



олимпиада

Трассектория будущего

Задание полуфинала для номинации DevOps

Постановка задачи	2
Пояснение эксперта номинации	2
Задание - игра «Заливка»	2
Требования, участвующие в оценке решения	4
Шкала оценки	7
Сроки для выполнения работы	7
Оформление проекта	7
Предоставление результатов	8
Плагии и несамостоятельное выполнение	8
Определение финалистов	8

Постановка задачи

Ознакомьтесь с заданием, подготовленным экспертом номинации.

Реализуйте задание в соответствии с требованиями.

Оформите и предоставьте результат.

Пояснение эксперта номинации

В этом году в качестве задания я решил предложить вам игру – «Заливка».

Когда два игрока, находясь в противоположных углах площадки и выбирая цвет, "заливают" поле по очереди. Игра идёт на гексагональном поле.

Это и сама по себе интересная и динамичная задача, и в нее можно поиграть с друзьями; если вы ее запрограммируете, то с компьютером.

Задание - игра «Заливка»

В игре есть два игрока (Игрок1, Игрок2).

Размерность поля 47 x 21 гексагональных клеток в виде прямоугольника.

На поле могут находиться стены, в заранее известной конфигурации.

Очевидно, что стены не находятся в позициях старта игроков, и симметричны для обоих игроков относительно центра.

От стартовых позиций игроков можно проложить маршрут друг к другу.

Каждая клетка поля имеет один из 10 цветов, заданных случайным образом:

1. Белый
2. Ярко-красный
3. Зеленый
4. Ярко-зеленый
5. Синий
6. Светло-синий
7. Желтый

- 8. Розовый
- 9. Оранжевый
- 10. Черный (стены, цвет не доступен для выбора)

Игроки начинают игру в противоположных углах.

Первый игрок начинает игру в левом нижнем углу.

Второй игрок - в правом верхнем.

Игроки выбирают цвет по очереди, начиная с Игрока 1.

Для выбора цвета разрешены те цвета, с которыми есть сопряжения (соприкосновения).

Нельзя выбрать цвет соперника.

При выборе нового цвета к пространству игрока добавляются клетки, соприкасающиеся с полем игрока, в которых указан данный цвет (т.е. при заливке цветом поле постепенно растет не менее, чем на 1 клетку).

Игра останавливается в тот момент, когда игрок не может выбрать цвет и совершить ход.

Побеждает тот игрок, у которого в момент окончания игры больше занятых клеток.

В случае сетевого взаимодействия сервер инициализирует поле и отправляет его игроку.

Дальше игрок и сервер обмениваются сообщениями о выбранном цвете.

Дополнительным плюсом будет разработка компьютерного противника для самостоятельной игры.

Желательно, чтобы компьютерный противник умел мыслить стратегически на несколько шагов вперед.

Т.е. заливка случайным образом - минимальный уровень.

Выбор стратегия наибольшего роста в пределах одного хода - базовый уровень.

Стратегия на n'шагов - повышенный уровень.

Требования, участвующие в оценке решения

Всего в оценке решения участвуют 25 требований. Некоторые из них предполагают дополнительные подпункты или комментарии о дополнительном плюсе, означающем “усиление” оценки. В условиях ограниченного количества времени, выделенного на подготовку решения, вы можете выбирать из 25 требований те, в которых у вас есть уверенность.

1. Наличие математической модели игры ("математика" компьютерного игрока как плюс)
 2. Наличие блок-схемы (декомпозиции) разработанного решения
 3. Скрипты сборки (make, cmake, bazel или др. согласно выбранного стека технологий)
 4. Генерация документирования кода (doxygen, javadoc, shinx и пр.)
 5. Наличие codestyle + инструментов соблюдения соглашения по выбранному стилю (автоформатирование, ссылки на выработанные решения)
 6. Дорожная карта развития проекта. Вехи, наличие автоматизированных отчетов по степени готовности проекта/этапа проекта. Формирование отчетов по количеству и сложности закрытых тикетов на разработку конкретным специалистом (BugTracker Systems)
 7. Наличие обоснования применения выбранной методологии разработки (RAD, Agile, Scrum, TDD, BDD)
 8. Если предполагается наличие GUI - то соблюдение требований по эргономике
- Наличие адаптации для людей с ограниченными возможностями как плюс
9. Unit тесты + отчеты по тестированию. Наличие отчета о покрытии кода тестами.
 10. Нагрузочное тестирование и анализ полученных графиков как плюс.
 11. Наличие документированного Pipeline Jenkins, Gitlab CI (Использование иного CI/CD допускается при наличии документирования процесса)
 - 11*. Наличие параметрического multibranch с передачей параметров сборки как плюс
 12. Наличие кодеметрики (оценка сложности алгоритмов O-большое и др. характеристики кода)
 13. Наличие базы данных как источника информации, тестов и пр.
 - 13*. Распределенная БД с решенными вопросами репликации и устойчивости как плюс
 - 13*. Наличие обоснованного EXPLAIN для самых сложных запросов
 14. Наличие описания бизнес процесса по модели BPMN или иной нотации (IDEF нежелателен)
 15. Наличие сетевого взаимодействия клиент-сервер с описанным протоколом
 16. Обоснование выбранного протокола (REST, SOAP, gRPC)
 17. Управление мониторингом системы (как вы это понимаете)
 18. Менеджмент и управление журналами (логами) работы программы

