

МИНОБРНАУКИ РОССИИ

РГУ НЕФТИ И ГАЗА (НИУ) ИМЕНИ И.М. ГУБКИНА

Факультет
Кафедра

комплексной безопасности ТЭК
безопасности информационных технологий

ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ ИГРЫ «МЕМО»

ВЫПОЛНИЛИ:

Студенты группы

КС-21-04

(номер группы)

Ермилова Екатерина Владимировна

(фамилия, имя, отчество)

Еськов Максим Дмитриевич

(фамилия, имя, отчество)

Москва, 2025

Содержание

1 Объект испытаний.....	3
2 Требования к программе.....	4
3 Требования к программной документации.....	8
4 Состав и порядок испытаний.....	9
5 Методы испытаний	10

1 Объект испытаний

Мемо

На минимальный балл:

У нас 10 пар карточек, содержащих цифры от 1 до 10. Мы их располагаем на поле 5x4. Сохраняем историю игр т у нас есть топ 20 лучших результатов.

На хорошую оценку:

Количество карточек можно задать самостоятельно от 5 до 30. У нас есть уровни. На первом уровне время на ход составляет до 1 минуты. На каждом следующем уровне время уменьшается на 5 секунд. Всего 10 уровней. Сохраняем историю игр т у нас есть топ 20 лучших результатов.

На отлично:

Перед началом игры мы можем сами ввести "пары" значений. Например, английское слово и его перевод. Набор пар можно сохранить и потом его использовать не один раз. В ходе игры мы можем воспользоваться 3-мя подсказками:

1. Открыть все карточки на 2 секунды.
2. Отменить ход
3. Показать пару к конкретной карточке.

Каждой подсказкой можно воспользоваться только 1 раз за уровень. Использование подсказки прибавляет + 2 хода к вашему результату.

У нас есть уровни. На первом уровне время на ход составляет до 1 минуты. На каждом следующем уровне время уменьшается на 5 секунд. Всего 10 уровней. Сохраняем историю игр т у нас есть топ 20 лучших результатов.

Мы выбрали на хорошую оценку (4):

Количество пар карточек можно задать самостоятельно от 5 до 30. У нас есть уровни. На первом уровне время на ход составляет до 1 минуты. На каждом следующем уровне время уменьшается на 5 секунд. Всего 10 уровней. Сохраняем историю игр, то есть у нас есть топ 20 лучших результатов.

2 Требования к программе

2.1 Общие требования

Отчуждаемость исходного кода.

Устойчивость к некорректным действиям пользователей.

Корректная поддержка одновременного обращения большого количества пользователей (больше, чем может одновременно участвовать в одной игровой сессии).

2.1.1. Регистрация и аутентификация игроков

В рамках данной реализации мы будем использовать локальных пользователей, которые создаются непосредственно в самом приложении и заносятся в базу данных

- Возможность создания учётной записи или входа в существующую.
- Хранение статистики игроков.

2.1.2 Подключение к игре

Игрок проходит авторизацию для подключения к игре.

2.1.3 Настройка параметров игры

- Игрок выбирает уровень игры от 1 до 10
- Игрок выбирает количество пар карточек от 5 до 30.

2.1.4 Генерация поля игры

- Алгоритм генерирует в случайном порядке пары карточек и их номинал

- Устанавливается таймер хода в соответствии с уровнем игры.

2.1.5 Процесс игры

- Игрок выбирает две пары карточек и открывает их.
- Время на ход ограничено, в соответствии с уровнем игры.
- Если игрок не успел сделать ход, то игра считается проигранной и завершается.
- Если игрок находит одинаковую пару карточек, то карточки остаются номиналом вверх до окончания игры.

Если все пары карточек найдены успешно, то игра завершается, а результат игрока записывается в топ.

2.1.6 Проверка пар карточек

Проверка номинала пар карточек

Если значение пары карточек одинаково, то карточки остаются на поле номиналом вверх до конца игры.

Если значение пары карточек различно, то карточки переворачиваются шапкой вверх, ход игрока начинается снова, таймер обновляется.

