

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

бешень

ШКАЛА ДЛЯ ПРОЕКТА CPP MODULE 08 (/ПРОЕКТЫ/CPP-МОДУЛЬ-08)

Вы должны быть 1 студентом в этой команде

Git-репозиторий

git@vogsphere.msk.21-school.ru:vogsphere/intra-uuid-318c3ae

Введение

Методические рекомендации

для облегчения выявления ошибок (скачать для тестирования или автоматизации).

- Если вы не выполнили задание, которое собирает ошибки, вы должны прочитать всю тему до начала процесса оценивания.

- Используйте доступные флаги, чтобы сообщить о пустом репозитории, неработающей программе, ошибке Norm, чистоте итак далее.

В этих случаях процесс оценки завершается и окончательная оценка равна 0 или -42 в случае списывания. Однако, за исключением списывания, учащимся настоятельно рекомендуется вместе просмотреть данную работу, чтобы выявить ошибки, которые не следует повторять в будущем.

- Вам никогда не придется редактировать какой-либо файл, кроме файла конфигурации, если он существует. Если вы хотите отредактировать файл, найдите время, чтобы убедиться, что причина оцениваемого учащегося убедительна, что выбора согласны с этим.

- Также необходимо проверить отсутствие утечек памяти. Любая память, выделенная в куче, должна быть должным образом освобождена до окончания выполнения.

Вам разрешено использовать любой из различных инструментов, доступных на компьютере, так как как ut еч к и, valgrind или е_fence. В случае утечек памяти отметить соответствующий флаг.

Вложения

subject.pdf (<https://cdn.intra.42.fr/pdf/pdf/41305/en.subject.pdf>)

Предварительные испытания

Если подается мошенничество, оценка здесь останавливается. Используйте флаг «Обман», чтобы сообщить об этом. Принимайте это решение спокойно, взвешенно и, пожалуйста, используйте эту кнопку с осторожностью.

Предпосылки

Код должен компилироваться с помощью C++ и флагов -Wall -Wextra -Werror. Не забывайте, что этот проект должен соответствовать стандарту C++98. Таким образом, C++11 (и более поздних версий) НЕ ожидается.

Целью этого модуля является использование STL. Затем с помощью контейнеров и алгоритмов авторизуется.

Любое из этих значений означает, что вы должны оценивать рассматриваемое упражнение: - Функция реализована в заголовочном файле (кроме шаблонных функций).

- Makefile компилируется без необходимых флагов и/или другого компилятора, кроме c++.

Любое из этих действий означает, что вы должны отметить проект как «Запрещено».

Функция ":

- Использование функций "C" (*alloc, *printf, бесплатно).
- Использование функций, не разрешенной в инструкции по упражнению.
- Использование "using namespace" или ключевое слово "друг".
- Использование внешней библиотек или функций из версий, отличных от C++98.

Да

Нет

Ex00: Легко найти

Как обычно, должна быть основная функция, которая содержит достаточные тесты, чтобы доказать, что программа работает должным образом. Если нет, не оценивайте это упражнение. Если какой-либо класс, не связанный с интерфейсом, не соответствует оргодоксальной анонимной форме класса, не оценивайте это упражнение.

Функция шаблона

Существует шаблонная функция easyFind(T, int), которая делает то, что требует субъект.

Он ДОЛЖЕН использовать алгоритм STL.

Если это не так (например, ручной поиск с использованием итераторов), считайте это неправильным.

Да

Нет

Ex01: Диапазон

Как обычно, должна быть основная функция, которая содержит достаточные тесты, чтобы доказать, что программа работает должным образом. Если нет, не оценивайте это упражнение. Если какой-либо класс, не связанный с интерфейсом, не соответствует оргодоксальной анонимной форме класса, не оценивайте это упражнение.

Класс и функции-члены

Есть класс, который соответствует ограничениям предмета.

Его функции-члены используют алгоритм STL, чтобы максимально точно найти результат. Найти кратчайший отрезок невозможно, только выиграв два младших числа (присмотреться к рассматриваемому примеру).

Да

Нет

Улучшенная функция addNumber

Есть более практичный способ добавления чисел, чем повторный вызов функции addNumber().

Да

Нет

Ex02: Мутировавшая мерзость

Как обычно, должна быть основная функция, которая содержит достаточное количество кода, чтобы программа работала должным образом. Если нет, не оценивайте это упражнение. Если как-либо класс, не связанный с интерфейсом, не соответствует требованиям оригинальной анонимной формы класса, не оценивайте это упражнение.

Класс MutantStack

Существует класс MutantStack, который наследуется от std::stack и предлагает все его функции.

У него есть итератор. Кроме того, как минимум операции, показанные в примерах темы, можно выполнять с помощью итераторов.

Да Нет

Лучшие тесты

Существует по крайней мере функция main(), которая имеет больше тестов, чем тестовых тем.

Да Нет

Рейтинги

Не забудьте проверить флаг, соответствующий защите

Хорошо Выдающийся проект

Пустая работа Нет файла автора Недоступная компиляция Norme Cheat d Crash

Утечки л Запрещенная функция

Заключение

Оставьте комментарий к этой оценке

Завершить оценку

Политика конфиденциальности (<https://signin.intra.42.fr/legal/terms/5>)

Условия использования видеонаблюдения (<https://signin.intra.42.fr/legal/terms/1>)

Правила процедуры (<https://signin.intra.42.fr/legal/terms/4>)

Декларация об использовании файлов cookie (<https://signin.intra.42.fr/legal/terms/2>)

Общие условия использования сайта (<https://signin.intra.42.fr/legal/terms/6>)

Правовые уведомления (<https://signin.intra.42.fr/legal/terms/3>)