

Sciper	style and conventions (max 2pts)	violation list	Comments on style and conventions
271829	1,5	[P5]50,93,163...	Code propre et lisible. Attention à bien définir tes Magic Numbers en début de programme ( et non dans les fonctions) avec des #define ou constexpr !
273712	2,0		Ton style est très bien, code est parfaitement indenté. Bravo!
274888			
275185	2,0		<p>warning [L2] 202 Replacing it with a line break would not be consistent with your indentation style</p> <p>warning [P5] 34</p> <p>In <i>filtrage</i>, the incrementation of <i>compteur</i> could have been made shorter and more legible by computing its value as :</p> <pre>compteur = (pix[...] == l) + (pix[...] == l) + (pix[...] == l) + ...</pre>
275336	1,5	[P5] 38,56,81...	Nice code, the indentation is great. Be careful : The magic numbers must be defined out of the functions (with #define or constexpr).
276861	2,0		<p>warning [P5] 27, 146 Note that exit(0) is equivalent to exit(EXIT_SUCCESS)</p> <p>The code seems a bit dense to me, but it is still very well presented. Good job!</p>
279202	1,5	[P5] 15, 16, 31, ...	Clean code and nice structure ! The conventions are well-respected. Pay attention to the magic numbers.
282211	1,5	[P5]60,66,78,79,...	Il y a des problèmes avec des magic numbers mais ton code est très respectueux des conventions de programmation. Bravo !
282515	1,5	[P5]127, 176, 255...	Très bon style, bravo aussi pour les commentaires. Il manque encore une dernière relecture, il y a quelques erreurs d'inattention.
282708	1,0	[L22] 65 [L1] 185	Ton code est assez facile à lire, malgré 1-2 erreurs d'indentation qui s'y sont glissées.
282712	2,0		Code très lisible. Bon travail de définition de symboles (sauf ZERO qui n'apporte aucune information). Plusieurs nouveaux noms n'apportent aucune information, par ex: Tableau_de_int. De plus ce dernier nom de type est pour un vector de vector ce qui peut induire en erreur. parfois la lecture d'un élément de tableau prend 3 lignes au lieu d'une seule : 128-130, 161-163.
282756	0,5	[L1] 124, 216, 261, [L22] 56, 59, 63, 116, [P5] 17, 18, 21, 22, 23, 24, 25, 26, 27, 28, 29	Le style est généralement bon, mais c'est dommage pour l'indentation. Les paramètres en constantes globales n'ont pas été très bien compris. Il ne faut pas en mettre trop, mais espacer un peu les lignes aide beaucoup pour la lisibilité du code: il est ici un peu trop compact et donc dur à lire. Mettre un peu plus de commentaires peut aussi aider pour la lisibilité.

282844	1,0	[L1]60,90,147 [P5]20,21	Attention à bien garder une indentation uniforme Pense aussi à mettre des commentaires pour expliquer ton code L'intérêt des constexpr/define/enums est d'expliquer ce que représentent ces valeurs. Ici certains noms que tu donnes sont juste le nombre en toutes lettres, ce qui du coup ne résoud pas le pb des "magic numbers"
282988	1,5	[P5]82, 85, 147, 295	Assez propre et lisible, bon choix de noms de variables. Conseils: éviter les nested loops trop profonds et repenser l'utilisation des breaks, il y a souvent une approche plus propre. Attention aussi aux magic numbers.
283389	1,5	[L22] 54	Style soigné, très bonne implémentation des fonctions. Très bonne utilisation des commentaires!
283391	1,5	[L14] 248, 295, 301	Style excellent, très soigné, les implémentations des fonctions sont très pertinentes. Mettre au moins quelques commentaires peut aider à comprendre le code. C'est très bien d'utiliser des fonctions avec type de retour non void, mais passer les arguments par référence plutôt que par valeur peut dans certains cas être bénéfique pour la vitesse d'exécution.
283419	1,5	[L22]48,51,54,...106,115	code globalement clair, bon usage des symboles pour les constantes. Attention: faire un effort pour l'alignement des instructions sur plusieurs lignes, warning: un simple return xxx sur une seule ligne suffit (au lieu de 4 dans votre code) après un bloc "if" qui se termine par return (ex: 149-152).
283470	1,0	[L1] 332 [P5] 152, 206, 218, ...	Bon respect des conventions, dommage pour les magic numbers. Bon effort de présentation avec l'utilisation des commentaires.
283792	2,0		Très bonne documentation et structure du code ! Continue ainsi. Attention aux noms de constantes : elles doivent être en majuscule.
284194	2,0		Excellente présentation du code bravo ! Très bonne utilisation des commentaires pour améliorer la lisibilité et apporter des explications.
284284	1,5	[P5]75, 200, 261	Très bon travail dans l'ensemble ! Un bravo particulier pour l'indentation qui est excellente, notamment au niveau de ton travail sur les voisins d'une case. Seul petit souci : laisse passer moins de nombres magiques
284564	1,5	[P5]113, 300	Du très bon travail ! Rien à redire sur l'indentation. Pour les constantes, c'est vraiment dommage, tu avais fait le define mais tu as oublié de l'utiliser à plusieurs endroits !
284867	0,5	[L14] 95, 96, 127, 295 .. [L22] 222, 235, 36-43 .. [L1] 251, 271	Plusieurs erreurs de conventions : fais bien attention à les corriger pour tes autres projets dans l'avenir, car c'est facile à régler et ça peut te faire perdre beaucoup de points. Notamment, ton indentation est incohérente (surtout l'alignement des paramètres des fonctions).
284896	1,0	[L22] 358 [P5] 100, 181, 209,307	Très bon code en général à l'exception de la petite faute d'attention à la ligne 358. D'autre part, tu n'initialises pas de valeurs globale constante et donc tu as pas mal de soucis avec les magic numbers. Ces valeurs sont des valeurs paramétrisables qui pourraient changer plus tard et donc ce serait utile de les avoir en tant que variables globales (define, constexpr, etc). Ton code est agréable à lire et bien indenté, bravo!
286317	2,0	[P5] 93, 126, 131, 180, 195, 230, 255, 279	Ce serait bien d'aérer un peu plus ton code, comme par exemple dans ton main (ajouter des commentaires ce serait encore mieux). Je vois que tu as fait de l'effort pour les magic numbers mais hélas tu en as encore quelques uns. Sinon ton indentation est très bien et ton code aussi, bravo!
286386	1,5	[L1]178,180,182...	Excellent respect des conventions sauf la petite étourderie dans la fonction filtrage. Continue ainsi
286509	0,5	[P5] 27, 29, 36, 39, 42, 50, 51, 60,96, 112-115, 149, 158, 184 [L14] 140 [L22] 106	Bonne indentation en général à part une petite erreur à la ligne 140. Cependant trop de magic number et aucune constantes définies. petite erreur en retour à la ligne d'une expression ce qui n'est pas très élégant

286570	0,0	[L1] 84, [L11] 217, 251, 309, [L14] 128, 149, 229, [P5] 16, 18	Bonne utilisation des structures, et bonne initiative avec les commentaires. Il y a par contre nombre d'améliorations à apporter à votre code, au niveau de l'implémentation. Les constantes globales doivent avoir des noms symboliques, remplacer 6 par SIX n'a aucune utilité, et nommer des valeurs qui n'ont pas vraiment de symbolisme (des coefficients dans une formule) est inutile et peut induire le lecteur en erreur. C'est inutile d'utiliser une boucle for pour 3 itérations seulement (quand le nombre 3 est connu à l'avance, ce qui est le cas ici). C'est mieux d'écrire directement if (test) plutôt que if (test == false).
286641	2,0		Excellent code : les conventions sont bien respectées, le code est très lisible et agréable à lire. Joli travail !
286721	1,5	[L2] 62, 346	Le style est soigné, des choix intelligents de conception ont été faits, surtout au niveau des typedef. Dommage pour le wrapping des arguments de sortie_image, cela a du être de l'inattention.
286734	1,5	[P5]11-17	L'intérêt des constexpr/define/enums est d'expliquer ce que représentent ces valeurs. Ici les noms que tu donnes sont juste le nombre en toutes lettres, ce qui du coup ne résoud pas le pb des "magic numbers"
286955	1,5	[P5]70,75,110,168	Tu as très bien suivi les conventions de programmation mais il reste quelques magic numbers.
286962	0,5	[L14] 266 [L22] 25, 26, 28, 42, [P5] 22, 66, 126, 273	Cela me fait de la peine de t'enlever autant de points car je vois quand même que tu as mis de l'effort. C'est dommage les petites fautes comme les alignements dans les prototypes et la petite faute d'indentation. Cela dit je vois que tu as utilisé des define mais que tu n'as peut être pas compris leur utilité, vu que tu defines une constante TROIS = 3. Alors on defini des constantes globales dans le code car ces valeurs peuvent être paramétrisables, cela veut dire qu'elles peuvent changer plus tard, par exemple on veut utiliser un autre format ou d'autres couleurs que RGB 8 RGBA ou grayscale) et donc c'est bien d'avoir un moyen de changer ces valeurs à un seul endroit et que ce changement soit repercuté partout dans le code. Parexemple ton TROIS = 3, si plus tard nous decidons d'utiliser RGBA alors ce 3 devient 4 et t'auras donc une variable TROIS = 4 dans ton code, qui est incohérent. Sinon ton code est bien indenté en général et il est agréable à lire. Désolé pour les points perdus.
287054	1,0	[L22] 329 [P5] 409, 417	Global constants are either all in CAPS or all in small letters. You have MAX and epsilon, so try to be consistent. Your code is very well indented and looks very good. You still have some small issues with magic numbers such as the size of the neighbouring pixels ( this is a parameter that could easily be changed) and a minor issue of attention in a long line. Nice job!
287056	1,0	[L22] 306 [P5] 211, 232...	The code is well indented and nice to read. Try to define more constants than just MAX to remove every Magic Number of your code.
287428			
287433	1,5	[P5]72, 186	L'indentation est très bonne ! Attention toutefois aux magic numbers
287443	1,0	[L1] 115, 138, [P5] 119	Implémentation claire et concise, très bien expliquée par les commentaires. Rien à rajouter, mais dommage d'avoir oublié d'utiliser une constante pourtant déjà définie (3 au lieu de NBC_PIXEL à la ligne 119).
287459	2,0		Good and extensive use of constexpr but be careful to keep it relevant. <i>two_char</i> and <i>three_char</i> seem quite close to meaninglessness.  And I think I'll just let the emptiness of this cell account for all there is to say.
287480	0,5	[L2] 63, 64, 65, 23, 34, ... [L14] 104 [L22] 23, 30, 32, ...	Dommage pour les soucis d'indentation, sinon le code en général est relativement bien aéré.
287529			