2.1.7 Завершение игры и определение победителя

Игроку зачисляется **поражение**, если он не успел выполнить ход в отведенное время.

Игроку зачисляется **победа**, если он открыл все пары карточек.

Пользователь может завершить игру принудительно, выполнив процедуру завершения игры.

2.1.8 Реализация действий после завершения игры

Задача: после завершения игры обновить статистику игроков.

Если итог игры **победа**, игрок записывается в топ.

Добавить информацию о победах игроков и набранных очках.

Обновить рейтинг игроков в базе данных.

2.2 Требования к интерфейсу

2.2.1 Интерфейс для создания игры

Задача: разработать интерфейс для взаимодействия с игрой.

В интерфейсе должны быть реализованы процедуры регистрации и авторизации с возможностью ввода входных параметров для логина и пароля пользователя.

В интерфейсе должна быть реализована процедура с возможностью ввода входных параметров для ввода уровня игры и количества пар карточек.

Интерфейс взаимодействия должен быть реализован в Oracle PL/SQL.

2.2.2 Интерфейс для входа в игру

Задача: создать интерфейс для начала игры.

После авторизации, пользователь выбирает уровень, где 1 уровень – 60 секунд на ход; 2 уровень – 55 секунд на ход и так далее до максимального 10-го уровня, где время хода длится 15 секунд.

После выбора уровня пользователь выбирает количество пар карточек в диапазоне от 5 до 30 пар.

Новая игра начинается, после прохождения игроком авторизации выбора уровня, количества пар карточек и запуска процедуры старта игры игроком.

2.2.3 Интерфейс для выбора пар карточек

Задача: разработать интерфейс для выбора пар карточек

Игрок вводит номера позиций карточек, которые хочет открыть и после этого запускает процедуру для открытия пары карт.

Как только игрок запускает процедуру с вводом карточек, пара карточек открывается.

2.2.4 Интерфейс выхода из игры

Задача: разработать корректный выход из игры

После того, как пользователь завершил игру по одному из вариантов, представленных в пункте 2.1.7, игрок может выполнить процедуру выхода из игры.

2.2.5 Интерфейс просмотра топ игроков

Для просмотра таблицы результатов пользователь вызывает процедуру результатов и получает вывод топ-20 игроков.

2.2.6 Интерфейс для отображения состояния игры

Задача: показывать текущие данные о игре, включая время на ход.

Интерфейс должен отображать как начальное поле игрока, так и поле с промежуточными результатами, а именно:

А) Ситуация после выбора пары карточек (если совпали, то поле обновляется и эта/эти пары карточек остаются «перевернутыми»)

Б) Ситуация после выбора пары карточек (если текущие выбранные карточки НЕ совпали, то пара показывается, но при выборе следующей пары, предыдущие «переворачиваются» шапкой вверх).

Интерфейс будет отображать завершение игры при попытке хода, в том случае, если время на ход уже вышло.

3 Требования к программной документации

Для проведения всех необходимых проверок должны быть предоставлены следующие материалы:

- 1) SQL скрипт для развертывания схемы базы данных;
- 2) SQL скрипт для создания пакета, реализующего логику игры;
- 3) SQL скрипт для создания тела пакет, реализующего логику игры;
- 4) SQL скрипт, содержащий набор тестовых команд, для запуска и реализации игры.

4 Состав и порядок испытаний

Для проведения испытаний необходим 1 сервер с установленным СУБД Oracle версии не ниже 12-ой, а также хотя бы 1 рабочее место конечного пользователя, с доступом по сети к серверу БД.

На рабочем месте должно быть установлено приложение для доступа к СУБД Oracle, такое как SQL Developer или DBeaver.

Порядок поведения испытания будет следующим:

- 1) Развертывание БД игры Мемо на предоставленном сервере. Для этого необходим пользователь базы данных с правами разработчика.
- 2) Развертывание пакета с логикой игры на сервере. Для этого необходим пользователь базы данных с правами разработчика.
- 3) Создание пользователей, под которыми будет проходить тестирование игры и предоставление им созданной ранее роли. Для этого необходим пользователь базы данных с правами администратора.
- 4) Проведение позитивного тестирования.
- 5) Проведение негативного тестирования.
- 6) Проведение нагрузочного тестирования.
- 7) Проведения иных проверок.
- 8) Составления протокола испытаний.

