

(https://profile.intra.42.fr)

Remember that the quality of the defenses, hence the quality of the of the school on the labor market depends on you. The remote defences during the Covid crisis allows more flexibility so you can progress into your curriculum, but also brings more risks of cheat, injustice, laziness, that will harm everyone's skills development. We do count on your maturity and wisdom during these remote defenses for the benefits of the entire community.

SCALE FOR PROJECT CPP MODULE 07 (/PROJECTS/CPP-MODULE-07)

You should evaluate 1 student in this team



Git repository

git@vogosphere-v2.42.fr:vogosphere/intra-uuid-9811e3c9-6639-4



Introduction

Veuillez respecter les règles suivantes:

- Restez poli, courtois, respectueux et constructif pendant le processus d'évaluation. Le bien-être de la communauté repose là-dessus.
- Identifiez avec la personne (ou le groupe) évalué les éventuelles dysfonctionnements du travail. Prenez le temps d'en discuter et débitez des problèmes identifiés.
- Vous devez prendre en compte qu'il peut y avoir de légères différences d'interprétation entre les instructions du projet, son scope et ses fonctionnalités. Gardez un esprit ouvert et notez de la manière la plus honnête possible. La pédagogie n'est valide que si la peer-évaluation est faite sérieusement.

Disclaimer

- Ne notez que ce qui est contenu dans le dossier git cloné de l'étudiant ou du groupe.
- Vérifiez que le dossier git de appartient bien à l'étudiant ou au groupe, que le projet correspond bien, et que "git clone" est utilisé sur un dossier vide.
- Vérifiez scrupuleusement qu'aucun alias n'a été utilisé pour vous tromper et assurez

vous que vous corrigez bien le rendu officiel.

- Afin d'éviter toute surprise, vérifiez avec l'évalué les potentiels scripts utilisés pour faciliter l'évaluation.

- Si, en temps qu'évaluateur, vous n'avez pas fait le projet courant, vous devez lire le sujet en entier avant de commencer l'évaluation.

- Utilisez les flags disponibles pour signaler un rendu vide, un programme ne fonctionnant pas, une erreur de norme, un problème de triche... Dans ces situations, l'évaluation est terminée et la note est 0 (ou -42 en cas de triche). Cependant, à l'exception des cas de triche, vous êtes encouragés à continuer la discussion sur le travail rendu (même si pas terminé) pour identifier les problèmes qui ont causé cet échec et éviter de les reproduire dans le futur.

- Pendant toute la durée de l'évaluation, aucun segfault ou autre arrêt du programme (inattendu, prémature ou incontrôlé) ne sera toléré. Dans ces cas-là, la note finale est 0. Utilisez le flag approprié.

Vous ne devriez jamais avoir à éditer un fichier mis à part un fichier de configuration, si celui-ci existe. Dans le cas où vous souhaitez modifier un fichier, vous devez expliciter clairement les raisons de l'édition et entre en accord avec l'étudiant évalué avant de faire quoi que ce soit.

- Il vous revient également de vérifier l'absence de fuite mémoire. Toute mémoire allouée sur la heap doit être proprement libérée avant la fin de l'exécution.

Vous pouvez pour cela utiliser les différents outils présent sur l'ordinateur, tels que leaks, valgrind, ou encore e_fence. En cas de fuite mémoire, cochez le flag approprié.

Guidelines

Vous devez compiler avec clang++, with -Wall -Wextra -Werror

Nous vous rapellons que ce projet est en C++98, et toute fonction/container des versions suivantes n'est pas attendue, et leur utilisation se soldera par un 0.

Ces points signifient que vous ne devez pas noter l'exercice demandé :

- Une fonction est implémentée dans un header (sauf pour les templates)
- Un Makefile compile sans les flags et/ou avec autre chose que clang++

Ces point signifient que vous devez tag le projet en "Fonction Interdite" :

- Utilisation d'une fonction "C" (*alloc, *printf, free)
- Utilisation d'une fonction interdite dans le projet
- Utilisation de "using namespace" ou de "friend"
- Utilisation d'une lib externe, ou des features des versions postérieures a C++98

Attachments

Exercice 00: Quelques fonctions

Dans cet exercice, l'étudiant doit écrire 3 fonction template simples : swap, min and max.

Types simples

Reférez vous au sujet pour l'output attendu avec des types simples, tels que int.

☒ Yes

☐ No

Complex types

Est-ce que les fonctions fonctionnent également avec des types complexes tels que :

```
class
Awesome {
public:
Awesome( int n ) : _n( n ) { }
bool operator==( Awesome const & rhs ) { return (this->_n == rhs._n); }
bool operator!=( Awesome const & rhs ) { return (this->_n != rhs._n); }
bool operator>( Awesome const & rhs ) { return (this->_n > rhs._n); }
bool operator<( Awesome const & rhs ) { return (this->_n < rhs._n); }
bool operator>=( Awesome const & rhs ) { return (this->_n >= rhs._n); }
bool operator<=( Awesome const & rhs ) { return (this->_n <= rhs._n); }
private:
int _n;
};
?
```

☒ Yes

☐ No

Exercice 01: Iter

L'objectif de cet exercice est d'écrire une fonction générique d'itération dans un array.

Ca marche ???

Testez le code suivant avec le rendu de l'étudiant :

```
class Awesome
{
public:
Awesome( void ) : _n( 42 ) { return; }
int get( void ) const { return this->_n; }
private:
```

```
int _n;
};
std::ostream & operator<<( std::ostream & o, Awesome const & rhs ) { o << rhs.get(); return o; }
template< typename T >
void print( T const & x ) { std::cout << x << std::endl; return; }
int main() {
int tab[] = { 0, 1, 2, 3, 4 }; // <--- J'ai jamais compris pourquoi on peut pas ecrire int[] tab. Ca aurait plus de sens
vous trouvez pas ?
Awesome tab2[5];

iter( tab, 5, print );
iter( tab2, 5, print );
```

```
return 0;
}
```

Si tout fonctionne correctement, ca devrait afficher :

```
0
1
2
3
4
42
42
42
42
42
```

☒ Yes

☐ No

Exercise 02: Array

Dans cet exercice, l'étudiant doit écrire un template de classe qui se comporte comme un array. Si l'allocation interne de l'array ne provient pas d'un new[], ne notez pas cet exercice. Demandez à l'étudiant de prouver que son template marche avec des types simples et complexes avant de noter.

Constructors

Est-il possible de créer un array vide et un array de taille spécifique.

☒ Yes

☐ No

Accès

Les éléments doivent être accessible en lecture et écriture via l'opérateur[]
(ou lecture uniquement si l'instance est const). L'accès à un élément hors-limites
doit renvoyer une std::exception.

✓ Yes

✗ No

Ratings

Don't forget to check the flag corresponding to the defense

✓ Ok

★ Outstanding project

📄 Empty work

💬 No author file

⚙️ Invalid compilation

📖 Norme

📑 Cheat

💥 Crash

💧 Leaks

🚫 Forbidden function

Conclusion

Leave a comment on this evaluation

Finish evaluation

General term of use of the site (<https://signin.intra.42.fr/legal/terms/6>)

Privacy policy (<https://signin.intra.42.fr/legal/terms/5>)

Legal notices (<https://signin.intra.42.fr/legal/terms/3>)

Declaration on the use of cookies (<https://signin.intra.42.fr/legal/terms/2>)

Rules of procedure (<https://signin.intra.42.fr/legal/terms/4>)

Terms of use for video surveillance (<https://signin.intra.42.fr/legal/terms/1>)