287609	1,0	[L2] 23, 30, 34... [P5] 134, 171...	Bonne indentation et très bons alignements mais attention à bien définir autant de constantes qu'il en faut pour paramétrer le problème. En ce qui concerne [L2], c'est de ta responsabilité de vérifier si l'impression est correcte.
287725	1,5	[P5]173,187,190	Bonne mise en page Pense par contre à mettre des commentaires pour expliquer ton code De plus il aurait fallu des constantes pour le seuil du delta et les voisins
288166	1,5	[L1] 310, 240	Attention à bien aligner les accolades qui ferment une condition/boucle ! Aussi, tu définis un MAX mais tu ne l'utilises pas partout où cela pourrait être considéré comme un magic number.
288252	2,0		warning 226 : why is <i>case</i> written in uppercase? warning [L2][L22] You probably ran into some trouble at the very end with your limitation on screen not being at 87 characters. As a result, you split the lines and try to align things correctly, but line wrapping occurred in an even weirder way.  Very good use of defines but do too much. INDICE_ZERO is essentially meaningless. It doesn't shed light on anything. Even INDICE_DEBUT would have made way more sense. However, you just don't need to use symbols for this. That said, when you have doubts about the use of a define, just go for it, and your mark may be grateful to you. Just make sure the name you use is relevant.  Well, my point is, there is not much to say.
288287	0,0	[L22] 10, 17, 20 [L11] 41 [L14] 133 [L1] 44, 74, 91, ...	La mise en page rend le code presque illisible. Il faut vraiment travailler là dessus. Notamment l'indentation des longues listes d'arguments de fonction, et la gestion des accolades.
288312	0,5	[L1] 45, 48 [L22] 57, 181, 264 [P5] 82, 102	Fais très attention à l'indentation, et aux noms que tu donnes à tes variables et tes fonctions. Ton code serait tout de suite presque parfait. Revois aussi comment gérer les magic numbers.
288423	1,5	[L1] 267	Bon code et l'indentation est cohérente, mais c'est plus lisible si tu mets un tab entier, ou du moins un peu plus d'espaces. Sinon, c'est un peu plus difficile de distinguer les différents blocs.
288530	0,0	[L1]264, 198 [L11] 57-69, 88-113, ... [L14] 127-132, 140-142 [P5] 100, 167, 236	Fais très attention à l'indentation de ton code, c'est impératif pour la lisibilité. Entre autre, le corps d'une fonction doit toujours être plus indenté que son nom et ses accolades entourantes. Attention aussi aux magic numbers.
288553	1,0	[L22] 23, 27... [P5] 37, 165, ...	Bon travail d'indentation, le code est très lisible mais il le serait encore plus en alignant les lignes trop longues entre elles. Attention à bien définir tes constantes au début du programme (en dehors des fonctions) à l'aide de #define ou constexpr.
288580	0,5	[L1] 111, 127 [P5] 91, [L22] 208, 213,	Ton indentation est spéciale mais je la considère correcte, cependant tu ne restes pas constant (dans tes if, des fois tu mets un espace et des fois pas, par exemple) donc je suis obligé de t'enlever des points. Sinon ton code est bien, je peux lire facilement. Je te conseille tout de même de regarder les conventions d'indentation et peut-être tu trouvera un style conventionnel qui te plaît. Attention au magic number (nombre de couleurs) en effet, cette valeur pourrait changer plus tard en fonction de tes images., même si je vois que tu as fait un effort je suis aussi obligé de t'enlever des points. Très bon code en général.

288820	1,5	[P5] 116, (139), 261, 285	<p>My first normal printing in a while... Thanks.  You don't need to create a <i>dummy</i> variable for each character you skip!  (actually you don't need to create one at all... see <code>isteam::ignore()</code>)  ==&gt; And after seeing <code>sortie_image</code> I realized you wanted to store them for the output. In this case, you would still be better off with a <i>string</i> !</p> <p>Don't write MAX in uppercase if it's just a variable.  ... but should it be?</p> <p>You can keep function names shorter...  I feel <code>traduction_intensite_a_code_couleur</code> is sort of... extreme.  Maybe you could write  <code>lafonctionquitraduitlesintensitesencodescouleursetquiprenddeuxparametres</code>.  Or just <code>intensite_vers_code_couleur</code>. Or <code>vers_code_couleur</code>. The choice of the clearest is naturally open to debate...</p> <p>Overall the code is kept clear and is properly formatted.</p>
288825	1,0	[L22] 139 [P5] 115, 203	<p>Ton indentation est spéciale mais je la considère correcte car tu restes constante.  Je te conseille tout de même de regarder les conventions d'indentation, tu en trouveras peut être une qui est plus à ton goût dans les conventions. De ce fait, ton code est bien indenté, presque parfait. Il y a un petit soucis avec les magic numbers mais je vois que tu as fait l'effort d'y penser. Continue comme ça, bravo!</p>
288838	1,5	[P5] 89, 131, 161,255, 256	<p>Code bien indenté et agréable à lire en général. Ce serait bien d'avoir un peu plus de commentaires dans le code. Attention aux magic number, ces valeurs paramétrables dans ton code qu'on voudrait pouvoir changer facilement ( nb de couleurs, format du fichier, etc. Bravo!</p>
289028	2,0		<p>Très bon style, constantes pertinentes, bravo!</p>
289067	1,5	[L14] 253	<p>Bravo pour une bonne indentation et bonne utilisation des <code>define</code>, malheureusement une petite erreur de double indentation ligne 253</p>
289121	2,0		<p>Très bon travail sur le style. Rien à redire.</p>
289122	1,5	[L22]36, 82, 165... ;	<p>De l'excellent travail ! Indentation et constantes bien utilisées. Attention toutefois à la convention L22 sur l'alignement des instructions sur plusieurs lignes.</p>
289188	1,5	[L22] 225	<p>N'hésites pas à aérer ton code. Les enums ne devraient pas servir à ça, utilises des <code>constexpr</code> à la place. La struct <code>Rvb</code> est une très bonne idée, <code>Rvbi</code> est moins utile. Attention aux conventions de nommage de tes variables.</p>
289234	1,5	[L14]105-106	<p>Mis à part un léger souci d'indentation, le code est propre et lisible.</p>
289285	1,5	[P5]267	<p>Bonne utilisation des <code>Defines</code> et <code>typedef</code>, bonne mise en page  Des commentaires plus explicites que simplement répéter le nom de la fonction serait utiles pour la lecture du code</p>
289292			
289304	1,0	[L14] 53, [P5] 60, 85, 116,	<p>Bonne indentation à l'exception de l'endroit mentionné, code très soigné. Bravo. Attention aux magic numbers, en effet, nous pouvions utiliser un autre nombre de couleurs ou vouloir faire varier le nombre de voisins dans le filtrage. Bonne utilisation des <code>define</code>.</p>
289324	1,5	[P5] 62	<p>Style excellent, les implémentations sont très adaptées.</p>
289596	1,5	[P5]102,196,214,221	<p>Code could use more/better comments ; instead of putting in literal values in the code (magic numbers), set them as a global constant (e.g. with <code>constexpr</code> or <code>#define</code>) with a name explaining what it represents: this makes the code more modular and clearer</p>
290067	1,5	[P5]40,104,155	<p>Plutôt que de coder en dur des valeurs dans le code, mets les en constante globale (p.ex. avec <code>constexpr</code> ou <code>#define</code>) avec un nom qui décrit ce que ça représente : cela rend le code plus modularisable et plus clair  De plus, mettre des commentaires pour expliquer ce que fait chaque fonction et comment serait utile</p>

290470	1,5	[P5] 54,106, 177, 180, 208, 249	Trop de magic numbers ! utiliser #define pour le nombre de composantes des couleurs
291182	1,5	[P5] 121, 208	Excellent code, agréable à la lecture. Attention à l'étourderie du nombre "3" mais il est clair que les conventions sont bien maîtrisées. Bon travail !
295745	1,5	[L2] 114 [L14] 161, 182	Bon respect des conventions en général, n'hésite pas à aérer un peu ton code !
295758	1,5	[L1] 245	Le style est bon, pas grand chose à améliorer de manière générale. Comme dit avant, réduire l'imprévisibilité du code avec des fonctions qui n'ont pas un type de retour void peut être bénéfique. Aussi, si il y a besoin d'optimiser la code pour qu'il s'exécute plus vite, une idée peut être de calculer des constantes qui ne changent pas (e.g. <code>rvb_couleurs.size()</code> ) en dehors des entêtes des boucles, afin de ne pas appeler inutilement cette méthode à chaque début d'itération.
295762	2,0		Style parfait, bravo!
295784	1,5	[P5] 96, 133, 232, 259	warning [L22] 190 Overall, the code is clear.
295798	0,5	[P5] 53 [L1] 115,135 [L14] 257-260	Bonne indentation générale. Malheureusement double indentation dans la fonction filtering et inconsistences dans <code>read_user_color</code> , utilisation du magic number 3 pour le nombre de composantes des couleurs
295808	1,0	[L22]29,109 [P5]67,177	Il faut mettre des commentaires pour expliquer les trucs spéciaux que tu mets, p.ex. lignes 44-45 ; et en général pour décrire chaque fonction
295810	1,5	[P5]27, 60, 82...	Très bon travail ! Le code est bien structuré et indenté. Continue tes efforts sur les Magic Numbers pour n'en laisser aucun qui traîne.
295813	0,5	[L22]45, 51, 52, 58 [R2] 11, 12 [P5]187, 208, ...	Cinq for-loops imbriqués sont un signe qu'il faut plus de décomposition en fonctions. Pour la prochaine fois, veille à imprimer ton code avec les numéros de lignes. Attention aussi aux magic numbers. Et à l'indentation. A l'exterieur des fonctions, <code>constexpr</code> ou <code>#define</code> doit être utilisé.
295821	1,5	[P5] 77, 97, 127, 246	Choisir des meilleurs noms de fonction et de variables. Attention aux magic numbers, aux variables en majuscule. Les commentaires sont très bien ! Le code est clair et agréable à lire.
295855	0,5	[L14] 31, 96, 153; 194, 211, 303 [L22] 191-192 [P5] 112	Dommage pour les mauvaises indentations à quelques endroits, le code en général est bien espacé et lisible.
295890	1,5	[L2]95	Excellent code, très bonne documentation et gestion des Magic Numbers. Attention à l'étourderie ligne 95 et aux constantes qui doivent être en majuscule
295913	0,5	[L22] 40, 56 [L1] 126, 231, 246 [L11] 260	Certains de tes DEFINE n'ont pas de sens ( <code>pixels_1</code> , <code>pixels_2...</code> ). Tu essaies d'éviter les magic numbers, mais tu ne résouds pas le problème si tes DEFINE ne sont pas plus explicites sur ce que représentent tes constantes.
295923	1,5	[L11] 67, 108, 169, 191	Attention, la dernière ligne d'une fonction fait partie de la fonction et doit être indentée en conséquence. Pense aussi à aérer ton code avec des lignes blanches, cela le rendra plus lisible.

