

Клавогонки

Клавогонки – гонки на клавиатуре. Несколько игроков соревнуются в том, кто быстрее и без ошибок напечатает текст. Идея – <https://klavogonki.ru/>

Клавогонки – многопользовательская игра. Игру координирует сервер, к которому подключаются клиенты.



Обычный, открытая игра, таймаут 20 сек
01 20



Эпидемия ящура. Известен как препарат Коха. Сыворотка и вирус. Процент вакцинированных лошадей. Эпизоотии. Императорские конюшни в Мюрцштеге, Нижняя Австрия. Квалифицированные ветеринары. Мистер Генри Блэквуд Прайс.
[Любезное](#) предложение беспристрастной проверки.

Люб

[Показать экранную клавиатуру](#)

[Как играть?](#) [Пожаловаться на текст](#)



4 участника

+20% очков



Это вы



№1864
[selinvi55](#)



1 место

01:00.1

261 зн/мин

11 ошибок (4,20%)

Гость



Интерфейс <https://klavogonki.ru>

Сервер

Сервер принимает входящие tcp-соединения с помощью `ServerSocket`, порт задается первым параметром командной строки. Порт по умолчанию – 5619

Формирование группы

Сервер формирует группы для участия в гонке по 3 игрока. Если не удаётся набрать группу в течение 30 секунд, то группа состоит из столько игроков, сколько подключились. По мере подключения или отключения игроков сервер сообщает клиентам список игроков, попадающих в группу. Если игрок отключился, то он выбывает из группы, и его место может занять новый игрок.

Сервер поддерживает несколько групп одновременно.

Отправка текста

После того, как сервер сформировал группу, за 5 секунд до начала гонки он отправляет подключенным игрокам текст, который игроки должны набрать.

Размер текста – около 300 символов, включая знаки препинания и пробелы. Сервер выбирает текст случайно из 5 заранее заготовленных файлов. Файлы с текстами находятся в ресурсах сервера – `src/main/resources/texts`. Тексты можно взять из книг или статей

По истечении 5 секунд, сервер отправляет сообщение о начале гонки.

Гонка

Обновления от сервера

Во время гонки сервер раз в секунду отправляет игрокам:

1. Массив статусов игроков. Для каждого игрока:
 - a. Имя игрока
 - b. Количество набранных символов
 - c. Количество совершенных ошибок
 - d. Флаг, определяет ли данный элемент массива текущего игрока. По этому флагу клиент понимает, нужно ли в таблице отмечать игрока подписью «это Вы»
 - e. Флаг ошибки. Если сервер обнаруживает, что игрок более не соединен с сервером, то сервер устанавливает флаг ошибки для данного игрока в `true`

и считает это обновление для данного игрока последним (больше ему обновления не отправляет)

2. Время, оставшееся до конца гонки

Обновления от клиентов

Во время гонки клиенты отправляют серверу обновления:

1. Сколько ошибок игрок совершил при наборе текста
2. Сколько символов набрал игрок

Эту информацию сервер использует в дальнейшем для отправки обновлений клиентам.

Окончание гонки

Гонка заканчивается, если выполняется одно из условий:

1. Все игроки набрали текст
2. Прошло более 3-х минут с начала гонки

При окончании игры сервер отправляет игрокам итоговую таблицу результатов.

Итоговая таблица содержит:

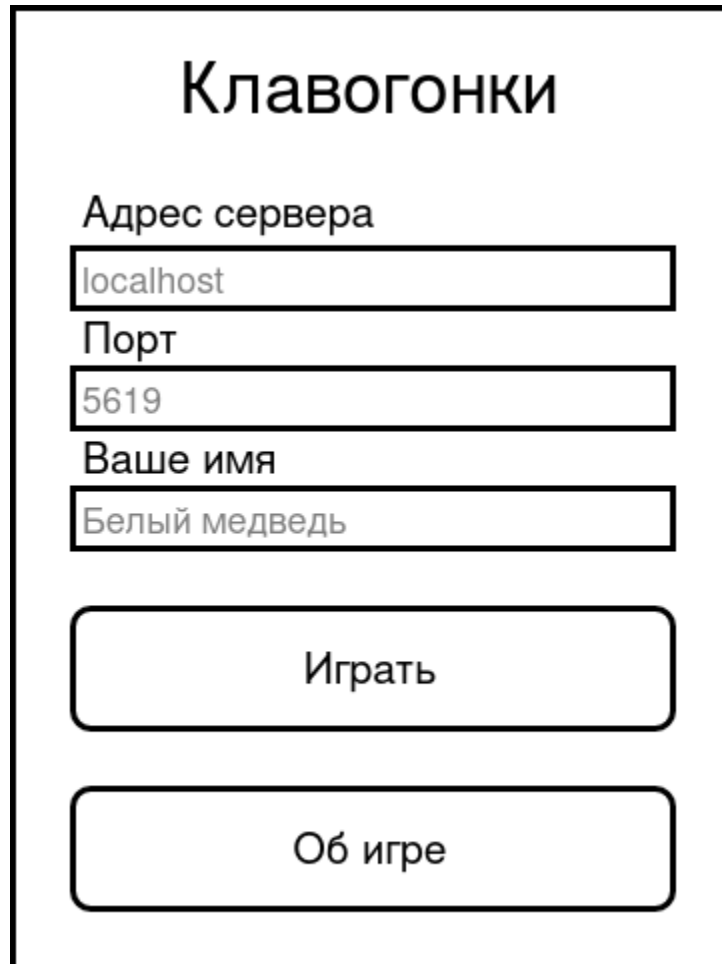
1. Имя игрока
2. Время набора текста
3. Количество совершенных ошибок
4. Количество набранных символов. Если игрок не успел набрать текст, то это число будет меньше длины текста

После этого сервер закрывает соединения с группой клиентов.

Клиент

Клиент – приложение, с помощью которого игроки участвуют в клавогонках

Экран запуска



Клавогонки

Адрес сервера
localhost

Порт
5619

Ваше имя
Белый медведь

Играть

Об игре

Пример экрана начала игры

Экран запуска содержит:

1. Поля для ввода:
 - a. Адреса сервера, по умолчанию – localhost
 - b. Порта, по умолчанию – 5619
 - c. Имени игрока, по умолчанию выбирается случайное из заранее предопределенного списка имен

2. Кнопки:

- а. «Играть». При нажатии клиент устанавливает соединение с сервером, после чего открывает экран гонки
- б. «Об игре». При нажатии отображается информация об авторе игры и сообщение проверяющему

Значения по умолчанию используются, если пользователь ничего не ввёл. Эти же значения отображаются подсказками к полям ввода. Пример – на рисунке выше.

Экран гонки

Клавогонки

Игроки

1. Арина 52%, 3 ошибки, 210 сим/мин

2. Петя 51%, 3 ошибки, 200 сим/мин

3. Иван (это Вы) 30%, 2 ошибки, 146 сим/мин

До конца игры
156 секунд

Текст для ввода

ChatGPT - чат-бот с искусственным интеллектом, разработанный компанией OpenAI и способный работать в диалоговом режиме, поддерживающий запросы на естественных языках. ChatGPT — большая языковая модель, для тренировки которой использовались методы обучения с учителем и обучения с подкреплением.

Вводите текст здесь

искусстве

Пример экрана процесса игры. Зеленым выделено текущее слово для набора, подчеркнуты оставшиеся буквы в текущем слове

Экран гонки содержит:

1. Таблицу игроков. Текущий игрок в таблице помечается отметкой «это Вы»

- a. До начала игры таблица отображает имена игроков
- b. Во время гонки порядок игроков в таблице такой:
 - i. Игрок *a*, набравший большее количество символов, чем игрок *b* выше в таблице игрока *b*
 - ii. Если количество набранных символов одинаковое, то в таблице выше тот игрок, который совершил меньшее число ошибок
 - iii. Если и время набора текста совпадает, оба игрока разделяют одно место
 - iv. Назначать места игрокам может или клиент, или сервер. На усмотрение студента
- c. Во время игры таблица отображает для каждого игрока:
 - i. Имя
 - ii. Скорость набора текста в символах в секунду
 - iii. Количество ошибок, совершенных игроком
 - iv. Процент набранного текста
- d. Если для игрока флаг ошибки – `true`, то в таблице показывается, что игрок потерял соединение с сервером

2. Счетчик

- a. До начала игры – время, оставшееся до начала игры в секундах. Его отправляет сервер
- b. Во время игры – время оставшееся до конца игры. Его тоже отправляет сервер

3. Текст, который игроки должны набирать. Текст становится доступен только после того, как сервер отправил текст. До тех пор пока текст недоступен должен отображаться текст «Здесь будет текст»

4. Поле ввода

- a. До начала игры поле заблокировано
- b. Во время игры поле разблокировано, и пользователь вводит в него текст
 - i. В базовом варианте в поле текст вводится полностью: если текст для ввода – «Привет, Java», то в поле ввода должен быть текст «Привет, Java»

