

# ШКАЛА ДЛЯ ПРОЕКТА CPP MODULE 05 (/ПРОЕКТЫ/СРР-МОДУЛЬ-05)

Вы должны оценить 1 студента в этой команде

Репозиторий Git

## Введение

Пожалуйста, соблюдайте следующие правила:

- Оставайтесь вежливыми, вежливыми, уважительными и конструктивными на протяжении всего процесса оценки. От этого зависит благополучие общества.
- Определите вместе со студентом или группой, чья работа оценивается, возможные дисфункции в их проекте. Потратьте время на обсуждение и обсуждение проблем, которые, возможно, были выявлены.
- Вы должны учитывать, что могут быть некоторые различия в том, как ваши коллеги могли бы понять инструкции проекта и объем его функциональных возможностей. Всегда сохраняйте непредвзятость и оценивайте их как можно более честно. Педагогика полезна только и только в том случае, если экспертная оценка проводится серьезно.

## Инструкция

- Оценивайте только ту работу, которая была включена в репозиторий Git оцениваемого студента или группы.
- Дважды проверьте, что репозиторий Git принадлежит студенту(ученикам). Убедитесь, что проект соответствует ожиданиям. Кроме того, убедитесь, что "git clone" используется в пустой папке.
- Тщательно проверьте, чтобы никакие вредоносные псевдонимы не использовались для того, чтобы обмануть вас и заставить оценить то, что не является содержимым официального репозитория.
- Чтобы избежать каких-либо неожиданностей и, если это применимо, просмотрите вместе все используемые сценарии

¶

□

`git@vogosphere.msk.21-school.ru:vogosphere/intra-uuid-193aaa9`  
(<https://profile.intra.42.fr>)

□

¶ (<https://profile.intra.42.fr/searches>)

inightin

для облегчения оценивания (скрипты для тестирования или автоматизации).

- Если вы еще не выполнили задание, которое собираетесь оценить, вы должны прочитать весь предмет до начала процесса оценки.
- Используйте доступные флаги, чтобы сообщить о пустом репозитории, нефункционирующей программе, ошибке нормы, мошенничестве и так далее.

В этих случаях процесс оценки заканчивается, и итоговая оценка составляет 0,

или -42 в случае обмана. Однако, за исключением списывания, студенту настоятельно рекомендуется вместе просмотреть сданную работу, чтобы выявить любые ошибки, которые не должны повторяться в будущем.

- Вам никогда не придется редактировать какой-либо файл, кроме файла конфигурации, если он существует. Если вы хотите отредактировать файл, потратьте время на то, чтобы объяснить причины оцениваемому студенту и убедиться, что вы оба согласны с этим.

- Вы также должны проверить отсутствие утечек памяти. Любая память, выделенная в куче, должна быть должным образом освобождена до конца выполнения.

Вы можете использовать любой из различных инструментов, доступных на компьютере, таких как leaks, valgrind или e\_fence. В случае утечки памяти установите соответствующий флаг.

# Вложения

О тема.pdf (<https://cdn.intra.42.fr/pdf/pdf/40215/en.subject.pdf>)

## Предварительные испытания

Если есть подозрение в мошенничестве, оценка на этом заканчивается. Используйте флаг "Чит", чтобы сообщить об этом. Примите это решение спокойно, мудро

и, пожалуйста, используйте эту кнопку с осторожностью.

Предпосылки

Код должен компилироваться с c++ и флагами-Wall-Wextra-Werror

Не забывайте, что этот проект должен соответствовать стандарту C++98. Таким образом, функции или контейнеры C++11 (и более поздних версий) НЕ ожидаются.

Любое из этих средств означает, что вы не должны оценивать рассматриваемое упражнение:

- Функция реализована в заголовочном файле (за исключением шаблонных функций).
- Makefile компилируется без требуемых флагов и/или другого компилятора, отличного от c++.

Любое из них означает, что вы должны пометить проект "Запрещенной функцией":

- Использование функции "C" (\*alloc, \*printf, free).
- Использование функции, не разрешенной в руководстве по упражнениям.

29.11.2012, 7:30 ВЕЧЕРА

Intra Projects CPP Module 05 Править

[https://projects.intra.42.fr/scale\\_teams/3863090/edit](https://projects.intra.42.fr/scale_teams/3863090/edit)

3/5

- Использование ключевого слова "using namespace" или "friend".
- Использование внешней библиотеки или функций из версий, отличных от C++98.

## Ex00: Мама, когда я вырасту, я хочу быть бюрократом!

Как обычно, должна быть основная функция, содержащая достаточно тестов, чтобы доказать, что программа работает должным образом. Если его нет, не оценивайте это упражнение. Если какой-либо неинтерфейсный класс не находится в ортодоксальной канонической форме класса, не оценивайте это упражнение.