Ведение логов работы программы. Обращаю внимание на раздельное использование потоков вывода и потоков ошибок

19. Использование систем статического анализа кода

20. Наличие системы уведомления команды разработчиков (RSS Feed, Jaber, IRQ все, что не Tg)

20*. Использование Maternmost бота для управления уведомлениями и ведения канала - как плюс.

21. Менеджмент ключей ssh, наличие цифровой подписи (иерархии цифровых подписей)

22. При необходимости - наличие описания стенда, версии компилятора, зависимости от сторонних библиотек

23. Использование многокомпонентной системы как плюс **

24. Наличие действительных положительных отзывов о работе независимыми игроками, сотрудниками кафедры. (Документ + цифровая подпись) (Цифровая подпись - это ключом созданный grg, а не от Госуслуг. И именно цифровая подпись - .sig)

24* Наличие в репозиториях закрытых ключей будет считаться серьезным недостатком. Все данные которые были подписаны найденным ключом считаются недействительными и обнуляются.

25. Наличие таблицы рекордов (top 10) для заданной конфигурации стен. Имя игрока, кол-во шагов, результат игры. Сортировка по результату; хранение таблиц в базе данных.

** Под многокомпонентной системой подразумевается наличие выделенных библиотек, выделенных в самостоятельные проекты.

* hook со стороны сервера с использованием API

Дополнительно проверяется, что доступ к БД осуществляется с необходимым минимумом привилегий. Работа с правами администратора считается грубой ошибкой. Все взаимодействия выполненные с необоснованно повышенными правами не будут учитываться в общем зачете.

Шкала оценки

Выполнение каждого пункта требований будет оценено по следующей шкале:

0 баллов – отсутствует, либо имеются критические ошибки

- 1 балл – формальное исполнение, реализовано «удовлетворительно»
- 2 балла – в наличии, реализовано на «хорошо»,
- 3 балла – в наличии, реализовано на «отлично», реализовано с «усилением»
- 4 балла – лучшие реализации среди участников

Максимальное количество баллов, которое может получить решение: 100.

Сроки для выполнения работы

Результаты предоставляются **до 20 марта включительно**.

Оформление проекта

Решение оформляется в виде архива, размещенного в папке любого облачного хранилища. Назовите папку по своей Фамилии Имени Отчеству.

В архиве должно быть представлено все, что может подтвердить выполнение требований: скрипты сборки, копии БД, цифровые подписи, доступы. Обязательно наличие файла README, в котором нужно указать, как ориентироваться в архиве.

Предоставление результатов

Отправьте ссылку на папку, включающую ваш архив, до **20 марта** включительно на почту info@tboimpiada.ru, указав в теме письма «Полуфинал_ТБ_DevOps_ФИО(участника)».

Проверьте, что ссылка на папку и документы, входящие в нее, предусматривают право просмотра читателям.

Плагиат и несамостоятельное выполнение

Работы, уличенные в плагиате, оцениваться не будут: участники, приславшие не оригинальные работы — из дальнейшего участия выбывают. Идентичные работы исключаются из соревнований без выяснения причин.

Участие индивидуальное. Командное участие не допускается.

Оргкомитет оставляет за собой право связаться выборочно с любым из участников полуфинала перед подведением итогов, чтобы убедиться в самостоятельном выполнении задания.

Оргкомитет также оставляет за собой право собрать участников полуфинала в режиме видеоконференции для защиты присланных проектов.

Определение финалистов

Итоги подведем не позднее 26 марта. Результаты полуфинала опубликуем на сайте Олимпиады в виде рейтинговых таблиц. Финалистами станут **15** человек, показавшие лучшие результаты. При определении финалистов будут учтены условия допуска (приложение 4, [Положение](#) об олимпиаде, стр. 57).