# Оглавление

[Оглавление 1](#_Toc151290465)

[Рейтинг 2](#_Toc151290466)

[Требования к отчетам по ЛБ 2](#_Toc151290467)

[Вариант 1. 3](#_Toc151290468)

[Вариант 2. 3](#_Toc151290469)

[Вариант 3. 4](#_Toc151290470)

[Вариант 4. 5](#_Toc151290471)

[Вариант 5. 6](#_Toc151290472)

[Вариант 6. 7](#_Toc151290473)

[Вариант 7. 8](#_Toc151290474)

[Вариант 8. 9](#_Toc151290475)

[Вариант 9. 10](#_Toc151290476)

[Вариант 10. 11](#_Toc151290477)

**ВАЖНО!**

* Проверка исходных данных на корректное значение при решении всех задач **обязательна**!
* Везде, где исходные значения не заданы явно, **предусмотреть возможность их изменения** пользователем в процессе тестирования.
* По умолчанию, вывод результата должен производиться **через консоль, либо через элементы на веб-странице**.
* Предпочтительно при выполнении ЛБ использовать **только фронтенд-технологии** (для хранения данных можно воспользоваться встроенными возможностями браузера)
* Названия файлов отчётов, загружаемых в электронный курс, **должны быть на русском языке** **и** **соответствовать шаблону**: **ГГГГ\_ФИО\_№№**, где ГГГГ – номер группы, ФИО – фамилия и инициалы студента, №№ – номер работы. **Файлы с именами, отличающимися от вышеприведённого шаблона, приниматься НЕ будут, и баллы за них также начислены не будут!**

## Рейтинг

Необходимо выполнить 8 ЛБ согласно заданию по своему варианту.

ЛБ оцениваются максимум в 10 баллов.

Варианты заданий выдаются по совпадению последней цифры варианта с последней цифрой номера студента в электронной ведомости.

Общая итоговая сумма баллов, полученная за ЛБ течение семестра, зависит от того, как сдавались работы (баллы могут быть снижены в процессе приёма при наличии недочётов) и соблюдения сроков сдачи (см. таблицы, в них приводится коэффициент успеваемости, на который будут умножаться баллы, полученные за каждую ЛБ).

**ОБРАТИТЕ ВНИМАНИЕ! Сдача любой ЛБ на «полные баллы» возможна только при условии, что для сдачи ЛБ использовалось не более одной попытки (за каждую последующую попытку баллы могут быть снижены)!**

Коэффициенты успеваемости для ЛБ №№ 1-8 (в зависимости от срока сдачи):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ЛБ\неделя | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
| **1** | 1 | 1 | 0,9 | 0,9 | 0,8 | 0,8 | 0,7 | 0,7 | 0,6 | 0,6 | 0,5 | 0,5 | 0,4 | 0,4 | 0,3 | 0,3 |
| **2** | 1 | 1 | 1 | 1 | 0,9 | 0,9 | 0,8 | 0,8 | 0,7 | 0,7 | 0,6 | 0,6 | 0,5 | 0,5 | 0,4 | 0,4 |
| **3** | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,9 | 0,8 | 0,8 | 0,7 | 0,7 | 0,6 | 0,6 | 0,5 | 0,5 |
| **4** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,9 | 0,8 | 0,8 | 0,7 | 0,7 | 0,6 | 0,6 |
| **5** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,9 | 0,8 | 0,8 | 0,7 | 0,7 |
| **6** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,9 | 0,8 | 0,8 |
| **7** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,9 |
| **8** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

## Требования к отчетам по ЛБ

1. Шрифт – Times New Roman, 14 пт.
2. Интервалы: междустрочный – 1,5 строки, интервал до и после абзаца – 0 пт.
3. Отступ первой строки – 1,25
4. Рисунки и подписи к ним выравниваются по центру.

**Требования к структуре отчета:**

1. Титульный лист
2. Задание
3. Основная часть: код и результат работы скриптов (скриншоты, описание реализации)
4. Выводы по работе

## Вариант 1.

Реализуйте проект “TODO - список дел”.

“TODO” - простое в использовании приложение, благодаря которому можно составлять списки дел (заметок), зачеркивать готовые и упорядочивать их посредством блокнотов, которые можно также создать для конкретных событий или дел.