295927	1,5	[P5] 34/241/214/247	<p>warning [P5] 45, 192  warning [P5] 68  <i>when you properly define a symbol, it might be a good thing to think of using it...</i></p> <p>(superslight) warning [P5] 175, 200  <i>To me it is just "expanding" a sum you could have written with a loop, but who knows when you encounter some picky corrector...</i>  <i>In case of doubt, just go for the safest way, it will also be time-saving.</i></p> <p>Parentheses may become crucial in <i>define</i> s when using operators, but they are probably not <i>necessary</i> on isolated values.  You might also want to align alll your <i>define</i> d values / comments.</p>
295954	1,5	[P5] 76, 114, 204	Bonne rigueur dans le respect des conventions. N'hésite pas à adopter un style un peu plus aéré.
296023	1,0	[L2] 94, 141, 164, ... [P5] 77	Code bien organisé avec de bons espacements. Très bonne utilisation des commentaires.
296026	2,0		Bon respect de l'indentation et des Magic Numbers. Attention tout de même à d'autres convention non évaluées ici : Pas de variables ou constantes numérotées par exemple. Il faut utiliser #define ou constexpr pour les Magic Numbers ! (const définit des variables globales : proscrit).
296032	1,5	[P5]52, 56, 115, 117, 147, 232, 234, 237, 263	Assez propre et lisible, bon choix de noms. Conseils: utilise des symboles avc constexpr, et met toujours les accolades, cela rendra le code plus simple à éditer pour d'autres programmeurs. Attention aussi aux magic numbers.
296046	1,5	[P5] 21, 33, 38, 45, ...	Bon respect des conventions. Les commentaires sont utiles et pourraient être davantage utilisés.
296048	1,0	[L1] 164, 173, [P5] 75, 79, 95	Implémentation plutôt bonne, et style soigné. Une petite remarque: la fonction contour travaille énormément pour pas grand chose, plutôt que de scanner toute l'image avec une double boucle for, c'est mieux de juste tracer 4 lignes.
296050	1,5	[P5] 121, 138...	Le code est très propre, bien documenté et agréable à la lecture. Attention, EPSILON n'est pas suffisant comme constante : veille à définir tous tes autres paramètres comme MAX etc...
296057	1,5	[P5] 51	Bon style, bon usage des commentaires.

296092	1	[L1] 206 [P5] 76, 114, 203, 214...	<p>the very beginning of your code is painful just to see.</p> <p>Skipping lines shouldn't be so costly, so you might want to give it a try.</p> <p>As suggested by the very same lines, the rest of the code is haunted with (dark) magic numbers.</p> <p>Bear in mind that this rule is about making the code easy to understand and more robust to slight changes within the specifications.</p> <p>Not sure the <i>blitz-closing</i> of brackets contributes significantly to clarity (e.g. l. 185). Please also note that you do not need brackets when using a single instruction within an if-statement.</p> <p>Triple slashes have a meaning and relate to documentation. For regular comments, you'd probably want to stick to the red-displayed ones (and possibly think of changing your syntax highlighting parameters).</p> <p>Don't strive to come up with atrocious workarounds when it comes to conventions. They might seem rigid, but they do have a meaning.</p> <p>If your functions tend to get too long, making your code highly illegible will only make things worse.</p> <p>In the end, you won't have lost the point for the 40-line rule, but you will have produced a very unpleasant code and almost managed to annoy a rather peaceful corrector...</p> <p>As much as you pay attention to the proper execution of your program, you should not lose sight of the fact that the code is meant to be read by a human.</p> <p>There might be no point in saying it's just a catastrophe, but it is difficult to convince oneself you were driven by pristine intentions formatting your code.</p> <p>See this as a (significant...) opportunity to improve the quality of your work, and</p>
296093	1,5	[L22] 42, 55, 188	Très bonne implémentation, soignée, aucune remarque à reproduire ici. Bonne utilisation des commentaires.
296142	2,0		Excellent. Mention particulière pour le bon travail sur les commentaires.
296165	1,5	[P5] 34,60, 62, 79, 121, 167,	Très bon code, très bonne indentation, mais tu n'as pas du tout fais attention aux magic numbers, il y en a partout. Pense au futur, si tu veux changer ton code avec un format different ou des valeurs differentes, alors tu devrais changer des valeurs un peu partout dans ton code au lieu de les change à un seul endroit. Aussi un peu plus de commentaires serait pas mal. Sinon bravo!
296169	1,0	[P5] 18, 57, 137, 152 [L1] 51	Code relativement bien espacé, attention aux magic numbers !
296189	1,0	[P5] 51, 68, 85, 193 [L22] 104, 172	Code tres agréable à lire, propre et logique. Attention à la mise en page, aère un peu plus et n'oublie pas les constexpr pour les magic numbers.
296202	0,5	[L22] 38,42,44,46.196,226 [L1] 136 [L14] 200	Indentation bonne en général excepté une double indentation en ligne 200. Cependant l'utilisation des accolades n'est pas consistante à travers le code (parfois à la fin de la ligne, parfois en dessous), exemple en ligne 136. les longues instructions sur plusieurs lignes ne sont pas alignées



296232	1,5	[P5] 198, 204	<p>warning: indentation poltergeist at line 124 ... 😬</p> <p>Try to keep the name of the symbols as meaningful as possible. As a start point I would suggest that you try not to actually include the number it hides in the name, to try to come up with something more significant.</p> <p>Leaving blank lines makes the code more convenient to browse (and restrain you from putting too much in a function!)</p> <p>Note that exit(0) is equivalent to exit(EXIT_SUCCESS). Is this what you meant?</p> <p>I wouldn't recommend closing several brackets in a row for nested loops. I don't think it is particularly beneficial to the code legibility...</p>
296235	1,0	[L14] 19, 20, 21, 22	Attention aux doubles indentations. Sinon, ton code est assez facile et agréable à lire.
296253	1,0	[L14] 131-134, [L2] 152, 202, 264, 265	Plusieurs indentations incorrectes: tu mets souvent des doubles indentations. Plusieurs lignes dépassent aussi la limite de caractères, ce qui induit un saut à la ligne, pénalisé dans les conventions.
296351	1,5	[L14] 327 [P5] 125	Style soigné, dommage pour les inattentions. Bon usage des commentaires. Attention aux noms des constantes pas très clairs ("CONDITION")
296372	1,5	[P5] 102, 164, 168	Très bon respect des conventions, le code est bien espacé et commenté.
296399	0,0	[L1] 43, 44, 66, 67, 68, [L11] 40, 125, 154, [L22] 28, 35, [P5] 54, 58, 79, 193	Les implémentations sont bonnes. Malheureusement les conventions de style ne sont presque pas du tout respectées, ce qui rend le code extrêmement dur à lire. Mettre des commentaires aurait été une bonne idée pour rendre le code plus compréhensible.
296442	0,0	[L14] 309, 314, 319, 102, [P11] too many lines, [P22] 40,41, 44,45, 102, and many others, [P5] 199	Tu ne respectes pas les conventions de programmation et donc ton code est très difficile à lire et pas du tout agréable. Je te conseille de jeter un petit coup d'oeil à ces conventions pour la suite.
296443	1,0	[L22] 99, 242, [P5] 71	Bon code. Fais attention aux lignes trop longues et que tu sépares en deux lignes, la deuxième doit être au bon niveau de la première pour que ce soit bien visible. Petite utilisation de magic number pour le nombre de couleurs dans un for ( cela pourrait varier)
296451	1,5	[L1] 380	En dehors de la ligne 380, ton code est très joli et bien indenté. Cela le rend très facile à lire ! Petit warning aux lignes 134 et 135, tu ne devrais pas avoir autant d'indentations après la première partie du COUT
296465	1,0	[L1] 325, 329, 384 [P5]68, 149, 154, 160, 198, 309, 312, 322	Le code est clair et compréhensible, il faut juste un peu plus d'attention aux détails d'indentation et aux paramètres de projet.
296487	0,0	[L11]28-91,144-186,...[L1]59,70,136,...[L14]74-89,127-130,...[P5]117,...	Les conventions de programmation sont hélas à retravailler...
296573	1,5	[L11] 43, [P5] 182	Le style est bon, mais malheureusement pas constant sur tout le code. Mettre au moins quelques commentaires peut être bénéfique pour la clarté du code. Il est possible de remplacer par compteur++ ou compteur-- aux lignes 174 et 176.
296575			

296642	1,0	[L1]43, 101 [P5]47, 108, 121...	Indentation dans l'ensemble juste, attention toutefois à quelques erreurs subsistantes. Au niveau des magic numbers, attention, tu en a oublié quelques uns
296643	1,5	[L14]215-224	Code bien commenté et globalement bien mis en page, agréable à lire :)
296644	1,5	[P5]65, 128, 191...	Très bon style, il manque juste les constantes pour remplacer les magic numbers.
296759	1,5	[P5] 79, 113, 159, 180, ...	Excellente présentation du code bravo ! Cela rend la relecture beaucoup plus facile et agréable.
296815	1,0	[L1] 202, 245, [P5] 89, 97	Le style est soigné, les implémentations des fonctions sont remarquablement logique et faciles à comprendre. Après avoir défini des constantes globales, il s'agit de s'assurer de bien les utiliser partout, c'est dommage par exemple d'avoir défini NBRMAX et d'avoir oublié de l'utiliser à la ligne 97. Les commentaires divisant le code en parties sont bien, mais en rajouter pour expliquer l'implémentation peut aussi être une bonne idée.
296816	1,5	[P5] 343, 350, 356	Le style est plutôt bon, l'indentation est remarquable. Dommage pour les nombres magiques, nb_pixel (le nombre de voisins) par exemple aurait du être défini comme constante globale (e.g. avec define). Aussi, rajouter un minimum de commentaires peut s'avérer fort utile, même si le code est assez facile à comprendre dans ce cas.
296820	1,0	[L22]26, 30, 33, 36, 39 [P5]64, 69, 84, 101, 141, 165,...	Attention au magic numbers, utilise plutôt des constantes définies au début du programme. Aligne aussi bien ton code, pour la facilité de lecture. Bon usage de commentaires.
296827	1,5	P[22] 373	Code agréable à lire et bien commenté, bravo! Dommage pour la petite inconsistance lors des retours à la ligne pour les prototypes trop longs. Bravo!
296833	0.5	[L1]65.. [L22]108, 175.. [P5] 60, 63...	Bon travail mais il faut être plus rigoureux dans le respect des conventions : Aligne tes expressions multilignes pour rendre le code plus clair et garde un style constant d'ndentation dans le code. Les noms de constantes doivent être plus explicite que "un" et "zero" sinon ils n'ont aucun intérêt.
296853	1,0	[L14] 120 [P5] 63, 83, ....	Très bon travail avec un code agréable à la lecture. Attention, il est important de définir tes magic numbers pour rendre ton code plus flexible ! Utilise des #define ou constexpr à cette fin.
296867	1,5	[L11]52	Le style est très bon, bravo pour les constantes. Il manque une relecture attentive, encore une ou deux erreurs d'inattention. Pour augmenter encore la lisibilité, ajoute un espace entre la fin d'une fonction et le début de la suivante.
296935	1,5	[P5] 31, 44, 85, 195	Note: exit(0) is equivalent to exit(EXIT_SUCCESS)  Leaving some blank lines can help with code clarity.  Calling a boolean notSomething feels sort or weird... I don't think it makes things simpler.  You don't need parentheses around a returned value. In the end the code is still rather clear.
296952	1,0	[L1]104, 115, 126 ; [P5] 87, 113 ;	Dans l'ensemble, le style est plutôt bon. Attention toutefois, dans certaines parties de ton code l'indentation est particulièrement chaotique : pense à plus te relire. Aussi, utilise plus de constantes pour les magic numbers.
296957	1,0	[L22]19,49 ; [L14]202 ; [P5]36, 70...	Dans l'ensemble code très soigné, bon travail ! Fais toutefois plus attention à la convention L22 sur l'alignement des instructions sur plusieurs lignes : c'est très important pour la clarté de ladite instruction pour un éventuel relecteur ! Attention également aux Magic Numbers : utilise plus de constantes

