour démultiplier le nombre de fichiers, toutefois vous avez fait ça habilement et tous les fichiers du module tools respectent leur principe "bas niveau"... Pour être vraiment nickel, il faudrait éviter toute fonction hors-classe : vous pouvez le faire ! Votre fonction removeNode ne sert qu'à ce que la Ville puisse supprimer un noeud, ce qu'elle peut aisément faire en le retirant de son vecteur, le destructeur de Noeud peut se charger de prévenir les voisins pour une déletion parfaite des liens, et les fonctions de tools peuvent devenir des méthodes des classes correspondantes. A vous de voir ! Encore bravo en tout cas, je suis impressionné
315788	3,00	[L1] 41,58,75,122,123 noeud.cc: more than an instruction on one line/ 154,164,184-194 noeud.cc; 23 noeud.h, ... etc... :indentations are not correct./ noeud.h,ville.h: public and private keywords are indented and the methods/attributes of the class are indented 2 times	Warning: too many static variables declared in a function (8) noeud.cc line 33-35.	Architecture is good and the code is well spaced out except "decodage_ligne" function: don't try to save lines in the reading function by putting multiple instructions in the same line, you'd better delegate some tasks to sub-functions. Few other comments: there is a switch case (in noeud.cc line 112-125) that have some case with nothing inside, some case without break and some case with break but on the same line and with different indetations each time; the coding style is not always uniform; you need to externalize the definition of methods of the class; and to finish pay attention to indentation rules and conventions posted on moodle.
315824	4,00	ok	warning : [P5] 19, 134 ville.cc	très bonne architecture, le code est parfaitement indenté, continuez comme ça.
315856	4,00		code clair et lisible, bon travail	bon travail dans l'ensemble, faites plus attention aux contraintes de la donnée

315930	4,00	ok	warning : [P5] attention aux magic numbers (36 projet.cc, 93, 95 ville.cc) ; [L1] essayez de rester constants dans votre style d'indentation	bon programme, les noms des variables et constexpr sont pertinents.
315948	4,00		Très bon style attention cependant : [P5]projet.cc:11 :n'utilisez pas de constantes du style (ONE 1) cela est inutile.	Bravo pour votre rendu parfait, l'architecture est bien respectée, vous utilisez les classes correctement en respectant l'encapsulation. Attention tout de meme au petit détail sur l'utilisation des constantes.
315974	3,00	[L1] -city.cc l.24,26 - node.cc l.70,75,134,136 -tool.cc l.17	you have wierd line returns in city.cc. You have an inconsistant tab policy with the : part of function init in multiple areas (tool.cc l.17 or node.cc same line for exemple). You also need to take care of wraping line wierd stuff (node.cc l.70) and align second line arguments start with first character of the first line arguments (node.cc l.75). Warning[L1] be consistent with switch in one you have lines btw cases in another not.	Architecture de modules correcte. Faites attention à déclarer correctement le contenu de vos modules. Mise en forme inconsistente à laquelle il vous faudra porter attention. Faites aussi gaffe aux quelques }; qui ne sont pas nécessaires
315983	3,00	[L1] 41,58,75,122,123 noeud.cc: more than an instruction on one line/ 154,164,184-194 noeud.cc; 23 noeud.h, ... etc... :indentations are not correct./ noeud.h,ville.h: public and private keywords are indented and the methods/attributes of the class are indented 2 times	Warning: too many static variables declared in a function (8) noeud.cc line 33-35.	Architecture is good and the code is well spaced out except "decodage_ligne" function: don't try to save lines in the reading function by putting multiple instructions in the same line, you'd better delegate some tasks to sub-functions. Few other comments: there is a switch case (in noeud.cc line 112-125) that have some case with nothing inside, some case without break and some case with break but on the same line and with different indetations each time; the coding style is not always uniform; you need to externalize the definition of methods of the class; and to finish pay attention to indentation rules and conventions posted on moodle.

