

Программирование 1 курс

Общая информация

1. [Вводное занятие](#)
2. [Лекции -- осенний семестр](#)
3. [Лекции -- весенний семестр](#)
4. [<ссылка на рейтинг>](#)

Рейтинговая система

Перед началом выполнения практических заданий по дисциплине “программирование” студент должен ознакомиться с данными правилами и пройти тест. Студенты не прошедшие тест считаются неприступившими к выполнению практической части дисциплины“

Тест для **осеннего** семестра: <http://e.moevm.info/mod/quiz/view.php?id=760>

Тест для **весеннего** семестра: <http://e.moevm.info/course/view.php?id=18#section-1>

Оценка по дисциплине формируется по результатам работы студента в течении семестра на рейтинговой основе и складывается из выполнения следующих компонент (этапов):

- Выполнение и защита лабораторных работ,
- Написание контрольных работ,
- Прохождение онлайн-курса на <http://e.moevm.info/>,
- Посещаемость лекций,
- Прохождение онлайн-курса на по программированию (**для осеннего семестра**),
- Прохождение онлайн-курсов по Git и Linux (**для осеннего семестра**).

За выполнение каждой компоненты (этапа) студент получает определенное количество баллов, сумма которых определяет оценку за экзамен. На лабораторных работах студент либо защищает лабораторные работы, либо проходит задачи онлайн-курса.

Таблица 1. Количество заданий и баллы за их выполнение

№	Программирование 1 (осенний) семестр	Кол-во баллов
1	4 лабораторные работы ;	2 балла за каждую лабораторную работу.
	Создан pull-request соответствующий требованиям содержащий исходный код лабораторной работы (загружаемый код должен быть принят системой проверки онлайн-курса) и отчет	Суммарно 8 баллов за три лабораторные работы
2	Защита лабораторной работы (<i>защита возможна только после создания пулл реквеста – см. п. 1</i>)	{0,1,2,3} баллов каждая Максимум 12 баллов за три лабораторные работы
3	Контрольная работа	0-5 баллов

№ Программирование 1 (осенний) семестр		Кол-во баллов
4	Прохождение основного онлайн-курса в соответствии с дедлайнами для заданий курса (первые три модуля) (балл выставляется пропорционально степени прохождения, округление к меньшему целому, только ровно 100% дают 5 баллов. Задания непосредственно лабораторных не учитываются в этой метрике)	0-5 баллов
5	Онлайн-курс по Git	0-3 балла
6	Онлайн-курс по Linux	0-3 балла
№ Программирование 2 (весенний) семестр		Кол-во баллов
1	4 лабораторные работы	2 балла за каждый пулл реквест по лабораторной работе. Суммарно 8 баллов за 4 пулл реквеста по 4-м лабораторным работам
2	Защита лабораторной работы	{0,1,2,3} баллов каждая защита Максимум 12 баллов за защиты 4-х лабораторных работ
3	Контрольная работа	0-3 балла
4	Прохождение соответствующего онлайн-курса: http://e.moevm.info/enrol/index.php?id=18	0-5 баллов (балл выставляется пропорционально степени прохождения, округление к меньшему целому, только ровно 100% дают 5 баллов)

Оценка, которую студент может получить в течении семестра определяется суммой набранных баллов, указанных в Таблице 2.

Таблица 2. Формирование оценки

Оценка	Граница баллов осеннего семестра	Граница баллов весеннего семестра
Удовлетворительно	≥ 15	≥ 10
Хорошо	≥ 24	≥ 21
Отлично	≥ 31	≥ 25

Если студента не удовлетворяет оценка по итогам работы в семестре (на основании баллов), он может отказаться от нее и сдать экзамен. В таком случае, накопленная за семестр оценка теряется. Оценка за курсовую работу равняется оценке за ее защиту.

Оценка за курсовую работу равняется оценке за ее защиту.