5 Методы испытаний

5.1 Порядок действий развертывания среды для выполнения программы и методики испытаний представлен ниже:

1. Подключиться к удаленному рабочему столу, удовлетворяющему требованиям раздела 4 – Состав и порядок испытаний;
2. Запустить приложение доступа к БД Oracle DBeaver;
3. Выполнить подключение к БД с правами администратора и открыть новый SQL-скрипт.
4. Открыть файл с названием tables.sql, который прилагает к файлу ПИМ.
5. Открыть новый редактор SQL, скопировать содержимое файла tables.sql в DBeaver и запустить создание таблиц. (таблицы рекомендуется создавать по порядку содержимого файла сверху вниз)
6. Открыть файл с названием package. sql, приложенный к ПИМ.
7. Скопировать содержимое файла в скрипт DBeaver и запустить сборку спецификации пакета package (полностью выделив содержимое ctrl+a и запуск).
8. Открыть файл с названием procedures. sql, приложенный к ПИМ.
9. Скопировать содержимое файла в скрипт DBeaver и запустить создание пакета procedures (полностью выделив содержимое ctrl+a и запуск).
10. Открыть файл с названием start_file. sql, приложенный к ПИМ.
11. Скопировать содержимое файла в скрипт DBeaver. Затем, также сверху вниз выполняем анонимные блоки. Например, для регистрации пользователю требуется ввести свой логин и пароль, поле для ввода представлено ниже.

BEGIN

player_level_pkg.register_player('логин', 'пароль');

END;

Примечание - Включите вывод DBMS_OUTPUT (если используете SQL*Plus или аналогичный клиент)

SET SERVEROUTPUT ON; - команда включения

После ввода, со слова BEGIN до слова END; выделяется полностью анонимный блок и запускается в oracle.

После регистрации необходимо **повторно** ввести логин и пароль в блок
player_level_pkg.login('логин', 'пароль');

Выбор уровня и количества пар карточек производятся в блоках

BEGIN

player_level_pkg.set_level('XX');

END;

Соответственно, где XX – значение уровня (от 1 до 10), заполнение анонимных блоков выполнять в соответствии с пунктом 2.1.3 (выбор уровня и количества пар карточек ограничен).

BEGIN

player_level_pkg.set_pairs('XX');

END;

Соответственно, где XX – количество пар карточек (от 5 до 30)

12. После выполнения всех пунктов методики испытаний, представленных выше для запуска игры требуется выполнить следующий блок:

BEGIN

player_level_pkg.start_game;

END;

13. Для выбора пары карточек в поле player_level_pkg.make_move(XX, XX) блока:

BEGIN

player_level_pkg.make_move('XX', 'XX');

END;

Вместо XX-вводятся номера пар карточек, которые пользователь хочет открыть.

14. Если пользователю необходимо проверить статус игры, то предусмотрено выполнение блока, но выполнение блока будет невозможно, если пользователь НЕ авторизовался и/или не задал **все** параметры игры:

```
BEGIN
    player_level_pkg.track_game_status;
END;
```

15. При необходимости принудительного выхода пользователь может выполнить блок player_level_pkg.exit_game.

16. Для просмотра топа игроков, пользователю следует выполнить блок:

```
BEGIN
    player_level_pkg.show_top;
END;
```

17. Для разлогирования из пользователя используется процедура

```
BEGIN
    player_level_pkg.logout;
END;
```

Для запуска игры другим игроком, который не является создателем игры, следует раздать права на исполнения пакета и вписать имя создателя с точкой в анонимных блоках:

1. От администратора (**каждая строка запускается по ОТДЕЛЬНОСТИ**):

```
GRANT SELECT, INSERT, UPDATE ON game_cards TO game_player;
GRANT SELECT ON player_results TO game_player;
CREATE USER с IDENTIFIED BY с;
GRANT CREATE SESSION TO с;
GRANT game_player TO с;
```

2. Подключаемся через «Новое соединение» к пользователю «с»

3. От пользователя с:

SET ROLE game_player; --применяем роль

BEGIN

developer.player_level_pkg.register_player('0', '12345678');

END; -- developer в данном случае создатель пакета

Далее все блоки запускаются с приставкой «developer.»*

*вместо developer имя администратора

5.2 Методы проверки Функциональных требований

Проверка требования «Регистрация и авторизация пользователя».