316049	2,00	[L1]noeud.h:9,12,20,27,ville.h:8,11 [L1]noeud.cc:43-52 [P5]noeud.cc:101,104,119,134	<ul style="list-style-type: none"> - les mots clé private/public ne doivent pas être indentés - Attention noeud.cc:43-52 : le corp de la fonction est indenté deux fois - les états sont défini en tant qu'entiers hardcodé et devraient être définis en tant que constantes afin qu'on puisse comprendre le code <p>Warning Style :la fonction "collisionPassage" est difficile à lire(noeud.cc:58-67), le calcul des projections peut se faire entierement dans le module tools tant que vous ne faites pas allusion aux modules noeud/ville.</p>	Bon rendu, cependant il faut revoir un peu le style ainsi que votre encapsulation
316117	2,00	tout vos includes [L1] , et {} [P5.2] ville.cc l17 ,l15 , noeud.cc	mauvaise utilisation des #ifndef , pas besoin pour chaque inclusion ; etrange style d indentation , il est difficile de s y retrouver ex : lecture de ville.cc . Airez votre code !!! , ex create_link . Public/Private ne doivent pas etre indente. Donnez des nom plus indicatif que i,j,t1,t2,w	Tres bon programme et structure du code ! faites attention au style , et surtout a vos #ifndef
316132	3,00	[P5] city.cc 208, 300 test.cc 26-29, 36	warning : [L1] common_tools.cc 56	Vous devez complètement revoir l'architecture de votre code ! Vous avez beaucoup trop de modules essayer de revoir votre architecture à partir de celle conseillée dans la donnée. A part ça les conventions sont très bien respectées. Attention tout de même aux magic numbers.
316258	3,00	[P5]jofile.cc:41,52,math.cc:96,math.h:30,ville.cc:27	Très bonne présentation mais contrôlez la taille maximale de vos lignes.	Vous créez plus de fichiers que demandés et vous vous compliquez un peu la vie. Vous avez en revanche un makefile minimalise très beau et le projet dans son ensemble est bon !

316306	4,00	ok	-	Bon projet. Mise en forme claire, juste la structure s2d qui n'a pas un nom explicite.
316312	3,00	[L2]noeud.cc:436,446,456,474,475,483,...	[L2]noeud.cc:436,446,456,474,475,483,... : Les lignes sont trop longues (max 87 caractères sinon passez à la ligne)	- Le rendu est bon. Bravo pour la bonne encapsulation des classes. Cependant, vous utilisez trois classes pour représenter les noeuds sans utiliser l'héritage, cela amène beaucoup de répétitions dans votre code. Je vous conseille d'utiliser l'héritage et de revoir votre organisation de classes.
316333	4,00	ok	Warning [P5] char values names not explicit (ville.cc l.71-72 & noeud.cc l.23)	Bon rendu, manque parfois un peu de clarté ((les commentaires c'est bien) mais agréable à lire, belle rigueur de mise en forme. Des namespaces pour certaines fonction seraient appréciés pour un peu de clarté supplémentaire.
316368	4,00	ok	public/private ne doivent pas être indentés . les méthodes qui ne changent pas les attributs de la classe doivent être const. warning : vous pouvez initialiser vos structs en 1 ligne comme : " Point p = {0,0} " au lieu de modifier chaque attribut un à un	Bon travail ! vous avez un très bon code mais faites attention au dernier Warning
316437	4,00	Ok	Warning [L1] some spaces are not consistent	Très belle mise en forme, bien segmentée. Attention au using namespace std généralisé.
316518	4,00	ok	Warning : comments should not be avoided	Great job ! the code is well structured , easy to read and clean.

316535	4,00		<p>Warning : noeud.cc: 25, 26, ville.cc: 43, vous utilisez des magic numbers. Vous devriez définir un symboles ou une constexpr pour éviter cela, afin d'écrire content[POSITION_X], par exemple, plutôt que content[2] qui n'a pas de sens pour le lecteur</p>	<p>Très bon projet dans l'ensemble, code très agréable à lire et vous avez le bon goût de ne commettre que des demi-impairs qui ne vous coutent (pour le moment) pas de point. Notamment pour la rupture d'encapsulation, j'ai épluché votre code, et à aucun endroit vous n'agissez directement sur le vecteur ensemble depuis noeud, vous ne faites que le lire pour détecter les erreurs. Vous n'auriez pas dû vous acharner à faire tous les tests dans Noeud : certains doivent être faits dans Ville, et cela vous évitera d'avoir à donner accès à tous les noeuds de la ville à une classe qui ne le doit pas. Corrigez toutes ses remarques, et votre projet sera parfait ! Faites par ailleurs attention au français, choisissez votre camp définitivement, et n'utilisez pas le capslock pour un type (MODE), contentez vous des #define. Encore bravo pour l'ensemble !</p>
316536	3,00	[L1]noeud.cc:12,26,55,58,64,74,79...	<p>-Incohérence de style d'accolades à multiples endroits : a travers le code vous mélangez deux styles d'utilisations d'accolades soyez plus cohérent pour le prochain rendu</p> <p>- Warning : n'indentés pas les mots clé private/public</p>	<p>- Très bon rendu bravo, faites attentions à la cohérence du style d'utilisation des accolades.</p>
316830	3,00	[P5]ville.cc:77,78,81,82,85,86,89,...	<p>Utilisez des enums pour éviter les magic numbers ! Vous pouvez aussi écrire vos getters sur d'avantages de lignes. Les autres règles des conventions de programmation sont respectées !</p>	<p>C'est un bon projet mais pensez à corriger les divers points énumérés aux sections précédentes.</p>