- ii. На 1 дополнительный балл реализовать вариант, как в настоящих клавиатурках. Там в поле ввода отображается последнее не полностью введенное слово, как на рисунке выше. Пример:

1. Текст – «Привет, Java»
2. Пользователь вводит «Привет,», нажимает пробел
3. Поле ввода очищается
4. Пользователь вводит «Java»
5. Ввод текста полностью засчитывается

- c. Поле ввода блокируется, когда пользователь полностью вводит текст или выходит время игры

Ошибки ввода

Ошибка – когда игрок вводит символ, не совпадающий с текущим символом в тексте.

Примеры:

1. Текст – «Привет». Игрок вводит «п», удаляет символ «п», вводит «П». В результате – одна ошибка
2. Текст – «Привет». Игрок вводит «п», «р», удаляет символ «р», а затем символ «п» и вводит «Привет». В результате – одна ошибка
3. Текст «Привет, Java». Игрок вводит «П», «Р», затем удаляет «Р», вводит «ривет», пробел, затем удаляет пробел, вводит запятую, пробел и «Java». В результате – 2 ошибки:
 - a. Из-за ввода «Р»;
 - b. Из-за ввода пробела вместо запятой;
4. Текст – «Привет». Игрок вводит «п», «р», удаляет символы «р» и «п» и вводит «Р», удаляет символ «Р», вводит «Привет». В результате – одна ошибка

Когда программа обнаруживает ошибку ввода, она сообщает об этом пользователю. Например, показывает текст «Исправьте опечатку».

На бонусный балл реализовать:

1. Подсветку текущего положения текста в оригинальном тексте
2. При вводе ошибки подсветить текущее положение в тексте красным

Пример – на рисунке ниже:



Обычный, открытая игра, таймаут 20 сек
00 23



Эпидемия ящура. Известен как препарат Коха. Сыворотка и вирус. Процент вакцинированных лошадей. Эпизоотии. Императорские конюшни в Мюрцштеге, Нижняя Австрия. Квалифицированные ветеринары. Мистер Генри Блэквуд Прайс. Любезное предложение беспристрастной проверки.

Исправьте опечатку!

вац

[Показать экранную клавиатуру](#)

[Как играть?](#) [Пожаловаться на текст](#)



4 участника

+20% очков



Это вы



№1864
selinvi55



Гость



Обновления от сервера

Получая обновления от сервера, клиент обновляет:

1. Таблицу игроков
2. Время, оставшееся до конца игры

Раз в секунду клиент отправляет обновления серверу:

1. Сколько ошибок игрок совершил при наборе текста
2. Сколько символов набрал игрок