Проверка регистрации и авторизации пользователя проходит в соответствии с таблицей 1. Регистрация пользователя происходит с помощью процедуры register_player, авторизация – login. У обеих процедур 2 входных параметра – логин и пароль. Допустимые значения входных параметров, следующие:

1. Логин – не менее 1 символа и не более 20-ти;
2. Пароль – не менее 8 символов.

Разрешено использовать русский и английский языки, нижние подчеркивания и цифры. Пробелы и спец символы запрещены (допустимые значения: 'maxim546', '123123').

Проверка процедуры «Запуска игры»

Проверка процедуры запуска игры пользователем проходит в соответствии с таблицей 1. Для запуска игры пользователю необходимо произвести вход в систему в соответствии с требованиями игры: авторизация, выбор уровня игры, выбор количества пар карточек.

Проверка процедуры «Выбора уровня игры»

Проверка выбора уровня игры пользователем проходит в соответствии с таблицей 1. В игре 10 уровней сложности, от 1 до 10 включительно, процедура принимает 1 параметр уровня. Входной параметр для ввода уровня подразумевает ввод цифр от 1 до 10 включительно.

Проверка процедуры «Выбора количества пар карточек»

Проверка выбора пар карточек пользователем проходит в соответствии с таблицей 1. Пользователь имеет возможность выбрать от 5 до 30 пар карточек включительно.

Проверка процедуры «Выбора пар карточек во время игры»

Проверка выбора пар карточек во время игры пользователем проходит в соответствии с таблицей 1.

Входными параметрами могут являться: ('1', '6'), ('2', '3') и т.д..

Проверка процедуры «Статуса игры»

Проверка статуса игры пользователем проходит в соответствии с таблицей 1.

Проверка процедуры «Завершения игры»

Проверка завершения игры пользователем проходит в соответствии с таблицей 1.

Проверка процедуры «Таблицы результатов пользователей»

Проверка просмотра таблицы результатов пользователей пользователем проходит в соответствии с таблицей 1.

Таблица 1 – Состав входных параметров
и ожидаемых результатов

№ ис пы та ния	Состав входных параметров	Ожидаемый результат	Результат, полученны й в ходе испытания
Проверка процедуры регистрации(register_player)			
1	player_level_pkg.register_player('1333', '12345678');	Пользователь создан. Никнейм "1333" успешно зарегистрирован. Теперь войдите в игру.	
2	player_level_pkg.register_player(' ');	Ошибка «Никнейм не может быть пустым или состоять только из пробелов!»	
3	player_level_pkg.register_player(' ','');	Ошибка «Никнейм не должен содержать пробелы!»	
3	player_level_pkg.register_player('1333','');	Ошибка «Пароль должен быть не менее 8 символов!»	
4	player_level_pkg.register_player('1333',' '); *в пароле 8 пробелов	Ошибка «Пароль не должен содержать пробелы!»	
5	player_level_pkg.register_player('1333','12345');	Ошибка «Пароль должен быть не менее 8 символов!»	
6	player_level_pkg.register_player('012345678901234567891','12345678');	Ошибка «Никнейм должен быть от 1 до 20 символов»	
7	player_level_pkg.register_player('1','123456789012345678901345678901');	Пароль не должен превышать 30 символов!	
8	Вводим никнейм, который уже был зарегистрирован player_level_pkg.register_player('1333', '12345678');	Никнейм "1333" уже зарегистрирован. Выберите другой никнейм.	

