

C++ Bug Check List sem1 mise à jour : 28 septembre 2018, Ronan Boulic

	<i>Symptômes</i>	<i>Détection de la cause</i>	<i>Cause(s)</i>	<i>Comment l'éviter</i>
AVANT TOUT	Le bug est toujours là	Mes modifications ne changent rien	J'édite le mauvais fichier source	Ouvrir une seule fois le fichier dans l'éditeur
Règles de base : Compilation Edition de liens	Beaucoup d'erreurs signalées par le compilateur	Se concentrer sur la PREMIERE erreur, puis recompiler	Une seule vraie erreur peut induire une cascade d'erreurs secondaires	Compiler fréquemment, ne pas attendre d'avoir écrit tout le code source
	Aucune erreur signalée mais exécution incorrecte	Ajouter l'option de compilation <code>-Wall</code>	Un warning est une erreur qui s'ignore	Ne pas ignorer les warning
	Erreur signalée sur le mot clef <code>constexpr</code>	Ce mot clef a été introduit avec C++11	Le compilateur utilise une version plus ancienne du C++	Ajouter l'option de compilation <code>-std=c++11</code>
	Je ne trouve pas l'erreur dans la ligne signalée	Lire la ligne caractère par caractère	Faute de frappe	Options de l'éditeur : Couleur, taille, etc..
	Pas d'erreur dans la ligne signalée	Chercher dans lignes précédentes, y compris dans les fichiers inclus	Oubli de ; ou de parenthèse, ou d'accolade,...	INDENTATION automatique
	Erreur sur appel de fonction	Vérifier les fichiers en-têtes, déclarations, définitions, appels	Incohérence entre fichiers en-têtes, déclarations, définitions, appels	Le prototype des fonctions doit être décidé au début puis doit rester stable
	Erreur avant *	Juste après un type défini par l'utilisateur	pas un type de base	Oubli du typedef du type utilisateur
	<code>Undefined symbol</code> <code>Undefined reference</code>	Edition de liens	Faute de frappe d'un nom:variable, fonction Oubli, inversion ou corruption de .o ? lib ?	Code lisible Fichier Makefile complet Vérifier l'ordre des fichiers .o et libs
	redefinition	Edition de liens	Inclusion du .cc au lieu du .h	Code lisible
	Règles de base : Exécution	Je ne sais pas où est le bug	Utiliser ddd (trace) Commenter le code jusqu'à disparition du bug + Jeu de tests	Pour toutes les causes
le bug continue à se manifester		Faire une seule modification du code à la fois : compiler et refaire les mêmes tests	Pour toutes les causes	Ne pas oublier de forcer l'affichage quand on ajoute des cout de test
Le résultat affiché est-il correct ?		(Redirection des entrées et des sorties) sur jeu de tests, comparer les sorties avec l'éditeur	copier-coller Mauvaise expression Pb Mémoire, div. par 0	Vérifier les instructions d'affichage Empêcher les div par 0 Connaitre le langage C++ Cf ci-dessous
Segmentation fault Core dumped		Jeu de tests ddd :trace, breakpoint.. Commenter le code pour localiser le bug	Mauvais accès mémoire : pointeur ou débordement de tableau Allocation dynamique Cascade d'appels	Initialiser et tester les pointeurs avant usage Tester les bornes des indices de tableaux Faire un dessin...
Fonctionnement incorrect		Jeu de tests couvrant toutes les familles de cas possibles	Analyse incomplète Expressions logiques? = au lieu de == = ! au lieu de !=	Phase d'analyse Éviter les expressions logiques complexes Faire des if imbriqués
Aussi très utile	<p><u>Verbaliser</u> : décrire oralement les symptômes du problème à quelqu'un d'autre, même si cette personne ne connaît pas le problème ou le langage utilisé. Au besoin faire semblant de téléphoner. Formuler des hypothèses avec des phrases du genre : « c'est comme si... ».</p> <p><u>Faire un dessin</u> : pour tous les cas de segmentation fault impliquant des pointeurs</p> <p><u>Y réfléchir à deux fois</u> : après tout peut être que l'exécution observée correspond à ce qui était demandé ; il est important de bien connaître ses scénarios de tests</p> <p><u>Un regard neuf</u> : demander à quelqu'un d'autre de regarder le code ; après 10 minutes sur un morceau de code, on devient facilement aveugle à ses propres fautes de frappes.</p> <p><u>Noter les cas de bugs difficiles dans un carnet (symptômes, cause)</u> : et les relire de temps en temps</p>			