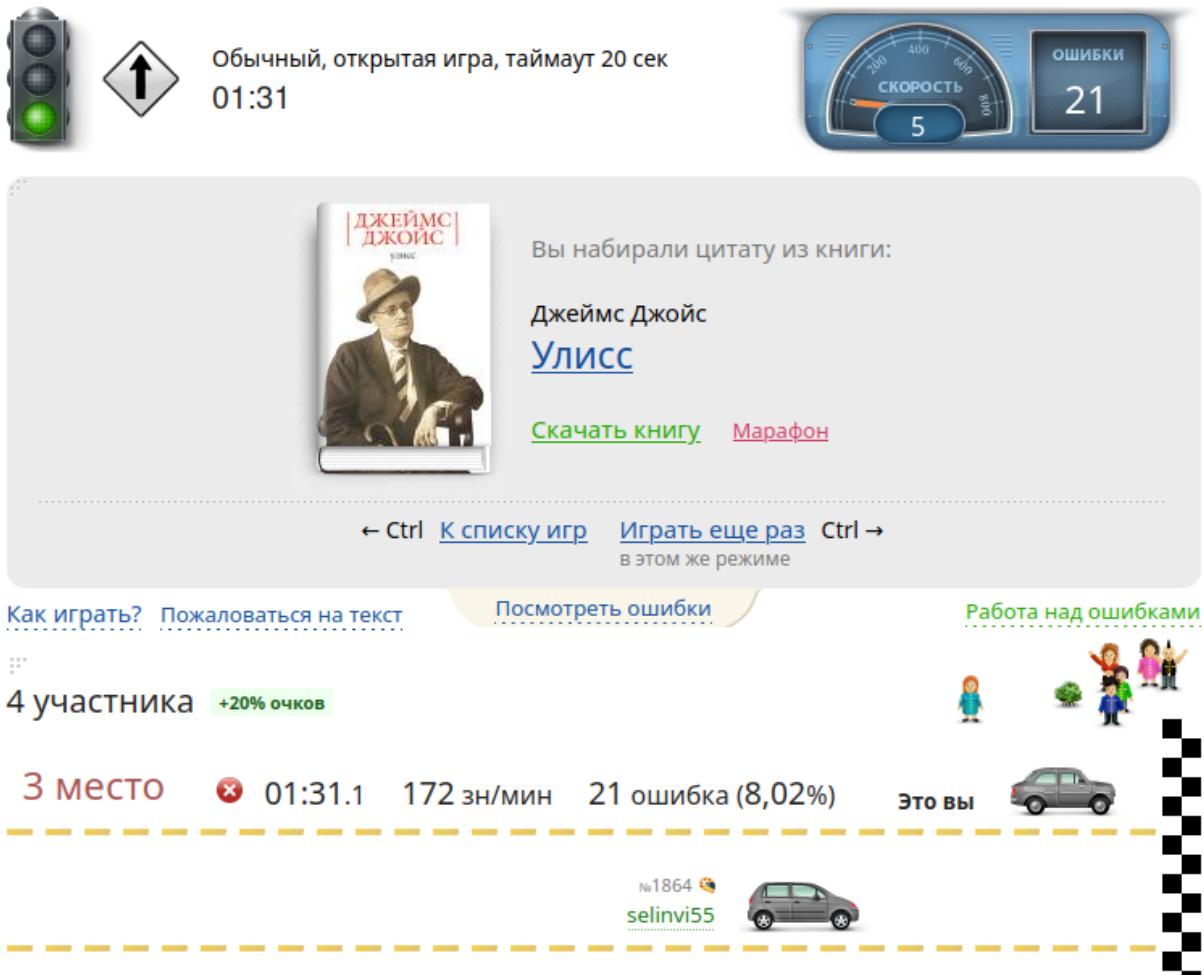
Конец игры

Получив сообщение об окончании игры, клиент:

1. Показывает итоговую таблицу

2. Поздравляет текущего игрока, если он победил
3. Предлагает игроку сыграть заново

Это может быть сделано в новом окне или в старом, на усмотрение студента. Пример окончания игры – на рисунке ниже:



Правила определения порядка игроков в итоговой таблице – такие же, как и в таблице во время игры.

Требования к реализации

1. Нельзя использовать сторонние зависимости, кроме JUnit и JavaFX
2. Maven-проект из двух модулей – клиент и сервер – с родительским `pom.xml`
3. Имя архива – **по формуле:** `HW3_LastNameFirstName.zip`
4. Для создания GUI использовать только Swing или JavaFX

5. Покрытие тестами каждого из приложений более 30%
6. Для запуска предоставить скрипты для своей операционной системы
7. Приведенные в описании данного задания иллюстрации, заимствованные с сайта <https://klavogonki.ru> для примеров, не являются обязательными. Они служат только для пояснения содержания того, что показывается на экранах приложений.

Оценивание

Если нет взаимодействия по сети, т.е. реализована только одиночная игра без сервера, то максимальная оценка – 7 со всеми бонусами. Если бонусные задания не реализованы, то максимальная оценка – 5.

Если реализован и клиент, и сервер, то максимальная оценка без бонусов – 8, с бонусами – 10.

Можно дополнительно реализовать что-нибудь, не описанное в тексте.

При необходимости можно добавлять дополнительные поля в протоколе, менять интерфейс программы. На рисунках представлены лишь примеры. Главный принцип – отклоняться разумно, не в ущерб функциональности. Например, не показывать таблицу со списком игроков нельзя, но можно изменить текст кнопки «Играть» на «Начать игру»

На что обратим внимание

1. Полноту реализации;
2. Наличие серьезных багов;
3. Обработку ошибочных ситуаций;
4. На корректное закрытие ресурсов (файлов, сокетов) при завершении их использования и при выходах из приложений сервера и клиента;
5. На валидацию ввода: аргументов командной строки, полей ввода;
6. Использование ООП, декомпозицию задачи, соблюдение соглашений об именовании, чистоту кода;
7. Использование возможностей maven: управление зависимостями, сборку jar'ов, запуск тестов через maven, наличие родительского pom.xml;
8. Использовании возможностей Java, в том числе и модульной системы, многопоточности;
9. Использование GUI;
10. Скрипты запуска;

11. Тесты.