296958	1,5	[L1]244,275	Code très propre et soigné, beau travail ! Quelques erreurs d'inattention t'empêchent d'obtenir la note maximale mais dans l'ensemble continue comme ça
296960	1,0	[R2]16 [P5]106, 177	Propre et facile à lire. Attention à l'utilisation de single statements sans accolades, elle est permise, mais en général déconseillée car rend vulnérable aux erreurs le code. Attention aussi aux magic numbers. A l'extérieur des fonctions, constexpr et #define doivent être privilégiés.
297072	0,0	[L1]norm_input & threshold_value_input methods,[L22]norm_input method,[P5]threshold_value_input & equality_counter & filter methods	Please put line numbers in as asked... Code could use some comments ; instead of putting in literal values in the code (magic numbers), set them as a global constant (e.g. with constexpr or #define) with a name explaining what it represents: this makes the code more modular and clearer
297075	1,0	[L22]28, 173, 233..., [P5]147, 202...	Le style est très bon et lisible. Fais attention aux constantes et à la convention [L22], et ça sera parfait.
297096			
297097	1,5	[P5]113,148,152,158,160,201,209,255,261	Code bien commenté Plutôt que de coder en dur des valeurs dans le code, mets les en constante globale (p.ex. avec constexpr ou #define) avec un nom qui décrit ce que ça représente : cela rend le code plus modularisable et plus clair
297151	1,0	[L22]58 ; [L1]236	Style dans l'ensemble très propre et appliqué. Dommage qu'il y aie quelques erreurs d'inattention !
297176	1,5	[P5]59, 104,116...	L'indentation est très bonne, bon travail ! Attention cependant aux magic numbers : pense bien à créer des constantes pour les remplacer
297202	1,0	[L1] 58, 305 [P5] 66, 77, 103, 141	Attention à la lisibilité, tu as beaucoup de fors imbriqués et ton code est très compact, n'aies pas peur de mettre des sauts de ligne. La représentation des RGB pourrait être plus claire que juste 3 indexes dans un tableau. Les typedef pourraient aussi être plus explicites. Mis à part ça, le code est bien présenté, aucune grosse erreur de mise en page, c'est très bien.
297250	1,0	[P5] 41, 76, 135 [L11] 219	Bon usage des commentaires, malheureusement le code manque beaucoup trop d'espaces. Les boucles et les if sur une ligne sont une mauvaise pratique à proscrire ! La logique générale du code reste relativement bonne.
297284	0,5	[L2] 35, 156, 194, 203 [L11] 78 [L14] 150, 151	Bonnes conventions dans l'ensemble, excepté pour le non-respect de la limite de caractères par ligne et 1-2 fautes d'indentations. Bon usage des define pour éviter les magic numbers !
297290	1,5	[P5]126, 130, 146, 156,219	Le code est très propre et facile à lire, excellent choix de noms de variables et de structures. Seul bémol: attention aux magic numbers!
297768	1,0	[L1]42,61,99... ; [P5]38,41,44...	Dans l'ensemble ton style est beaucoup trop compact : utilise plus d'espaces et de retours à la ligne afin de le rendre plus lisible. Egalement, fais plus attention à la présence de magic numbers.

297797	0,5	[L1] 67, 122 [L14] 176, 190 [P5] 122, 146, 251	<p>Regarding indentation and brackets, try to stick to, for instance:</p> <pre> for (...) {     if (...)         function_call();      while (...)     {         ...     } } </pre> <p>For the time being it sort of looks like random formatting, and it is not especially pleasant.</p> <p>Don't be discouraged by the fact you lost quite a few points for style. The formatting is very important. The code you write is meant to be read by a human. Correcting these few mistakes, you will suddenly get better results! Keep going!</p> <p>Oh, by the way: Poltergeist spotted at line 63. ... 😬 !</p>
297846	1,5	[P5]94, 154, 168...	Très bon style, il manque juste les constantes pour remplacer les magic numbers. Attention, variable globale.
298006	1,5	[P5]104,123,132...	Code globalement clair et bien indenté mais attention il faut absolument définir tes Magic Numbers pour n'avoir que le strict nécessaire de nombres bruts dans ton code (ex : EPSILON et MAX)
298011	1,0	[L22]88, [P5]53, 109, 121...	Très bonne indentation et l'alignement est bien respecté sauf une petite étourderie ligne 88 (le non alignement des paramètres de seuillage trouble la lecture). Attention, il faut définir tous tes Magic Numbers !! (et les utiliser : l. 109).
298386	2,0		Excellent code, agréable à la lecture. Les commentaires sont particulièrement pertinents. Continue !
299066	0,5	[L14]89, 120, 167, [L2]26, 135, 175..., [P5]100, 126, 219...	Les fautes commises sont faciles à corriger, il manque l'attention aux détails. Le code n'a pas besoin de contenir autant d'espaces, même s'il vaut mieux être un peu trop aéré que pas assez.
299245	1,5	[P5]176, 213,237	Très bon respect des conventions ! Continue tes efforts pour ne laisser aucun magic number.
299248	1,5	[P5] 80, 93, 95, ...	Excellente présentation du code bravo ! Bonne rigueur dans le respect des conventions.
299274	1,5	[P5]44,51,85...	Code clair, bien indenté et bien espacé mais attention à utiliser les constantes définies (l.85) et surtout définis autant de magic numbers qu'il en faut (EPSILON et autres).
299290	0,5	[P5] 269, 325, [L1] 98, 189, 191 [L22] 197, 217	Choisis des meilleurs noms pour tes variables, fonctions, typedef. Sinon le code est très difficile à lire. N'hésites pas aussi à aérer ton code. Attention à l'indentation. Les commentaires sont très utiles, merci ! C'est toujours une bonne pratique !
299391	1,0	[L1] 73, 261, 267 [L14] 244-249, 306, 332	Tu mets des indentations beaucoup trop grandes dans certaines parties de ton code. Dans d'autres parties, tu oublies d'indenter les instructions contenues dans un bloc de condition. Sinon, bon usage des DEFINES !

299450	0,5	[L14] 348 [L22] 250, [P5] 118, 207,255, 285, 342, 346, 356,	Bon code en général, agréable à lire et bien indenté excepté quelques occasions. Petit souci aussi avec les magic numbers pour le nombre de pixels voisins, etc, et longues lignes qui doivent être correctement alignées à leur retour. . C'est très bien réfléchi de penser au overflow du nombre de filtrage, hélas tu as crée un magic number ( tu pouvais créer une constante globale pour y rémédier, imagine si on changeait le type des filtrages en long int, alors dans ce cas tu devrais aussi changer ce chiffre d'overflow.)
299479	2,0		Parfait ! L'indentation et les alignements sont excellents. Attention tout de même à la ligne 278 (warning).
299492	1,0	[L2]41, 76, 190..., [P5]50, 116, 207...	L'indentation et le formatage sont bons, mais le code est rendu peu lisible par le choix des noms de fonctions et de variables. Tes noms sont très longs! Essaie de les raccourcir. Par exemple: pas besoin de précéder les noms de matrices par "tab_", c'est impliqué par le type!
299496	1,5	[P5]44,118,198,258,265	Bravo pour les conventions de programmation. Il reste toutefois quelques magic numbers
299506	1,5	[P5] 56, 187, 245, ....	Tip of the day: Using symbols to replace magic numbers help other people (including your future self) understand and maintain your code !
299554	1,5	[P5] 86, 101, 123	Les commentaires sont très bien, merci ! Attention à la duplication de code et essaie peut-être de trouver des noms de variables/structs un peu plus clairs. Aussi, pense à aérer tes lignes avec des espaces. Sinon, on voit que tu as bien compris le problème et tu as de très bonnes idées. C'est très bien.
299561	0,5	[L14]24-30[L11]51- 67,92- 93,99,...[L1]62,96,97,...	Les conventions de programmation sont plus ou moins respectées sauf pour les indentatiions où il y a plein de problèmes.
299645			
299667	1,5	[P5] 33, 37, 76, ...	Très bon travail : un code propre, bien indenté, bien aligné et bien commenté. Attention à ta gestion des Magic Numbers : il faut des #define ou constexpr en début de programme pour le rendre plus flexible.
299689	1,0	[L22] 205, [P5] 112,	Code très bien indenté et agréable à lire, bien joué pour les variables pour les codes d'erreur, dommage que t'aies oublié de faire la même pour la couleur maximale (255). Quand tu as une ligne trop longue dans le prototypage d'une fonction, la deuxième ligne doit être alignée avec le début des arguments. Bravo sinon, tres bien!
299724	1,0	[L1] 69, 155 [P5] 109	Chisir des meilleurs noms de variable. À part ça, très agréable à lire, bonne présentation !
299882	1,5	[L1] 104, 142,	Le plus gros défaut est le choix des noms de variable qui sont souvent un peu obscures. Gardes un style d'indentation constant et n'hésites pas à aérer ton code.
299885	1,0	[L1] 108, 157, 223, 287, 296 [P5] 122	Ton code est bien indenté. Ton indentation est spéciale mais je la considère correcte, cependant il faut que tu restes consistante ce qui n'est pas le cas à quelques reprises. Je vois que tu as fait gaffe aux magic numbers raison pour laquelle j'ai de la peine à t'enlever des points pour ton petit oubli de la valeur de threshold hardcodée. Pense à plutôt créer des constantes globales, surtout si elles peuvent se répéter dans ton code. Ton code est très bien en général, continue comme ça!
299892	1,0	[L22]174, 246, [P5]118, 134, 349	Le style est bon, lisible et bien aéré (peut-être un peu trop pour les define et les prototypes). Pour la convention [L22], tu la suis correctement en ligne 214. Prend cet exemple là pour le reste.
299894	1,5	[P5]213	Très propre et lisible. Bon usage de #define. A part un malheureux magic number, seul conseil: toujours mettre les accolades, même dans le cas de "single statement", cela met ton code à l'abri d'erreurs d'inattention.
299895	1,0	[L1] 72 [P5] 143	Le plus gros problème est le choix des noms de variable, de fonction et des typedef. "a" n'est pas du tout explicite, pareil pour un "saisir_tab2". Attention à l'indentation. Peut être aérer tes lignes avec plus de caractères espace. Attention aux comparaison double int.