9	Вводим никнейм, который содержит спец символы <code>player_level_pkg.register_player('№', '12345678');</code>	Никнейм может содержать только буквы (включая русские), цифры и подчеркивание!	
10	Пользователь авторизовался; выбрал уровень; выбрал количество пар карточек; начал играть, сделав ход и пробует зарегистрировать игрока	Игра уже началась! Нельзя выполнять это действие, пока не завершите игру с помощью и не разлогинитесь с помощью <code>logout</code>	
Проверка процедуры аутентификации(login)			
11	Заходим под пользователя, который зарегистрирован <code>player_level_pkg.login('1333', '12345678');</code>	Авторизация успешна для никнейма "1333".	
12	Заходим второй раз под пользователя, который зарегистрирован <code>player_level_pkg.login('1333', '12345678');</code>	Вы уже вошли в игру как "1333". Сначала выйдите из игры.	
13	Выходим из игры и ещё раз заходим <code>player_level_pkg.login('1333', '1234567890');</code>	Ошибка «Неверный пароль для никнейма "1333"»	
14	<code>player_level_pkg.login('kate', '123456780');</code>	Ошибка «Никнейм "kate" не зарегистрирован. Сначала зарегистрируйтесь» .	
15	Пользователь авторизовался; выбрал уровень; выбрал количество пар карточек; начал играть, сделав ход и пробует залогиниться	Игра уже началась! Нельзя выполнять это действие, пока не завершите игру с помощью <code>logout</code> .	
Проверка процедуры выбора уровня игры(set_level)			

16	Пользователь НЕ авторизовался <u>player_level_pkg.set_level('2');</u>	Сначала войдите в игру с помощью никнейма и пароля	
17	Пользователь авторизован <u>player_level_pkg.set_level('10');</u>	Уровень установлен: 10	
18	Пользователь авторизован <u>player_level_pkg.set_level('100');</u>	Уровень должен быть от 1 до 10	
19	Пользователь авторизован <u>player_level_pkg.set_level('dd');</u>	Введено некорректное значение! Уровень должен быть целым числом от 1 до 10.	
20	Пользователь авторизовался; выбрал уровень; выбрал количество пар карточек; начал играть, сделав ход и пробует установить уровень	Игра уже началась! Нельзя выполнять это действие, пока не завершите игру с помощью <code>exit_game</code> .	
Проверка процедуры выбора количества пар карточек(<code>set_pairs</code>)			
21	Пользователь НЕ авторизовался <u>player_level_pkg.set_pairs('10');</u>	Сначала войдите в игру с помощью никнейма и пароля	
22	Пользователь авторизовался <u>player_level_pkg.set_pairs('10');</u>	Количество пар карт установлено: 10	
23	Пользователь авторизовался <u>player_level_pkg.set_pairs('100');</u>	Количество пар должно быть от 5 до 30!	
24	Пользователь авторизовался <u>player_level_pkg.set_pairs('dd');</u>	Введено некорректное значение! Количество пар должно быть целым числом от 5 до 30.	
25	Пользователь авторизовался; выбрал уровень; выбрал количество пар карточек; начал играть, сделав ход и пробует установить уровень и пробует установить новое количество карточек	Игра уже началась! Нельзя выполнять это действие, пока не завершите игру с помощью <code>exit_game</code> .	

Проверка процедуры запуска игры(start_game)			
26	Пользователь авторизовался; выбрал уровень; выбрал количество пар карточек player_level_pkg.start_game;	Игровое поле создано. Уровень: XX, время на ход: XX сек Игровое поле (Уровень: XX, время на ход: XX сек): И появится само поле	
27	Пользователь НЕ авторизовался player_level_pkg.start_game;	Сначала войдите в игру с помощью никнейма и пароля	
28	Пользователь авторизовался, но НЕ выбрал уровень player_level_pkg.start_game;	Уровень не установлен. Задайте уровень с помощью set_level.	
29	Пользователь авторизовался; выбрал уровень, но выбрал количество пар карточек player_level_pkg.start_game;	Количество пар не установлено. Задайте количество пар с помощью set_pairs.	
30	Пользователь авторизовался; выбрал уровень; выбрал количество пар карточек; начал играть, сделав ход player_level_pkg.start_game;	Игровое поле создано (игра перезапускается, генерируется новое поле, где карточки под другими значениями)	
Проверка процедуры выбора пары карточек во время игры (make_move)			
31	Пользователь не авторизован player_level_pkg.make_move ('2', '4');	Сначала войдите в игру с помощью никнейма и пароля	
32	Пользователь авторизован, но не выбрал уровень/количество пар карт/не начал игру player_level_pkg.make_move ('2', '4');	Вы не можете сделать ход, пока не начнете игру!	
33	Пользователь авторизован и начал игру player_level_pkg.make_move	Карты не совпали. Закрываем обратно / Карты совпали!	

