

Групповой проект - контрольная работа по программированию

Вступление

В качестве контрольной работы за третий модуль вам предлагается объединиться в команды и сделать небольшой консольный проект. Ниже будут описаны какие-то формальные требования, но надо понимать, что:

1. Они могут немного меняться, в частности баллы за формальные требования
2. Так как такое мероприятие проводится первый год, большая часть требований будет сформулирована неформально. Если есть необходимость официально закрепить какие-то из них, нужно сказать мне об этом.

Проект подразумевает самостоятельное обучение некоторым техническим и социальным навыкам, например, работе с консолью linux, формированию рабочей команды, git и так далее. Разумеется, приветствуются консультации с преподавателем (в том числе групповые, очные), инициированные учащимися.

После представления проекта могут быть выставлены три оценки (**константирующая**, **формирующая**, **творческая**), причем одному участнику будут выставлены как минимум две оценки, вторая может зависеть от роли в проекте. Оценка состоит из (групповая) + (личная). Групповая оценка одинакова для всех участников группы, личная зависит от вклада в проект каждого участника.

Стартовая оценка по каждому из типов равна 1.

Да, получение частичных баллов возможно, округление будет известно позже (с очень большой вероятностью будет округление вниз).

Формирование команд

В команде должно быть от 2 до 4 человек. *Одинокий волк* это НЕ команда, так что 2 человека как минимум. Да, больше 4 человек можно, но необходимо понимать, что от команды в 6 человек будут ожидания как от двух команд по 3 человека.

Возможные должности:

- Разработчик (a.k.a. backend)

- Дизайнер (a.k.a. frontend)
- ТехЛид
- ТимЛид
- и так далее, что сами придумаете

Этапы и дедлайны

Важно понимать, что всё контрольные точки (далее КТ), включая итоговую можно сдавать когда угодно до 30.03, даты выбраны в соответствии с вашим расписанием каникул. Первый дедлайн для группы по пятницам, второй для группы во понедельникам соотв. Каждый дедлайн заканчивается в 23:59.

1. [14.02/17.02] КТ1

Формирование группы, распределение ролей в ней, формулирование базовой идеи проекта

2. [21.02/24.02] КТ2

Презентация четко сформулированной идеи проекта, итогового состава и заготовки кода

3. [07.03/10.03] КТ3

Первый показ кода

4. [14.03/10.03] Планирование даты защит

Необходимо проставить дату итоговой защиты проекта. Таблица для группы [ИЗ](#), [И6](#).

Напоминаю, что, если проект не защитится к концу модуля, оценка всех участников будет приравнена к 1.

5. [24.03/29.03] Итоговая точка (a.k.a. ИТ)

Последняя возможность презентации проекта

Формальные требования

Выполнение формальных требований будет оцениваться в баллах к какой-либо оценке. Цветом будет обозначаться тип оценки, к которой добавляется балл. Если буква заглавная, то балл идет к групповой оценке, иначе - к личной.

• Уникальность идеи [**+0.5Б**, **+1Б**]

Балл засчитывается, если идея проекта не повторяет существующие известные продукты или является прямым развитием данных. Я от вас буду ждать игровых проектов, так как это самое лёгкое и содержательное, но это не значит, что реализовывать можно только их.

Если взять идею X, изменить в реализации пару строк кода, и получить вашу идею Y, то уникальной она не будет (условно идея крестики-нолики, где вместо крестиков зажигалка, а вместо ноликов спичка, уникальностью не обладает).

- **Сложность и объем кода [+16, +16]**

Субъективная оценка. Будут смотреться использованные алгоритмы, логика проекта. Условно не зачтены будут проекты без использования сложных принципов и идей (крестики-нолики, кликеры и тд).

- **Архитектура [+0.5Б, +0.5Б, +1Б]**

Чтобы получить балл за этот пункт, необходимо в явном виде представить внутреннюю архитектуру разработанного проекта и показать её гибкость, возможность масштабироваться и тд.

- **Баги [+0.5Б, +1Б]**

Ваш проект должен пройти стресс-тестирование на баги и не упасть во время презентации. Если был найден баг, но не критический, то балл умножается на 0,5. Если баг был критическим, например полностью слетел прогресс или чтобы выйти из состояния нужно перезапустить программу, то балл умножается на 0.

- Балл умножается на 0.5 если были найдены некритические баги позволяющие продолжить пользоваться приложением
- Балл умножается на 0 если были найдены критические баги, сбрасывающие прогресс, закрывающие программу и тд.

- **Презентация [+0.5Б, +1Б]**

Время выступления не более 7 минут.