299964	1,0	[L11] 16-19 [P5] 126, 146, 204	Attention aux magic number. Par convention, les constantes créées avec #define sont toutes en majuscule. Evite aussi de mettre plusieurs commandes sur une seule ligne, cela complique la lecture.
299971	1,5	[L22] 100, 227	Ton style est très bien, code est parfaitement indenté mais attention aux longues lignes en deux temps où tu dois aligner correctement la deuxième à la première. Bravo!
300038	0,0	[P5]33,42,43,45,128,...[L1]47,53,70,...,[L14]41[L22]343	Il y a de nombreux soucis pour la présentation du code. Reporte-toi au document conventions de programmation. Point positif: ton code est naturellement aéré et est facile à lire.
300048	0,5	[L1] 83, 105 [L22] 174, 187 [P5] 15, 70, 119 ...	Le code est propre et bien indenté sauf une étourderie lignes 83 et 105. Attention à bien définir le nombre suffisant de constantes pour tes Magic Numbers.
300089	1,0	[L22] 70, 109, 160, 183, 241, [P5] 223,	Tes noms de variables et fonctions sont un peu trop longs ce qui fait que tu as des lignes un peu trop longues. Pense à avoir des noms plus concis. A un moment tu utilises la valeur 0.001, mais cet valeur pourrait changer en paramètre selon ton envie (ou celle du prof) plus tard. C'est donc un magic number, ce serait bien de l'avoir en tant que variable globale pour pouvoir facilement la modifier à volonté. TON code est très bien indenté à l'exception des longues lignes, je vois que tu as fait de l'effort et j'ai de la peine à enlever ces points, désolé. Bravo!
300093	0,0	[L1]41-64, 96-104, 140, ... [L14]198 [L22] 51, 54, 58,... [P5] 117, 119, 129	Le code est bien aéré, mais le positionnement des accolades fermantes le rend difficile à lire (et à debugger). Plus de consistance dans l'indentation nécessaire. Attention aussi aux magic numbers.
300114	0,5	[L2]37,43,69,71,74,105, 119,140,150,154,161,163,...[L22]45,46[P5]322,355	Tu as de nombreux problèmes de retour à la ligne. Peut-être dû à la manière dont ton code est imprimé. Pense à contrôler la mise en page. Sinon, tu utilises astucieusement les symboles pour éviter de nombreux magic numbers.
300131	1,5	[P5]71, 80, 165...	L'indentation est très propre sur tout le code, bravo ! Attention néanmoins aux magic number, pense à plus utiliser les constantes
300166	0,5	[L14]69-71,78-80,96-102,113-114...[L1]100[L22]108,	Il y a de nombreux problèmes vis-à-vis des conventions de programmation. Tu as par contre bien utilisé les symboles pour être l'un des rares sans magic numbers.
300247	1,0	[L11] 173 [L1] 101, 97, 357-360	Attention à bien indenter les parties du code dans les conditions, les boucles mais aussi les fonctions ! Sinon, toute ta fonction main semble bien indentée. Petit warning pour le magic number de la ligne 273 et le retour à la ligne 20.
300253	0,5	[P5] 47,99,118, 126,196,202,219, [L22] 31 ,136, [L14] 123-130	Bonne indentation bravo, fais attention aux alignements sur plusieurs lignes ainsi que les doubles indentation. Prochaine fois utilise aussi des define ou constexpr au lieu de magic numbers
300284	0,0	[L1] 147, 157, [L11] 70, 76, 101, [L22] 156, 303, [P5] 141	Les implémentations sont logiques, bien réalisées. Dommage pour les conventions de style, qui sont respectées de manière assez approximative. Pour optimiser la vitesse d'exécution du code, il peut être utile de calculer les tailles des tableaux dans les entêtes des boucles en dehors de ces entêtes afin d'éviter de les recalculer au début de chaque itération.
300411	1,5	[P5]22, 106, 110...	Code bien documenté et clair à la lecture. Attention à ta gestion des Magic Numbers (définis autant de constantes qu'il en faut).
300443	1,5	[P5]353	A un magic number près, ce code est parfaitement structuré, et bien documenté
300456	1,5	[P5]63,71,122,150	Bonne mise en page Plus de commentaires pour mieux expliquer les différentes parties de ton code auraient été bien Plutôt que de coder en dur des valeurs dans le code, mets les en constante globale (p.ex. avec constexpr ou #define) avec un nom qui décrit ce que ça représente : cela rend le code plus modularisable et plus clair

300540	1,0	[L1] 280 [P5] 144, 167, 192, 222, 233, 240...	<p>As we might fear it seeing the first couple of lines, the code is overrun with magic numbers! ㄿㄿㄿ</p> <p>Bear in mind that this rule is about making the code easy to understand and more robust to slight changes within the specifications.</p> <p>warning [L1] 192</p> <p>It makes the corrector unhappy to take off points for that kind of problems :( Think of them next time! (... and every little thing's gonna be alright ♪)</p>
300556	0,5	[L1] 100 [L2] 21, 24, 65, 152, 189, ... [P5] 36, 142, 192, 227	<p>warning [L14] 41</p> <p>Indentation is quite weird throughout the main function</p> <p>There might have been some confusion with the magic numbers. You basically didn't use any symbol to replace them, but you also write MAX in capital letters... as if it was one... You are supposed not to leave any raw number in your program by replacing them with some literal expressions so they become meaningful and the code can be adapted easily.</p> <p>You should pay more attention to the basic conventions. There is just a set of simple rules to follow so your code can look clearer and be more easily understood by everyone... and for now it will above all stop the point haemorrhage...</p> <p>Array translates into <i>tableau</i> ^^' [i+ : you can ask Father Christmas for printer cartdriges, though it looks quite cool this way...]</p>
300558	2,0		<p>warning [L22]196 warning alignment 223</p> <p>About symbols and magic numbers: you probably don't need to define things like PREMIERE_LIGNE, AVANT_DERNIERE_LIGNE... However, in case of doubt, I strongly recommend that you write your the way you did. You will not be penalized for extra symbols, but you could be for magic numbers.</p> <p>When it comes to legibility, I have to say that the code didn't live up to my expectations after turning the first page. Black and white probably doesn't help... but I think you should skip more lines and try to make your code a bit more "airy".</p> <p>Anyway, there is not much to say, and I appreciate it. Thanks! x)</p>
300559	1,5	[P5] 155, 279	<p>The code formatting is ok.</p> <p>Special choice of colors... which wouldn't necessarily be a problem if the light grey was not that light...</p>
300594	1,5	[P5]50,170,296-298	Malgré quelques magic numbers, tu as très bien suivi les conventions de programmation
300615	1,5	[L22] 172	Mis à part un problème d'indentation, ce code est extrêmement propre, c'est très bien.
300620	1,5	[P5]65, 94, 196...	Code très bien indenté, bon travail de ce côté-là. Fais toutefois attention aux magic numbers, pense à utiliser des constantes pour les remplacer !

300649	1,0	[L22]24, 83... ; [L2]94 ; [P5]112, 128, 244...	Indentation tout à fait correcte sur l'ensemble du code, bravo ! Fais toutefois plus attention à la convention L22 sur l'alignement des instructions sur plusieurs lignes. Au niveau des magic numbers, tu en as repéré un certain nombre, mais en a laissé filer d'autres, attention
300664	2,0		
300669	1,5	[P5]124,193,196,199	Bonne mise en page Pense par contre à mettre des commentaires pour expliquer ton code De plus il aurait fallu des constantes pour le seuil du delta et les voisins
300780	1,5	[P5]37,80,269	Le code est très bien présenté malgré quelques magic numbers. Bravo !
300782	1,0	[L22]31,126[P5]42,104-107,114,133,189	Les conventions de programmation sont bien respectées mais il y a quelques soucis avec les longues lignes et les magic numbers.
300803	1,0	[L1] 151, 223	Très bon code en général à l'exception des deux petites fautes d'indentation (d'attention plutôt) aux endroits mentionnés. Bravo!
300840	1,5	[L1] 144	La mise un page est pas mal, tu pourrais parfois utiliser des noms de variable plus clairs.
300845	0,5	[L1] 317 vs 362 [L22] 195, 206.. [P5] 26, 29, 164...	Très bon code mais il faut être plus rigoureux avec les conventions . Attention aux magic numbers (continue tes efforts), ton style d'indentation variable (choisir un style et s'y tenir) et attention au bon alignement de tes expressions trop longues pour simplifier leur compréhension.
300882	1,5	[L22] 60	Penses à peut être utiliser des typedef pour tes vectors. Les constexpr sont parfaits. Code très agréable à lire, bravo !
300883	0,5	[L2] [L14] [P5]	WARNING: No line numbers, no syntax highlighting  Alignment of ++ in filtrage?  Don't use const variables within functions instead of define/constexpr  Mysterious spaces here and there at the beginning of lines.  If it wasn't for the very special printing, the code would probably be sort of legible.
300886	1,0	[P5]106,115,138,177[L11]293-296	Tu as bien suivi les conventions de programmation mais tu as fait plusieurs magic numbers
300899	1,5	[P5] 93	Le code est bien indenté et le style est bon. Attention aux magic numbers surtout les valeurs mises dans tes modulus. Ce serait sympa de plus commenter ton code, pour toi et pour nous assistants, peut être tu retravailleras dessus plus tard et tu seras eheureux d'avoir plus commenté ton code. Fais gaffe si tu as trop d'arguments dans la déclaration de fonction, tu pourrais faire mieux. Cela dit, bravo.
300910	0,0	[L1]115, 148, [L11]main, [L14]23, 69, [L2]13, 135, 226..., [P5]5, 51, 142...	Le style est très chaotique et rend le code difficile à lire. L'indentation est la plupart du temps beaucoup trop petite (ex: ligne 198) ou incorrecte. Pour un bon exemple regarde ta fonction bordure, qui est correcte. C'est le point le plus important, les constantes et le wrapping sont secondaires dans ce cas.
300930	0,0	[L1] 206 [L14] 232, 236 [L22] 191 [P5] 59, 68, 73, ...	Très dommage pour le non respect des conventions, retirer les magic numbers et écrire quelques commentaires augmenterais grandement la lisibilité du code !
301001	0,5	[L2] 30, 33, 138, 160, 166 [L14] 237-240 [P5] 15-20	Attention aux magic numbers : tu essaies de pallier au problème avec les define, mais ce que tu fais est une mauvaise pratique. Le problème d'un magic number est qu'on ne sait pas à quoi il correspond. L'idée d'un define est de le remplacer par une expression plus explicite (ex: MAX, SEUIL..). Si tu remplaces 1 par UN et 2 par DEUX, tu n'as pas résolu le problème initial. Sinon, attention aux retours à la ligne (plusieurs lignes dépassent la limite de caractères). Enfin, une de tes boucles for a un surplus d'indentations.
301030	1,5	[P5]68, 85, 128	Très bon travail sur l'indentation, dommage que tu aies oublié quelques magic numbers



