

Procédure d'installation cppcheck

Le compilateur de Microsoft Visual C++ détecte principalement des erreurs de syntaxe. Cppcheck s'ajoute à votre environnement de programmation (compilation) permettant alors de détecter des erreurs de programmation additionnelles.

Utilisation des avertissements et de Cppcheck :

Le but de ces outils est de vous aider à découvrir certaines erreurs dans votre programme.

En plus d'indiquer les erreurs de syntaxe, qui empêchent le programme de compiler, le compilateur indique aussi des avertissements. Par défaut, le niveau d'avertissement est à 3 (/W3). Pour ce travail dirigé, il est demandé qu'il soit à 4 (/W4). Ceci permet alors de détecter certains problèmes possibles, tel que l'utilisation d'une affectation au lieu d'une égalité dans une condition (comme dans « if (x = 4) » qui devait probablement être « if (x == 4) »), les variables potentiellement non initialisées et les conditions constantes. Pour voir la liste des erreurs et avertissements, **sélectionner le menu Affichage > Liste d'erreurs** et s'assurer de sélectionner l'onglet avertissements. Les problèmes détectés par « IntelliSense » sont mis à jour à mesure que vous entrez le programme, mais les erreurs et avertissements de compilation le sont uniquement lorsque le programme est recompilé (raccourci : F7).

Cppcheck fait d'autres analyses, permettant dans certains cas de détecter des accès hors bornes dans les tableaux, les énoncés vides, des améliorations possibles au style de programmation (réduction de la portée d'une variable; initialisation à la déclaration), etc. Un fichier .zip de Cppcheck vous est fourni. Vous devez, dans un premier temps, **décompresser les fichiers de cppcheck.zip dans votre répertoire « Documents\Visual Studio 2010 », en y créant le répertoire « cppcheck »**. Pour exécuter automatiquement Cppcheck à partir de l'environnement Visual Studio, et que les avertissements soient affichés avec ceux de la compilation normale, on peut le configurer comme événement post-build.

Configuration des avertissements et de Cppcheck (à faire pour chaque nouveau projet) :

Dans l'explorateur de solution, faire un click droit sur le nom du projet, puis sélectionner « Propriétés ».

Dans la fenêtre « Page de propriétés de <nom du projet> »

Aller dans Propriétés de configuration > C/C++.

Changer le « Niveau d'avertissement » pour « Level4 (/W4) ».

Aller dans Propriétés de configuration > Événements de build > Événement post-build. (Choisir OUI)

Entrer dans le champ « Ligne de commande » (les guillemets doivent être conservés) :

"\$(VisualStudioDir)\cppcheck\docheck.bat" "\$(ProjectPath)"

Puis faire « OK ».

Après cette configuration, Cppcheck sera automatiquement exécuté après chaque « build » (raccourci : F7), si le programme ne contient pas d'erreur de compilation. Les résultats de l'analyse seront dans la fenêtre « Liste d'erreurs » avec les autres avertissements. Il suffit de double-cliquer sur l'avertissement pour aller à la ligne de code correspondante.

Votre programme ne devrait avoir aucun avertissement, ni par le compilateur, ni par Cppcheck. Pour tout avertissement restant (s'il y en a) vous devez ajouter un commentaire dans votre code, à l'endroit concerné, en précisant pourquoi l'avertissement peut être ignoré.

Résumé des actions à prendre :

1. Créer : Nouveau projet avec son fichier cpp
2. Dans le menu Affichage > Liste d'erreurs et s'assurer de sélectionner les avertissements.
3. Copier les fichiers décompressés de cppcheck dans y:\ Documents\Visual Studio 2010\cppcheck\
- 4a. Sélectionner le projet, PAS le fichier cpp, et avec bouton de droite de la souris, « Propriétés ».
- 4b. Dans la fenêtre « Page de propriétés de <nom du projet> »
- 4c. Aller dans Propriétés de configuration > C/C++.
- 4d. Changer le « Niveau d'avertissement » pour « Level4 (/W4) ».
- 5a. Aller dans Propriétés de configuration > Événements de build > Événement post-build. (Choisir OUI)

5b. Dans la même fenêtre, Entrer dans le champ « Ligne de commande » (les guillemets doivent être conservés) :

```
"$(VisualStudioDir)\cppcheck\docheck.bat" "$(ProjectPath)"
```

5c. Puis faire « OK ».

Cppcheck sera automatiquement exécuté après chaque « build » (raccourci : F7)

Vérification que cppcheck fonctionne correctement.

Test 1) Dans le programme principal (main) qui vous est fourni, au début par exemple, ajouter les instructions suivantes et compiler (F&, CTRL-ALT-F& pour générer ou mieux régénérer la solution):

```
int tab[10];
tab[10] = 99;
```

```
1>La génération a démarré 2013-11-21 17:45:34.
```

```
...
```

```
1>PostBuildEvent:
```

```
1> Checking test2.cpp...
```

```
...
```

```
1>test2.cpp(88): warning : (error) Array 'tab[10]' accessed at index 10, which is out of bounds.
```

```
...
```

Test 2) Revenir au programme principal (main), enlever les instructions suivantes (afin que le programme s'exécute sans problème) :

```
int tab[10];
tab[10] = 99;
```

et ajouter les instructions suivantes pour créer dynamiquement une variable réelle (float). Vous pouvez l'initialiser à 55, même si ce n'est pas requise) :

```
float * ptr;
ptr = new float;
```

Exécuter le débogger : Menu Débogger, démarrer le débogage

```
...
```

```
'test.exe' : Chargé 'C:\Windows\SysWOW64\msvcr100d.dll', Les symboles ont été chargés.
```

```
Detected memory leaks!
```

```
Dumping objects ->
```

```
...
```

```
Object dump complete.
```

```
Le programme '[1312] test.exe: Natif' s'est arrêté avec le code 0 (0x0).
```

Les messages "...out of bounds" et "Detected memory leaks!" confirme que cppcheck vérifie le code.

Commentaires TODO

Le code contient des commentaires « TODO » que Visual Studio reconnaît.

1. Activer l'option : dans le **menu Outils>Options...**, allez dans **Éditeur de texte > C/C++ > Mise en forme > Divers > Énumérer les tâches de commentaire**, pour mettre cette option à « True », puis OK.

Pour afficher la liste des TODO, allez dans le **menu Affichage, sous-menu Autres fenêtres, cliquez sur Liste des tâches**. Choisissez ensuite dans cette fenêtre d'utiliser les commentaires. Vous pouvez double-cliquer sur les « TODO » pour aller à l'endroit où il se trouve dans le code. Vous pouvez ajouter vos propres TODO en commentaire pendant que vous programmez, et les enlever lorsque la fonctionnalité est terminée.

Si requis, cliquer une première fois dans le programme et les TODO apparaissent dans les commentaires.