Бюрократ

Существует Makefile, который компилируется с использованием соответствующих флагов.

У него есть постоянное имя.

Он имеет класс, который колеблется от 1 (самый высокий) до 150 (самый низкий).

Исключения возникают при попытке создать Бюрократа со слишком высокой или слишком низкой оценкой.

Существуют функции для увеличения / уменьшения оценки, они выдают исключения, когда это уместно. Помните, что увеличение 3-го класса даст вам 2-й класс (1-самый высокий).

Используемые исключения наследуются от `std::exception` или от чего-то производного от `std::exception` (т. Е. Они улавливаются как `std::exception & e`).

Существует оператор `<<` для перегрузки `ostream`, который выводит информацию бюрократа.

## Ex01: Стройтесь, личинки!

Как обычно, должна быть основная функция, содержащая достаточно тестов, чтобы доказать, что программа работает должным образом. Если его нет, не оценивайте это упражнение. Если какой-либо неинтерфейсный класс не находится в ортодоксальной канонической форме класса, не оценивайте это упражнение.

Форма

Существует Makefile, который компилируется с использованием соответствующих флагов.

class.

It имеет имя, bool, который указывает, подписан ли он (в начале это не так), класс, необходимый для его подписания, и класс, необходимый для его выполнения.

~~A~~

ДА

☐ Нет

☒

ДА

☐ Нет

29.11.2012, 7:30 ВЕЧЕРА

Intra Projects CPP Module 05 Править

[https://projects.intra.42.fr/scale\\_teams/3863090/edit](https://projects.intra.42.fr/scale_teams/3863090/edit)

4/5

Все эти атрибуты являются частными и не защищены.

Оценки форм следуют тем же ограничениям, что и у Бюрократа

(исключения, 1 = высший, 150 = низший и так далее).

Существуют средства доступа к атрибутам и оператор << для

перегрузки ostream, который отображает полное состояние формы.

Существует функция-член Form::beSigned (), которая работает так, как описано субъектом.

Существует функция бюрократа::signForm (), которая работает так, как описано субъектом.

## Ex02: Нет, вам нужна форма 28В, а не 28С...

Как обычно, должна быть основная функция, содержащая достаточно тестов, чтобы доказать, что программа работает должным образом. Если

его нет, не оценивайте это упражнение. Если какой-либо неинтерфейсный класс не находится в ортодоксальной канонической форме класса, не оценивайте

это упражнение.

Формы, которые действительно что-то делают

Существует Makefile, который компилируется с использованием соответствующих флагов.

есть конкретные формы, которые соответствуют спецификациям

предмета (требуемые оценки, имена и действия).

Они принимают только один параметр в своем конструкторе, который является их целью.

Существует метод Form::execute(Бюрократ const & executor), который работает так, как указано субъектом.

Либо этот метод является чистым и проверки оценок реализуются в каждом

подклассе, либо этот метод выполняет проверки, а затем вызывает другой метод

в производном классе, который выполняет только действие.

Оба эти метода действительны.

Существует Бюрократ::executeForm(Form const & form), который работает так, как указано субъектом.

## Ex03: По крайней мере, это лучше, чем приготовление кофе

Как обычно, должна быть основная функция, содержащая достаточно тестов, чтобы доказать, что программа работает должным образом. Если

его нет, не оценивайте это упражнение. Если какой-либо неинтерфейсный класс не находится в ортодоксальной канонической форме класса, не оценивайте

это упражнение.

Стажер

☒

ДА

☐ Нет



ДА

☐ Нет

29.11.2012, 7:30 ВЕЧЕРА

Intra Projects CPP Module 05 Править

[https://projects.intra.42.fr/scale\\_teams/3863090/edit](https://projects.intra.42.fr/scale_teams/3863090/edit)

5/5

Окончательная оценка

Существует Makefile, который компилируется с использованием соответствующих флагов.

class.

It имеет функцию makeForm (), которая работает так, как указано субъектом.

Хорошая диспетчеризация

Функция makeForm() должна использовать какой-то массив указателей на

функции-члены для обработки создания Форм.

Если он использует нечистый метод, например, ветви if/elseif/elseif/else, или

какие-то другие уродливые вещи вроде этого, пожалуйста, считайте это неправильным.



ДА

☐ Нет



ДА

☐ Нет

## Рейтинги

Не забудьте проверить флаг, соответствующий обороне

♣ Выдающийся проект

☐ Пустая работа

Ù Нет файла автора

W Недопустимая компиляция

♣ Norme

ÿ Обмануть

d Крах

☐ Утечки

l Запрещенная функция

## Заключение

Оставьте комментарий к этой оценке

Проверка офлайн

МСК. Пир рассказал

Политика конфиденциальности

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

Условия использования для 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 пользования сайтом

(<https://signin.intra.42.fr/legal/terms/6>)

Юридические уведомления

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

☐ ОК