301033	1,5	[P5]44, 85	Très bon travail sur le style ! Une excellente indentation et une bonne utilisation des constantes dans l'ensemble. Pour ce qui est des constantes, privilégie de toutes les déclarer au départ afin qu'elles soient visibles. Juste dommage que tu aies raté un magic number, mais très bien dans l'ensemble !
301061	1,0	[L2] 40, 117, ... [L22] 218, 257	Attention aux lignes trop longues et à la mise en page des listes d'arguments trop longues. Utilises aussi des typedef pour clarifier le code.
301070	1,0	[L2] 150, 209, [P5] 63, 64, 65, 66, 67	Attention aux retours à la ligne et aux magic numbers ! Sinon, le code est clair et agréable à lire.
301107	1,0	[P5]103,121,124,126,154,160,250 [L22] 176	Bonne indentation du code, fais juste attention aux alignements Beaucoup trop d'usage de magic numbers, la prochaine fois utilise des constantes paramétrables
301221	1,0	[L1] 157, [P5]155, 186, 198...	Dans l'ensemble, le style est bon, mais il manque une relecture attentive pour obtenir tous les points.
301225	1,5	[P5]51, 82, 93, 113, 184	Très bon style, le code est structuré de manière très claire. il manque juste les constantes pour remplacer les magic numbers. Il manque aussi une relecture attentive pour les erreurs d'inattention (lignes 139, 216)!
301226			
301265	1,0	[L14]308 [P5] 177,184..	Bon respect des conventions. Attention aux doubles indentations (non volontaires ?). Continue tes efforts sur les #define pour ne laisser aucun paramètre numérique indéfini (EPSILON par exemple).
301266	1,0	[L1]27, 168... [P5]37, 140, 157...	Il faut revoir plsu rigoureusement l'indentation du code : chaque corps de fonction doit être décalée d'au minimum deux espaces par rapport à leur nom. De même pour les boucles et conditions. Attention aussi aux Magic Numbers qui doivent tous être définis au début du programme par des #define ou constexpr.
301268	1,0	[L1] 96, 108, 112, [P5] 36, 56, 86	Le style est correct, mais malheureusement pas constant sur tout le code. Mettre un peu plus de commentaires peut être une bonne idée. Le code entre les lignes 170 et 178 est répétitif, il y a plusieurs solutions à cela: par exemple, réunir tous les ifs en un seul, ou créer une liste des voisins.
301271	1,0	[L2] 161, 165, 254... [P5] 53, 81, 125...	Bon respect des conventions (indentation et alignement) mais le wrapping n'est pas respecté à multiples reprises : respecte la ligne verte de Geany des 87 caractères). Attention aux Magic Numbers.
301272	1,5	[P5] 19, 173, 176...	Très bon code , bien indenté et structuré. Attention néanmoins à définir autant de constantes qu'il en faut pour n'avoir aucun Magic Number.
301291	1,5	[P5] 114,144,147,185,203,262	Bonne indentation et consistance, malheureusement, beaucoup trop d'utilisation des magic number
301295	0,0	[P5] 53, 77 [L1] 90-94 [L2] 97, 224, 259 [L14] 270-309	Attention aux magic numbers, aux retours à la ligne et aux indentations trop grandes ou trop petites !
301300	0,5	[L1] 205, 272, [L22] 131, [P5] 101, 109, 180,	Bon code en général mais il y a de l'inconsistance dans ton style. EN effet, des fois tu mets des espaces après les for, if et d'autres fois pas. Fais attention aux magic numbers, tu n'as aucune variable globale alors que tu utilises les mêmes valeurs à plusieurs reprises. COde agréable à lire mais qui manque un peu de commentaires.
301303	1,0	[L1] 163, 200, 241, [P5] 39, 269, 280	Le style est correct, les commentaires sont très bien utilisés. Il aurait fallu plus soigner le style, et diviser le code en méthodes. Aussi, il faut éviter les nombres magiques, par exemple il faut déclarer 255 avec define en constante globale et lui donner un nom symbolique.
301305	1,5	[P5] 213, 265...	Très bon travail, une indentation irréprochable. Attention à bien définir assez de constantes pour supprimer tous les Magic Numbers !
301319	1,0	[L22] 55, 188, 247, [P5] 105, 179	Les implémentations sont concises et claires, et les commentaires améliorent beaucoup la clarté du code. Le style est soigné. Les chaînes interminables de cout sont par contre à éviter, surtout du moment où elles commencent à faire plusieurs lignes: les remplacer par plusieurs cout (par ex. ligne 246). Aussi, attention à la longueur des lignes: la ligne 122 dépasse presque.

301327			
301358	1,5	[L22] 132	Code très propre, clair, agréable à lire. Attention aux conventions de nommage des variables/fonctions et à l'indentation.
301366	1,5	[P5] 268, 278, 338	Présentation du code remarquable ! Excellent usage des commentaires bravo cela rend la relecture simple et agréable. Très dommage pour les magic numbers...
301367			
301369	1,0	[L22] 117, [P5] 252	Code agréable à lire est bien indenté, je vois que tu as fait des efforts. C'est à ce moment là que nous avons de la peine à enlever des points pour des petites fautes d'attention. Aussi, tu hardcodes le nombre de voisins alors que ceci est un paramètre qu'on pourrait vouloir changer -> magic number. Beau travail!
301395	2,0	[P5] 265, 272	Style parfait, bravo! Très bon code mais attention aux magic numbers tels que le nombre de voisins et de couleurs, ceux ci pourraient varier en tant que paramètre.
301403	1,5	[L22] 231	Les commentaires sont une bonne pratique ! Le code est extrêmement clair et agréable à lire. bravo !
301407	1,5	[P5] 56, 77, 79, 124, ..	Tu n'as aucune variable globale, c'est une bonne pratique d'en avoir si tu utilises ces variables à plusieurs endroits de ton code ( p.e. 255 MAX_COLOR). Cela implique que tu as des magic numbers partout donc tu es pénalisé. Ce serait sympa d'avoir plus de commentaires, imagine si tu veux retravailler sur ce projet dans quelques années. Certaines de tes fonctions ne passent pas les tableaux par référence, tes fonctions vont donc copier ces arguments et ce sera plus lent à l'exécution. Bonne indentation.
301411	1,5	[P5] 140, 164	Le code est excellent! Remarquablement logique, bien structuré, bien commenté, et en plus optimisé! Très facile à lire et à comprendre.
301412			
301418	1,5	[P5]39, 99, 126...	Très bon style, il manque juste les constantes pour remplacer les magic numbers.
301437	1,5	[P5] 62, 117, 122, 136, 198	warning 188: alignment Note that exit(0) is equivalent to exit(EXIT_SUCCESS). Is this what you meant?  You should replace "magic numbers" with meaningful symbols, so the code is easier to understand and maintain.  The code is clear and well-presented.
301438	1,5	[L22] 137, 145, 164, 182,	Style très bon à l'exception des petits retours à la ligne lors des lignes trop longues. En effet, celles-ci doivent s'aligner. Très bon code, bravo!
301446	0,5	[L1] 278 [L22] 38, 65, 144, 145 , 198, 206, [P5] 91, 244, 247	Cela me fait de la peine de t'enlever autant de points car je vois que tu as mis de l'effort. Je sais pas si les retours à la ligne sont voulus ou si c'est toi mais en tout cas ils sont incorrects, alors viens nous voir avec ton vrai code et si c'est correct dans ton vrai code on reverra. Tu as tout de même des petites fautes de magic numbers et une petite faute d'attention sur une indentation d'un return. Ton code est très bien sinon.
301450	1,5	[L22]145 ;	Le style est vraiment soigné, ça fait plaisir à voir. Une petite erreur d'alignement, mais au delà de ça : bon boulot !
301452	1,5	[P5] 188	Style bien aéré, bon respect des conventions.
301458	1,5	[P5]212,217,227	choix des noms correct en général. La dimension du tableau du voisinage est un magic number ; de plus il est utilisé à plusieurs endroits.
301464	1,0	[L22]31, 317, 336 ; [L14]143 ;	Très bon style de manière générale ! Fais un peu plus attention à l'alignement des instructions de plus d'une ligne, mais du reste, tout est très bien
301469	1,0	[P5]86[L22]183,200,236	C'est un code avec une présentation magnifique ! Revois peut-être la manière de décomposer une longue ligne de code.

301482	1,5	[P5] 146, 193 ...	Bon travail d'indentation et d'alignement ! Attention cependant à la définition de constantes pour tes Magic Numbers. Attention aussi à l'étourderie ligne 163 (warning).
301486	1,0	[P5]76,88,96,138,153,156[L22]156,195	Ce code est bien présenté. Aère-le un peu plus peut-être. Sinon, il y a quelques fautes pour les magic numbers et les longues lignes.
301491	2,0		
301494	1,0	[L14]165, [P5]98, 123, 157...	Le style est bon, il manque encore des constantes. Tu as aussi commis quelques erreurs d'inattention, relis-toi bien!
301496	0,5	[L1] [L22] [P5]	<p>WARNING: No line numbers, no syntax highlighting. Pages stapled in wrong order (3/4) if (boolean == true) looks redundant</p> <p>don't alter indentation if a line is too long, split it If you don't want to have an entire function inside an if-statement, just invert the condition and put a return.</p> <p>Please bear in mind your corrector as well as all the people that will potentially read your code are humans. They are a bit annoying, but it is easy to please them with colorful, airy and fluffy stuff.</p>
301548	1,0	[L22]126 [P5]101,138	Le code est bien commenté :) Tu declares un MAX const mais ne t'en sers pas toujours :(
301560	2,0	[P5] 41, 54 [L1] 39, 40, 150 [L2] 116, 123, 130 [L22] 108	Attention aux conventions de nommage, pas de majuscules. Le plus gros problème est l'indentation. N'hésites pas à utiliser des retours à la ligne pour aérer ton code, et fais attention à l'indentation.
301578	1,5	[L22]176, 219, 227...	Le style est bon, bravo pour les constantes pertinentes. Par contre, l'alignement des lignes trop longues n'est pas bon, à revoir.
301581	1,5	[L22] 1821,182	Très bon code, très bien indenté mis à part l'endroit mentionné. Bonne utilisation des define. Bravo.
301595	0,5	[L1]151, 272, [L2]67, 70, 308, [P5]70, 106, 127...	Les noms des variables ne sont pas bien choisis, mais au niveau de la structure du code, c'est lisible.