Примеры расчета оценок для **осеннего** семестра:

Удовлетворительно:

- Созданы pull-request ко всем лабораторным работам (8 баллов) + полностью пройден основной онлайн-курс (5 баллов) + контрольная написана на 2 балла (2 балла) = 15 баллов

- Создан pull-request к одной лабораторной работе (2 балла) + частично пройден основной онлайн-курс (3 балла) + контрольная написана на 4 балла (4 балла) + полностью пройдены онлайн курсы по Git (3 балла) и Linux (3 балла) = 15 баллов

Хорошо:

- Созданы pull-request ко всем лабораторным работам (8 баллов) + две лабораторные работы защищены на 5 (6 баллов) + одна лабораторная защищена на 3 (1 балл) + написана контрольная на 3 балла (3 балла) + полностью пройдены онлайн курсы по Git (3 балла) и Linux (3 балла) = 24 балла

Отлично:

- Созданы pull-request ко всем лабораторным работам (8 баллов) + все лабораторные работы защищены на 4 (8 баллов) + полностью пройден основной онлайн-курс (5 баллов) + пройдены онлайн курсы по Git (3 балла) и Linux (3 балла) + написана контрольная на 4 (4 балла) = 31 баллов

Примеры расчета оценок для **весеннего** семестра:

Удовлетворительно:

- Подготовлены PR ко всем лабораторным работам (8 баллов) + Частично пройден онлайн-курс (3 балла) + написана контрольная работа на 2 балла (2 балла) = 13 баллов

Хорошо:

- Пройден полностью онлайн-курс (5 баллов) + Написана контрольная работа на 3 (3 балла) + Подготовлены PR ко всем лабораторным работам (8 баллов) + Одна лабораторная работа защищена на 1 балл (1 балл) + Две лабораторных работы защищены на 2 балла каждая (4 балла) + Одна лабораторная работа защищена на 3 балла (3 балла) = 24 балла

Отлично:

- Пройден полностью онлайн-курс (5 баллов) + Написана контрольная работа на 3 балла (3 балла) + Подготовлены PR ко всем лабораторным работам (8 баллов) + Две лабораторные работы защищены на 3 балла каждая (6 баллов) + Две лабораторные работы защищены на 2 балла каждая (4 балла) = 26 баллов

Структура курса и график работ

Таблица 3. Крайний срок выполнения работ

	Лаб. работа №1	Лаб. работа №2	Контрольная работа	Лаб. работа №3	Лаб. работа №4	Курсовая работа	Выполнение курса moodle и создание PR для лаб. работ
Создание корректного PR для допуска на защиту	1 семестр: до 23:59:59 МСК 28 сентября 2 семестр: до 23:59:59 МСК 5 марта	1 семестр: до 23:59:59 МСК 19 октября 2 семестр: до 23:59:59 МСК 9 апреля	1 семестр: 9 ноября , 2 семестр: ориентировочно в середине апреля	1 семестр: до 23:59:59 МСК 16 ноября 2 семестр: до 23:59:59 МСК 7 мая	1 семестр: до 23:59:59 МСК 30 ноября 2 семестр: до 23:59:59 МСК 28 мая	За день до индивидуальной даты защиты	4 мая
Последний день защиты	следующая пара по расписанию после дедлайна на создание PR по указанной лаб. работе			следующая пара по расписанию после дедлайна на создание PR по указанной лаб. работе		Индивидуальный день защиты во второй половине мая	

Лабораторные работы:

- Лаб. работа №1 – [Обзор стандартной библиотеки](#)
- Лаб. работа №2 – [Линейные списки](#)
- Лаб. работа №3 – [Обход файловой системы](#)
- Лаб. работа №4 – [Динамические структуры данных](#)

Правила сдачи работ

Правила для лаб. работ

Для того, чтобы получить баллы за пулреквест нужно выполнить следующие требования:

1. Выполнить задание к лаб. работе и загрузить задание на *moodle* в соответствующий модуль курса. Задание считается выполненным только в случае успешного прохождения всех тестов. Написанный код может проходить все проверки, но содержать ошибки, которые могут повлиять на оценку при защите работы, поэтому необходимо соблюдать следующие требования: [правила оформления кода](#)
2. После выполнения работы необходимо описать свои действия и методику выполнения работы в виде отчета. Требования к отчету описаны тут: [оформление отчета по лабораторной работе](#)
3. Загрузить файлы с кодом и отчет в систему контроля версий в соответствии с правилами: [правила работы с репозиторием](#)
4. Создать пулреквест ([pull request](#)) на *github* странице репозитория
5. Баллы за пулреквест будут зачтены, если Ваш пулреквест прошел все проверки.
Описание меток пулреквеста и ошибок при автоматической проверке работы: [<link>](#)

Для того, чтобы получить **баллы за защиту**, Вам необходимо выполнить следующие требования: [правила дистанционной защиты лабораторных и курсовых работ](#)

Перезащита работы допускается только один раз, если по итогам предыдущей защиты студент получил 0 баллов. Перезащита работы возможна только на парах до дедлайна защит лаб. работы, которые указаны в графике.

Правила проведения контрольной и ее формат

Контрольная представляет из себя набор практических и теоретических заданий. Контрольная будет проходить в специальном курсе moodle: [<link>](#)

Каждое выполненное задание дает Вам балл. Результаты контрольной могут быть аннулированы, если не соблюдать правила проведения контрольной работы.

Правила для курсовой работы

Для того, чтобы защитить курсовую работу и получить оценку необходимо:

1. Реализовать программу в соответствии с заданием и правилами: [правила создания PR к курсовой работе](#)
Важно: Для весеннего семестра предъявляются дополнительные требования: [требования к курсовым работам на 4 и 5 -- весенний семестр](#)
2. Написать пояснительную записку в соответствии с правилами: [Оформление пояснительной записки к курсовой работе](#)
3. Создать пулреквест ([pull request](#)). Пулреквест должен пройти все проверки для допуска к защите и должен быть сделан вовремя.
4. Защитить курсовую работу в назначенный индивидуальный день защиты.

День защиты определяется преподавателем и может быть в интервале 2х последних недель семестра (без учета зачетной недели). Пересдача курсовых работ не предусмотрена и возможна только в назначенный день.

Расписание защит: [<link>](#)

[Описание проведения защиты курсовой и ее оценивания](#)

Коммуникация с преподавателем

Если Вы не нашли информацию, которую искали, то Вы можете написать письмо согласно правилам: [правила оформления письма](#)
Email для вопросов: pr1@moevm.info

Коммуникация во время проведения пар по дисциплине (лабораторных, контрольных, защиты курсовых и т.д.) будет проходить с помощью цифровых платформ. Вся коммуникация с преподавателем вне пар происходит только по электронной почте.

Ссылки на полезные ресурсы

Учебные пособия и книги

1. [Учебно-методическое пособие, новая версия \(черновой вариант\)](#)
2. [Учебно-методическое пособие \(старая версия\)](#)
3. [C in a Nutshell, Peter Prinz & Tony Crawford](#)

Примеры и инструкции

1. [Репозиторий с примерами кода](#)
2. [Простейшие команды для работы в консоли Linux](#)
3. [Простейшие команды для работы в консоли Linux с git](#)
4. [Отладка программ на языке C](#)
5. [Поиск ошибок с помощью gcc](#)
6. [Автоформатирование исходного кода](#)
7. [Перенаправление вывода в Linux](#)
8. [Интересный пример изменения const переменной](#)
9. [Наиболее часто встречающиеся ошибки, выдаваемые компилятором gcc](#)
10. [Сервис для визуализации выполнения кода/работы с памятью](#)

Документация

1. [Основы Git](#)
2. [Документация GCC](#)
3. [Документация GNU C Library](#)
4. [Linux kernel coding style](#)

From:

<http://se.moevm.info/> - **se.moevm.info**

Permanent link:

<http://se.moevm.info/doku.php/courses:programming?rev=1693222153>

Last update: **2023/08/28 12:29**