	(2', '4');		
33	Пользователь авторизован и начал игру <code>player_level_pkg.make_move('1','1')</code> ;	Карточки должны быть разными!	
34	Пользователь авторизован, начал игру и открыл карты, которые уже были открыты ранее <code>player_level_pkg.make_move('1','6');</code>	Одна или обе карты уже открыты!	
35	Пользователь авторизован, начал игру и сделал ход, когда время на ход истекло <code>player_level_pkg.make_move('1','6');</code>	Время на ход (15 сек) истекло!	
36	Пользователь авторизован, начал игру и сделал ход <code>player_level_pkg.make_move('1','d')</code> ;	Принимаются только числа в качестве ID карт!	
37	Пользователь авторизован, начал игру и сделал ход <code>player_level_pkg.make_move('1','31');</code> *но количество пар карточек до 30	ID карт должны быть от 1 до 30.	
Проверка процедуры статуса игры (track_game_status)			
38	Пользователь хочет проверить статус игры, НО не авторизовался <code>player_level_pkg.track_game_status;</code>	Вы не вошли в игру! Сначала выполните <code>login()</code> .	
39	Пользователь хочет проверить статус игры, НО не выбрал уровень игры	Уровень не установлен!	
40	Пользователь хочет проверить статус игры, НО не выбрал количество пар карточек	Количество пар карточек не установлено! Сначала выполните <code>set_pairs()</code>	
41	Пользователь авторизовался, выбрал количество пар карточек и уровень, но не произвел запуск игры	Игра еще не начата! Сделайте первый ход. Уровень: XX Количество пар карт: XX	

42	При вызове процедуры статуса игры track_game_status пользователем, с заранее всеми введенными входными параметрами и запуском игры	Игрок: YYY Уровень: XX Количество пар карт: XX Время на ход: XX сек	
Проверка процедуры завершения игры(exit_game)			
43	При запуске функции в любой момент player_level_pkg.exit_game;	Вы вышли из игры.	
Проверка процедуры просмотра таблицы результатов пользователей(show_top)			
44	При запуске функции в любой момент player_level_pkg.show_top;	Топ 20 игроков по очкам: (выводится таблица(если нет зарегистрированных пользователей, то таблица пустая))	
45	Игрок YY выиграл игру с XX количеством пар карточек	Топ 20 игроков по очкам: (выводится таблица с увеличенным на XX количеством очков у игрока YY)	
46	Игрок проиграл игру с любым количеством пар карточек	Топ 20 игроков по очкам: (выводится таблица без изменений очков игрока)	
Проверка процедуры выхода из пользователя logout			
47	В любой момент времени игры player_level_pkg.logout;	Вы разлогинились из системы. Все данные авторизации сброшены. Для продолжения игры необходимо заново войти с помощью login.	
Проверка многопользовательского режима			

48	Оба пользователя вошли, задали количество пар карточек и уровень и применяют player_level_pkg.start_game	У обоих пользователей: Игровое поле создано. Уровень: XX, время на ход: XX сек Игровое поле (Уровень: XX, время на ход: XX сек): И появится само поле	
49	Оба пользователя вошли, задали количество пар карточек и уровень и применяют player_level_pkg.track_game_statuses	Появляется статус игры у каждого игрока свой, со своим именем, уровнем, количеством пар и остаточным временем.	