301602	0,5	[L1] 356, [L14] 327, [P5] 225, 372, 432	Le code est d'une qualité rare. Le langage C++ est très bien maîtrisé, le code est presque parfait. Les commentaires sont utilisés de manière très adaptée, et sont d'une grande qualité explicative. Le code est bien optimisé, mais pour encore plus d'optimisation c'est possible de sortir les calculs de l'entête des boucles for afin de les calculer une seule fois plutôt que au début de chaque itération (par exemple <code>image_imported.image.size()</code> , ligne 277). Excellente initiative avec l'algorithme de filtrage qui utilise un <code>stable_sort</code> , cela joue beaucoup surtout pour ce programme. Bravo!
301607	2,0		Très bonne indentation et alignements. Veille à utiliser des noms de constantes plus parlants que <code>CHIFFRECONDITION</code> (warning). Attention aussi à ne pas TROP commenter.
301633	1,5	[L1] 81, 82, 148, 185	Le style est correct, les implémentations sont logiques. Ce serait une bonne idée de mettre un peu plus de commentaires.
301641	1,0	[L14]166, 167 [P5]132, 179	Code propre et lisible. noms de paramètres parfois peu descriptifs (L, C, etc..) Attention à l'indentation et aux magic numbers.
301653	0,5	[P5] 9, 20 [L2] 136 [L14] 223, 229, 231, 233	Bon effort pour le style et les commentaires, dommages pour les oublis de quelques conventions. Usage des constantes à clarifier.
301688	2,0	[P5] ? ? ? ?	Ton code est très bien indenté et ton style est très bien! Tu as tout de même eu des petits soucis avec les magic numbers comme le nombre de couleurs ( cela pourrait changer selon le format de ton image) et le nombre de voisins ( tu pourrais avoir envie de changer comment le filtrage se passe). Bravo en tout cas, très bon code.
301689	1,5	[P5] 123, 230, 240	As-tu eu des problèmes de formatage lors du print de ton code? Les for, return et certains types primitifs ne s'affichent pas et cela à des répercussions sur ton indentation. Sinon ton code est très bien, très bien commenté et j'ai pas grand chose à dire à part quelques magic numbers, comme le nombre de couleurs ou de voisins qui pourraient varier. Bravo!
301707	1,5	[P5]225,228,236	Code bien commenté et bien mis en page, agréable à lire :) Définir MAX et EPSILON etc comme constantes est bien mais pas suffisant, il aurait aussi fallu avoir des constantes pour le nombre de voisins et le nombre de voisins de la même couleur nécessaires pour "adopter" leur couleur, par exemple
301708	0,0	[L1]everywhere [L14]everywhere [L2]in function prototypes notably [L22]affectation&filtrage methods [P5]affectation,comparaison&lecture_parametre fonctions	On avait explicitement demandé, plusieurs fois, des numéros de ligne... L'indentation inconsistante rend le code peu lisible
301711	0,0		
301718	2,0		Très bon travail! Code propre et lisible, bon usage de <code>define</code> et les commentaires sont clairs.
301720	1,0	[L22] 229 [P5]102, 204	Code propre et lisible., bons noms de variables et de fonctions. Attention aux magic numbers et à l'indentation.
301750	1,0	[L1] 213 [P5] 48, 93, 118...	Très bon travail avec un code agréable à la lecture. Attention à l'étourderie l. 213. Aussi, il est important de définir tes magic numbers pour rendre ton code plus flexible ! Utilise des <code>#define</code> ou <code>constexpr</code> à cette fin.
301766	1,0	[L22] 96, 113, 169.. [P5] 39, 52, 158 ...	Code propre et lisible. Attention à bien aligner tes lignes trop longues pour les rendre plus compréhensibles et surtout il FAUT définir tes Magic Numbers au début de ton programme à l'aide de <code>#define</code> ou <code>constexpr</code> .

301786	0,5	[L1] 99, [L22] 106, 146, [P5] 116, 118, 215, 220, 250	Code concis et agréable à lire. Essaie d'utiliser plus de constantes globales pour t'éviter des magic numbers (vaelurs hardcodées) cela te permettrait de modifier ton code efficacement si besoin est. Ton code est bien indenté en général à quelques petites exceptions malheureusement, attention aux lignes trop longues qui, quand séparées en plusieurs lignes doivent s'aligner correctement.
301829	1,0	[L22]119,169[P5]170,179	Ce code est très bien présenté et il y a un réel effort dans sa qualité malgré un ou deux soucis ! Bravo !
301834	0,5	[L2]?, ?, ?, ?[P5]?, ?, ?, ? [L22]	Pense à numéroter tes lignes. En outre, tu as fait quelques fautes avec les conventions de programmation
301836	1,0	[L22] 28,31,32, 177, 228 [P5] 45	Bonne indentation, bravo. Attention aux magic number pour le nombre de composante de couleur et mauvais alignement quand une expression est écrite sur plusieurs lignes
301837	1,0	[P5] 23, 61, 99, 115,189,248 [L22] 121,150	Bonne indentation, cependant attention aux alignements sur plusieurs lignes, un mauvais alignement rend la lecture plus difficile- Attention aussi à l'usage de magic number par exemple pour le nombre de composantes de couleurs et epsilon
301844	0,0	[L14], [L1], [L11], [P5]	Il faudra faire un effort la prochaine fois sur le document que tu rends : il n'y a pas de numérotation des lignes, pas de couleurs... C'est très peu lisible. Sinon, plusieurs erreurs de conventions à travers le programme. Ton code contient des magic numbers. N'oublie pas de bien lire le document des conventions la prochaine fois ! Comme il n'y a pas de numérotation des lignes, je n'ai pas pu mettre les lignes où ces erreurs sont présentes...
301848	0,5	[L14]212, 227, [L22]26, 69, 132..., [P5]51, 139, 196...	Le style est presque bon, les éléments majeurs sont en place. Il manque les constantes, et l'attention aux détails/la relecture.
301850	1,0	[L22]22,26,60,229 [P5]33,126	Attention à correctment aligner tes lignes trop longues, et utilise plutôt des constexpr/define/etc plutôt que des valeurs en dur dans le code (p.ex. MAX pour 255)
301863	1,5	[P5] 69, 98, 160	Pour les noms, suivre une même convention sur tout le code (snake ou camel case). Les noms de variable sont complètement obscures. Attention aux lignes trop longues (ligne 178). Sinon, bonne présentation.
301873	1,0	[P5]174,256[L2]70	C'est un code très bien présenté et qui se lit facilement. Quelques petites erreurs de distractions sont toutefois présentes.
301881	1,0	[L14] 231 [P5] 16, 51, 98...	Nice code ! Try to define as much constants as you need to remove every Magic Number. Be careful of double indentation.
301900	1,5	[L22]56,59,78,100,280, 285...	code très clair, quelques oublis d'alignement. attention au choix de certains noms comme largeur et longueur pour une image: c'est ambigu.
301924	1,5	[P5] 51, 52, 57...	Really nice code (even the main..) ! However, try to remove every Magic Number by defining and using proper constants in the right place.
301927	2,0		Code sans failles, bravo! Main pourrait éventuellement être plus aéré (avec plus d'espaces), mais c'est surtout une question de goût. Les commentaires sont aussi très bien placés.
301940	0,0	[L1]122,345,353,...[L22] 322,326,470[L14],141,1 62,225,...[P5]166,178- 187,...	Il y a beaucoup de problèmes concernant les conventions de programmation. Mais en tout cas, ton code était facilement naviguable.
301949	1,5	[L14]330	Le code serait beaucoup plus compréhensible avec des commentaires (p.ex. l'enchaînement d'opérateurs ternaires) En général la mise en page est très bien et agréable à lire :)

301950	0,0	[L1]filtrage & attribution seuil methods [L11]all methods [L14]couleur_finale & calcul_seuil [L22]prototypes,main,valeur_entree,couleur_finale,seuillage,etc [P5]filtrage,nombre_couleur & couleurs	On avait explicitement demandé, plusieurs fois, des numéros de ligne... L'indentation inconsistante rend le code peu lisible, ce qui n'est pas arrangé par le manque total de commentaires
301956	1,5	[P5] 76, 96, 99, 121, 222	Les noms des typedef auraient pu être plus explicites ("image", "image_codee"), l'encodage de RGB aurait pu être plus clair que juste 3 indexes dans un tableau de taille 3, et il y a beaucoup de magic numbers. Mais dans l'ensemble le code est très agréable à lire, très bien présenté, commenté et structuré.
301959	2,0		Bonne initiative d'utiliser size_t. Sinon, rie à redire sur un code très propre. Veille peut-être à l'aérer de lignes blanches, cela le rendra plus facile à lire et débbugger.
301964	1,5	[P5] 89, 142, 143, 178	warning: alignment l. 196 The code is very clear, and it seems like there is not much to say ^^'
301965	0,0	[L14]69 ; [L22]92,142 ; [L11]141,278 ; [P5] 23, 31, 39...	Globalement insuffisant au niveau du style. L'indentation est en générale bonne mais de nombreuses erreurs subsistent. Attention également aux magic numbers, tu n'utilises aucune constante dans ton code.
301994	2,0		Le code est clair et bien indenté : c'est agréable à lire !
301998	1,0	[P5] 84, 95 [L14] 147-150	Attention aux magic numbers ! Bien revoir la définition de ces derniers pour pallier au problème. Sinon, le code est assez lisible et les conventions assez bien respectées, en dehors d'une boucle for à l'indentation erronée.
302017	2,0		Excellent code, excellente documentation : Parfait. C'est du travail de qualité.
302031	1,0	[L22]31,40 [P5]187	Propre et facile à lire. Certaines des valeurs auraient pu être définies comme constantes au début du code plutôt que d'être passées en paramètre. Les noms tout en majuscule sont par convention réservés aux constantes. Attention aussi aux magic numbers et à l'indentation.
302035	1,0	[L22]19, 49, 184..., [P5]111, 245, 338...	Bon style, il manque quelques constantes pour remplacer les magic numbers. La consigne [L22] est comprise vu les lignes 106, 197, donc une relecture de plus aurait été nécessaire. Modification des fonctions d'erreur, attention.
302038	1,0	[L2] 28, 154, 166, 174, 187 [P5] 88, 177	Fais attention aux conventions de nommage des variables (majuscules), et aux longueurs des lignes. Attention aussi à ne rien oublier de l'énoncé. Tes commentaires et le choix des noms de variables et de fonction aident bien à la compréhension et ton code est très du coup très clair et agréable à lire.

