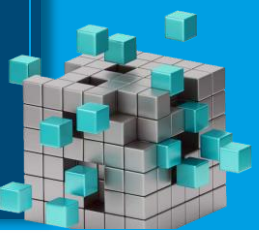
POLYTECH DIJON

TRAVAUX DIRIGÉS

Débogage



OBJECTIFS

- Acquérir des méthodes pour identifier et corriger des problèmes dans un code.
- Apprendre à utiliser les outils de débogage

> **Note**
> *Les outils présentés dans ce sujet sont ceux de Visual Studio Community 2022. Mais tout environnement de développement digne de ce nom propose des outils très similaires.*

> **Important**
> *Être en mesure de d'isoler et d'identifier l'origine d'un dysfonctionnement dans un programme est un savoir-faire essentiel du développeur. Si les LLM peuvent être des outils intéressants dans cette tâche, la maîtrise des techniques « manuelles » reste indispensable.*

I. PROBLEMES DE SYNTAXE

- Clonez ou téléchargez le dépôt https://github.com/cmeunier-ub/cpp-debogage-exercice-1.
- Si vous utilisez Visual Studio Community 2022, ouvrez la solution **`exercice1.sln`**
- Sinon, créez un projet avec votre environnement de développement et importer les fichiers **`.cpp`** et **`.h`** du projet exercice1
- Compilez le projet. Vous devriez voir apparaître quelques erreurs.
- Utilisez les indications du compilateur pour corriger les erreurs de syntaxe.

> **Astuce**
> *Commencer toujours par corriger les premières erreurs détectées par le compilateur. Souvent, les erreurs de syntaxe rendent le code difficilement compréhensible pour le compilateur qui détecte de nouveaux problèmes qui sont le résultat des erreurs précédentes.*

- Compilez de nouveau jusqu'à ce que plus aucune erreur ne soit détectée et que le programme s'exécute normalement.

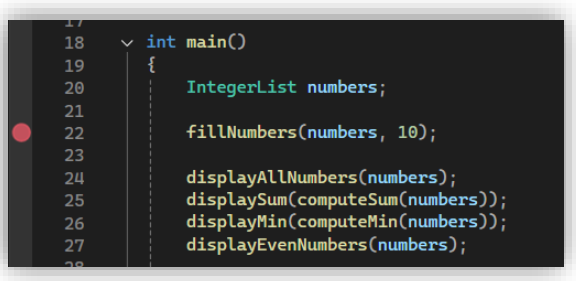
II. QUAND LA LOGIQUE S'EN MELE

- Clonez ou téléchargez le dépôt https://github.com/cmeunier-ub/cpp-debogage-exercice-2 et procédez comme précédemment pour ouvrir le projet.
- Compilez le projet. Tout se passe à merveille !
- Exécutez le programme. Oups ?

Exception

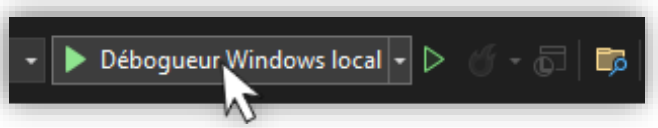
Cette fois-ci, le compilateur n'a pas détecté d'anomalie dans le code. C'est donc la logique du développeur qui semble être la cause de l'exception qui se lève lors de l'exécution du programme.

- Placez un point d'arrêt à la ligne 22 du fichier main.cpp

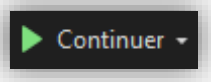


> **Note**
> *Pour placer un point d'arrêt, cliquez dans la « goutière » grise à gauche des numéros de ligne. Un rond rouge apparaît signalant que le programme se mettre en pause juste avant l'exécution de cette ligne de code.*

- Exécutez le programme avec le débogueur par défaut



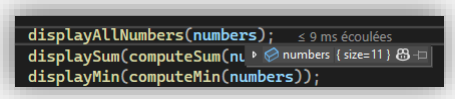
Le débogueur vous permet d'avancer à votre rythme au sein de votre programme grâce aux actions suivantes :

	**Continuer** Poursuit l'exécution du programme jusqu'au prochain point d'arrêt ou à la fin du programme.

	**Pas à pas** Poursuit l'exécution du programme jusqu'à la prochaine instruction de la fonction courante
	**Pas à pas détaillé** Poursuit l'exécution du programme jusqu'à la prochaine instruction de manière détaillée : suit les appels de fonctions
	**Pas à pas sortant** Poursuit l'exécution du programme jusqu'à la prochaine instruction de la fonction appelante

- A l'aide du pas à pas, identifiez la fonction qui provoque une exception.
- Placez un second point d'arrêt au sein de cette fonction et exécutez de nouveau le programme en mode débogage.
- Progressez pas à pas jusqu'à identifier le problème puis corrigez-le.
- Recommencez jusqu'à ce qu'il n'y ait plus d'exception à l'exécution.

> **Astuce**
> *Les exceptions donnent également des indices précieux quant à la cause du problème.*

> **Astuce**
> *En survolant avec la souris une variable accessible dans le scope, vous pourrez en voir la valeur à l'instant T :*
>
> 
>

Résultats étranges

- Maintenant que le programme s'exécute sans provoquer d'exception, étudiez les résultats affichés.
- Utilisez les méthodes de débogage pour identifier l'origine de chaque erreur et appliquer le correctif nécessaire.

III. CHANGEMENT DE TAILLE

- Clonez ou téléchargez le dépôt https://github.com/cmeunier-ub/cpp-debogage-exercice-3 et procédez comme précédemment pour ouvrir le projet.
- Compilez et exécutez le programme.

Vous voici dans la peau du développeur qui reprend le code de quelqu'un d'autre qui n'a vraisemblablement laisser aucune documentation. Ni UML, ni commentaire. Pourtant, il va falloir que vous parveniez à apporter quelques modifications.

Bon courage ! :-)

Menu principal

Le menu principal semble avoir des problèmes d'affichage si on appuie un peu trop sur les touches UP et DOWN.

- Identifiez où se situe le problème dans le code et corrigez le.

Baisser la difficulté

Les utilisateurs se plaignent d'une difficulté trop importante.

- Réduisez le nombre de couleurs composant la combinaison de 8 à 4.
- Augmentez le nombre de propositions possibles de 5 à 10.

Analyse des combinaisons

Il semble que l'analyse des combinaisons ne soit pas terminée.

- Ajoutez l'algorithme permettant de déterminer le nombre de bonnes couleurs mal positionnées.