Во время презентации каждый участник проекта **без страха** высказывается (да, уверенность во время выступления тоже будет учитываться) и находится в курсе происходящего. Также учитывается внешний вид самой презентации

- **Командообразование [+16]**

Каждый участник знает свою роль, отвечает только за свою часть проекта

- **UI [+16]**

У проекта существует выделяющийся(концептуальный) дизайн

- **UX [+16, +16]**

У проекта интуитивно понятный интерфейс, есть все необходимые кнопки (условно выйти из текущего состояния).

- **Формат кода [+0.5Б]**

Соблюден code-style, есть документация/комментарии.

- **Условные правила приличия [+0.5Б, +0.5Б]**

Соблюдены правила приличия, например:

- Нету `using namespace <smth>;`
- `auto` без причины - признак не очень хорошо написанного кода.
- Односимвольная переменная без обоснования (условные `math.e`, `math.pi` по другому не назовешь) приравнивают к нульзначному баллу за этот пункт.
- Просто набор переменных констант вынесен в `#define`, `enum class` или отдельный `namespace`
- Я не заметил UB (не бойтесь, оно у вас точно есть)

- **Соблюдение дедлайнов [+1Б, +1Б, +1Б]**

Все этапы выполнены и представлены вовремя. Если не соблюден 1 - балл умножается на 0,5. Более одного - на 0.

Все этапы выполнены и представлены вовремя. Если не соблюден 1 - балл умножается на 0,5. Более одного - на 0.

- **Экстра балл [+1Б, +1Б]**

Если в вашем проекте что-то сделано просто невероятно хорошо, будь то алгоритмы, сложнейшая логика или восхитительный дизайн, можно будет получить экстра балл. Да, первая команда получает +0.5Б за смелость.

Пример работы над проектом

Команда 1: "Arkanoid 2.0"

Участники:

- Разработчик 1 (лидер и разработчик)
- Разработчик 2 (дизайнер)
- Разработчик 3 (второй разработчик)

Идея проекта: Создание консольной версии игры "Arkanoid", где при столкновении с кирпичиком шарик делится на два меньших, что добавляет элемент стратегии и сложности.

Этапы работы:

1. **КТ1 (14.02 - 17.02):** Команда собралась и разделилась на роли: два разработчика и дизайнер. Сформулирована базовая идея проекта (консольная версия "Arkanoid"), начата работа над концепцией игры.
2. **КТ2 (21.02 - 24.02):** В процессе работы команда поняла, что лидер не справляется с ролью руководителя, хоть и является хорошим разработчиком. Дизайнер предложил переработать идею, добавив новый элемент: шарик будет делиться на два при столкновении с кирпичиком. Команда перераспределила роли, и каждый участник взял на себя четкие задачи.
3. **КТ3 (07.03 - 10.03):** К этому времени команда подготовила шаблон проекта с дизайном, но без логики игры. Были четко сформулированы ожидания от каждого участника, а также установлено рабочее время для совместной работы.
4. **Итоговая точка (24.03):** Проект был полностью готов, хотя реализованная идея отличалась от первоначальной. Игра работала без багов и была готова к презентации. Каждый участник уверенно представил свою часть работы, и команда получила высокую оценку за уникальность идеи и качество исполнения.

Формальные требования:

- Дизайнер + руководитель (5Б, 5Б)
- Разработчик 1 (5Б, 5Б)
- Разработчик 2 (4Б, 4Б)

Команда 2: "Проект без названия"

Участники:

- Лидер (отвечает за идеи)
- Разработчик 1
- Разработчик 2

Идея проекта: Изначально команда планировала создать проект, но в итоге осталась без четкой идеи.

Этапы работы:

1. **КТ1 (14.02 - 17.02):** Лидер команды "украл" идею у GPT, но не записал её. В команде был представлен список участников, где лидер гордо обозначил себя как "лидера".
2. **КТ2 (21.02 - 24.02):** Команда забыла идею, и от проекта осталась только название. Лидер не смог ответить на вопросы разработчиков и отложил их, ссылаясь на более важные

задачи.

3. **КТЗ (07.03 - 10.03):** К этому времени участники забыли о проекте. Разработчики, осознав, что не могут работать без четкой идеи, ушли в другой проект, оставив лидера одного.
4. **Итоговая точка (24.03):** Лидер попытался оправдаться, говоря, что проект будет, но не сразу. Однако команда не смогла представить ничего конкретного, и итоговая презентация провалилась.

Формальные требования:

- Уникальность идеи: +0Б
- Сложность и объем кода: +0Б
- Архитектура: +0Б
- Баги: +0Б
- Презентация: +0Б
- Командообразование: +0Б
- UI: +0Б
- UX: +0Б
- Внешний вид кода: +0Б
- Соблюдение дедлайнов: +0Б

Итоговая оценка: 0Б

Примеры кода

Можно взять в качестве примера какие-то известные открытые библиотеки или консольные игры:

- [ninvaders](#)
- [Консольный Ray Tracing](#)
- [Gif-visualize](#)