			Don't confuse symbols/global constants and variables.  Note : exit(0) is equivalent to exit(EXIT_SUCCESS)
302041	1,5	[P5] 67, 226	The code is relatively clear.
302072	0,0		
302076	1,5	[P5]107,403,408,414	Très bien fait ce code ! Il y a hélas quelques magic numbers.
302211	0,0	[L1] 274, 276, 278, [14] 138, 409, [L22]162, 201-203, 246, 440, [R2] 44-54,	J'ai de la peine à t'enlever des points pour ça mais dans le projet on vous a dit d'utiliser certains symboles pour définir des constantes globales que tu n'as pas utilisé, désolé. Cela me fait du mal de t'enlever autant de points car je vois que ce sont des petites fautes d'attention et que tu sais indenter ton code, mais t'en as quand même pas mal. Ton code est long alors la prochaine fois essaie d'utiliser des options d'auto formattage qui te sauveront la vie. Ton code est très bien en général et bien commenté, désolé.
302276	1,5	[P5] 12, 39, 88, 108, 121, ...	Code bien organisé avec de bons espacements. Notion de magic numbers à retravailler.
302282	0,5	[L2] 30, 32, 34, 52 [L11] 71 [P5] 83, 93	Attention : plusieurs soucis de conventions. D'abord, attention aux magic numbers (tu ne fais aucun define, constexpr, etc.) et aux retours à la ligne intempestifs. Enfin, ici et là, il y a des erreurs légères d'indentations qui sont pénalisées. Quand une condition if est trop longue et qu'un saut à la ligne est nécessaire, attention à bien aligner les paramètres qui suivent à ceux qui précèdent. Sinon, bon code !
302289	0,5	[L1]53, 78, 133... ; [L14]82 ; [P5]11, 72	Dans l'ensemble plutôt bien, mais sois plus rigoureux sur l'indentation. Aussi, attention aux magic numbers, utilise des constantes
302297	1,0	[L22] 178,213 [P5] 34, 44, 66...	Bonne indentation : le code est propre et agréable à lire. Attention aux Magic Numbers, ils doivent être définis en dehors des fonctions avec des constexpr ou des #define.
302327	1,0	[L1] 122 [P5] 405	Excellente présentation du code bravo ! Les commentaires sont aussi utiles, c'est une très bonne pratique.
302360	1,5	[L14] 400	Ton code est clair et bien indenté, si ce n'est que pour une malheureuse double indentation à la ligne 400. Dans tes prototypes de fonctions, attention à aligner tes paramètres de manière lisible.
302362	1,0	[L14]151-154, [P5]122,123,124, 151, 205, 261	Bien propre et structuré (merci pour les commentaires!), mais plus de constantes et de structures rendraient le code plus lisible et simple à éditer. Attention aussi aux magic numbers et à l'indentation.
302365	1,0	[P5] 47, 191 [L22] 191, 316, 318	Fais attention aux noms de variables/fonctions qui pourraient être plus clairs, et à l'indentation. Mis à part ça, le code est très lisible, c'est bien.
302376	1,5	[P5]67,110,123,246,250	Plutôt que de coder en dur des valeurs dans le code, mets les en constante <b>globale</b> (p.ex. avec constexpr ou #define) avec un nom qui décrit ce que ça représente : cela rend le code plus modularisable et plus clair ; Ici il faut fouiller dans le code des fonctions pour modifier les paramètres
302394	2,0		Très bonne mise en page Pense par contre à mettre des commentaires pour expliquer ton code
302416	1,0	[P5]99,263[L2]100,101, 104,105	Ce code est particulièrement agréable à lire. Attention aux lignes longues et aux magic numbers.
302508	1,5	[P5] 132, 142	Code bien présenté, manque un peu de commentaires. Très bonne utilisation des define, fais gaffe à la fonction show_img o?u tu ne passes pas par référence. Attention aux magic numbers, tels que le nombre de couleurs. Bon code en général.
302520	1,0	[L2]28, 162, [P5]56, 68, 177...	Très bon style, il manque juste les constantes pour remplacer les magic numbers. Tu profiterais aussi d'une relecture pour éviter le wrapping! Dommage de perdre des points pour ça!
302539			

302546	1,5	[P5] 80, 83, 108, 307, (317), 327, 342, 398, 402, ...	<p>Don't use comments to shed light on mysterious raw values. Use symbols (the way you did for EPSILON). It also makes it more robust to changes.</p> <p>You should also address the mysterious semicolon invasion in your code... They randomly appear at the end of function declaration, if statements, for loops...</p> <p>exit(0) is equivalent to exit(EXIT_SUCCESS): is this what you meant?</p> <p>if (boolean == true) looks redundant...</p> <p>Skipping some lines helps make things clearer. Without necessarily being longs, functions tend to get quite dense... cf. getThresholds</p> <p>"Blue" comments are theoretically related to documentation... which you might actually know, though I believe you felt it could also be... pretty ^^</p>
302551	0,5	[P5] 67, 229 [L1] 160, 186 [L2] 204, 211, 213	Attention au respect des conventions ! Les constantes définies au début doivent être utilisées dès que possible afin d'éviter les magic numbers.
302565	2,0		Le style est irréprochable, très soigné. Aucune remarque assez importante pour être reproduite ici.
302576	1,5	[P5] 125, 231	Très bonne indentation ! Le code est propre et agréable à lire. Seul bémol : quelques magic numbers.
302583	1,5	[L22] 149, 176, 258	Très bon code ! Attention à l'alignement des paramètres dans les lignes mentionnées : cela améliorerait la lisibilité. Excellente gestion des Magic Numbers.
302584	1,5	[P5] 63, 73, 131...	Très bon travail ! Le code est propre et agréable à lire. Attention tout de même à définir assez de constantes pour éliminer tous les Magic Numbers.
302659	2,0		Très bon code, bien indenté et facile à lire ! Juste un commentaire pour les magic numbers : certains nombres que tu mets sont un peu dans la "zone grise" des magic numbers, donc n'hésite pas à les remplacer les prochaines fois pour éviter tout soucis (ex: ligne 220, le 3 de ton analyseur).
302674	1,0	[P5]55,66,79,108,184,190,191,249,...[L22]55,66,143,...	Il y a quelques divergences avec les conventions de programmation mais ton code est tout de même bien présenté. Bravo.
302680	1,0	[L11]250[P5]323-330	Ton code est très bien présenté. Toutefois quelques magic number et un oubli d'indentation subsiste.
302699	1,5	[P5]86, 102, 130...	Très bon style, il manque juste les constantes pour remplacer les magic numbers. Modification d'une fonction d'erreur, attention.
302703	1,5	[P5]97, 120, 244...	Très bon style, il manque juste quelques constantes pour remplacer les magic numbers, et quelques noms de variables plus pertinents.



302747	1,5	[P5] 90, 124, 156, 223, 261	Bonne indentarion. Attention aux magic umbers. En effet, tu utilises -1 pour erreur mais imaginons tu veux avoir plusieurs types d'erreur alors -1 devra avoir un nom pour que l'erreur soit explicite (p.e. invalid argument et disons -5 ce pourrait etre un invalid file type). Dans la même réfexion, nous avons ici 3 couleurs mais nous pouvions utiliser des rgba ( avec l'alpha) ou juste utiliser des images gray scale et donc ce 3 changerait. Certaines fonctions ne passent pas les tableaux par référence et cela pourrait nuire à la performance de ton programme Sinon le code est commenté et lisible, bravo.
302759	0,5	[L1] 187, 227, [L22] 130, 219, 220, 221, [P5] 125, 127, 197	Implémentations concises et logiques, très bien faites. Bonne utilisation des commentaires. Il aurait été possible de définir des struct pour simplifier le code, par exemple un struct pour les couleurs qui stocke les trois composantes RVB.
302829	0,5	[L1] partout [L11] partout [P5] 107, 121, 223,	Ton code est un cauchemar à lire car il n'est pas indenté. Je sais pas si tu as eu des problèmes de format à la fin en printant mais si c'est pas le cas je te conseillerais de regarder comment un code doit être indenté avec les bonnes conventions. C'est très difficile pour moi de lire ton code et de comprendre ce qui se passe. Il manque des variables globales constantes ce qui fait que tu as eu des problèmes pour les magic numbers. Ceci est important car si tu veux changer quelque chose dans ton code tu as besoin de changer toutes les valeurs partout au lieu de changer dans un seul endroit. Sinon ton code a des commentaires qui permettent de bien se repérer.
302833	1,5	[P5]74, 113, 192...	Très bon style, il manque juste quelques constantes pour remplacer les magic numbers.
302875	0,5	[L11] 42 [P5] 83, 87, 212 [L1] 260	Attention au respect des conventions ! Le code pourrait être plus espacé pour améliorer la lisibilité.
302888	1,0	[P5] 86,99,100,201,218,232 [L22] 273	Très bonne indentation bravo. Cependant usage de magic number à plusieurs répétition ainsi qu'un mauvais alignement d'une epression sur plusieurs lignes
302896	1,5	[L11] 104, 173, 227, 238, 274	Attention, la dernière ligne d'une fonction fait partie de la fonction et doit être indentée en conséquence. Sinon, le style est trs propre et lisible. Pour la prochaine fois, imprime en couleur, svp.
302909	2,0		Très bonne mise en page et utilisation des defines/structs/typedef pense à mettre des commentaires pour expliquer ton code
302910	0,5	[L1]136-147,182 [L22]76,210 [P5]135	Attention à bien garder une indentation uniforme Pense aussi à mettre des commentaires pour expliquer ton code
302917	1,5	[P5] 120, 131, 133...	Le code est bien indenté et semble respectetr els conventions à la lettre. Attention tout de même à la gestion des Magic numbers en définissant plus de constantes en début de programme (avec #define ou constexpr).
302953	1,0	[L1]326, 336 ; [P5]88, 109, 118...	Très bon travail, les conventions sont dans l'ensemble bien respectées ! Attention cependant aux magic numbers, utilise plus de constantes

303000	1,0	[L22] 167 [P5] 16, 26, 43 ...	Code propre et bien documenté ! Poursuis tes efforts en définissant les constantes de ton problèmes (Magic Numbers) au début de ton code avec des #define ou constexpr.
303003	1,5	[L1]65-69, 83, 85, 117-125,249	Fais très attention à ton indentation, un cin ne justifie pas un décalage. Sinon, bon usage de #define.
303011	0,0	[L1]59,72,82-97,128,166-173 [L11]33-219 [L2]27,133 [P5]18,69, 82,160,182	Il faut indenter le corps des fonctions, ainsi que garder l'indentation consistante au sein du même bloc Plutôt que de coder en dur des valeurs dans le code, mets les en constante globale (p.ex. avec constexpr ou #define) avec un nom qui décrit ce que ça représente : cela rend le code plus modularisable et plus clair De plus, pense à commenter ton code
303028	2,0		Excellent code, agréable à la lecture. Les commentaires sont particulièrement pertinents. Continue !
303039	1,0	[L22] 45, 128 , 217 [P5] 132, 155, 208 ...	Code propre et bien lisible ! Attention à bien aligner tes lignes trop longues pour les rendre plus compréhensibles. Aussi, n'hésite pas à définir autant de constantes qu'il en faut pour tes Magic Numbers : MAX n'est pas le seul paramètre du problèmes (EPSILON aussi par exemple).
303084	0,0	[L1] 108, 121, 175, 237, 271 [L14] 336, 343 [P5] 107, 120, 174 [R2] 57	Attention à l'indentation. La struct Info cache ce que chaque fonction prend comme arguments et rend donc le code plus confus. Sinon c'est très aéré, ce qui est agréable.
303114	1,5	[P5] 58, 258, 270, 277	warning [L2] 235 / ~OK 302 warning indentation 292, 296  (one question remains: how did you get away with an accented character in the program? 233 )
303153	1,5	[L14]131-132	Parfait sauf à un oubli !
303778	0,0	[P5] 50 [L11] 40-45 [L1] 96 [L2] 146, 194, 210 [L14] 175, 190, 191	Plusieurs soucis de conventions : tout d'abord, fais bien attention à ce que tes indentations soient cohérentes. Tu en fais soit trop, soit pas assez. Chaque bloc d'instructions devrait être indenté : ça améliore beaucoup la lisibilité (ex: ta fonction main vs tes fonctions d'input). Attention aussi aux magic numbers : tu ne fais aucun DEFINE/enum/constexpr, ce qui est très utile pour éviter les magic numbers.
304034	0,5	[L2] 17, 18, 133... [L14] 122.. [P5] 12, 31, 82...	Très bon code mais il faut être plus rigoureux avec les conventions. Attention au wrapping de tes lignes : respecte la ligne verte de geany ou vérifie les petites flèches après impression. Attention aussi aux Magic Numbers : tu n'as défini aucune constante dans ton code.
304148	2,0		
304289			