Под свой вкус можно выбрать шрифт и размер текста. На главной странице будут располагаться наши блокноты с указанием времени их создания. Новые блокноты можно создать, нажав на «плюсик» в верхнем углу экрана. В открывшемся окне сразу же нам нужно написать заголовок будущего блокнота, а после этого начать записывать свои дела.

Лабораторные работы по заданию:

1. Реализовать возможность создавать блокноты и списки дел в них.
2. Реализовать возможность отмечать выполненные дела, удалять и редактировать заметки и блокноты.
3. Реализовать возможность сохранять все данные после обновления страницы.
4. Реализовать возможность окрашивать блокноты в разные цвета (в зависимости от приоритета).
5. Реализовать возможность осуществлять поиск по заметкам
6. Реализовать возможность устанавливать приоритет дел в списке
7. Реализовать возможность сортировать дела по приоритету, названию, дате
8. Реализовать анимацию интерфейса

## Вариант 2.

Разработайте игру “Сапер”.

Идея игры такая: на игровом поле M\*N где-то спрятаны мины. Игрок кликает по клеткам, открывая их. Если в клетке была мина, игрок проиграл. Если нет, то в клетке выводится цифра, показывающая общее число мин в соседних клетках. Если игрок открыл все клетки, кроме заминированных, он победил. Если игрок открывает клетку, рядом с которой нет мин, то все соседние клетки открываются автоматически (если на них тоже нет мин, то процесс продолжается).

Правой кнопкой мыши на неоткрытых клетках можно расставлять флажки.

Надпись «Вы победили» или «Вы проиграли» должна выводиться в окошке поверх игрового поля и содержать кнопку «Новая игра».

При реализации игры необходимо предусмотреть три варианта экрана:

* Стартовый экран - содержит наименование игры, поля ввода своего имени, поля с вводом размеров минного поля, количества мин, и значения таймера, а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.
* Основной экран игры - интерфейс должен содержать:
* блок с таймером - содержит таймер обратного отсчета с указанием минут и секунд в формате мм:сс, кнопку «Пауза»
* игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.
* Экран окончания игры - содержит результат игрока, и кнопку, позволяющую начать заново.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация визуализации результатов обработки событий нажатия правой/левой кнопок мыши.
4. Реализация алгоритмов выигрыша / проигрыша (по таймеру, либо по попадании на мину).
5. Реализация алгоритма рекурсивного открытия незаминированных клеток.
6. Реализация поддержки таблицы игроков с запоминанием результатов.
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 3.

Требуется разработать игру «Космический трубопровод».

При столкновении с космическим телом на Международной космической станции произошел процесс неконтролируемой разгерметизации. Требуется срочно восстановить разрушенную систему подачи воздуха. Для этого были направлены специалисты по монтажу систем вентиляции. Перед ними поставлена задача, восстановить систему воздуховода на поврежденном участке, соединив нужные участки труб между началом и концом разрыва (начало и конец разрыва находятся на противоположных сторонах).

Общие требования:

Начало и конец трубы находятся в верхнем левом и нижнем правом углах игрового поля соответственно. Клетки поля заполняются случайным образом изображениями фрагментов трубы. При каждом щелчке игрока по изображению фрагмент поворачивается на 90 градусов. Возможен перенос фрагментов трубы по игровому полю. Игра заканчивается либо с построением правильного пути, либо с окончанием времени таймера (проигрыш).

При реализации игры необходимо предусмотреть три варианта экрана:

* Стартовый экран - содержит наименование игры, поле ввода своего имени, поле с вводом значения таймера, а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.
* Основной экран игры - интерфейс должен содержать:
  + блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс.
  + игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.
* Экран окончания игры - содержит результат игрока, и кнопку, позволяющую начать заново.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация расстановки фрагментов трубы на игровом поле.
4. Реализация обработки перетаскивания фрагментов трубы по игровому полю.
5. Реализация обработки поворотов фрагментов трубы.
6. Реализация логики проверки работоспособности (соединения начала и конца) трубопровода.
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 4.

Необходимо разработать игру по типу морского боя.

Игра представляет собой два поля, разделенных на квадратные области, в которых находятся фрагменты «кораблей». В одном из полей находятся «корабли» игрока, а в другом – скрытые «корабли» противника (компьютера). Длина кораблей может варьироваться, их общее количество неизменно. После размещения «кораблей» в полях, их вращение или изменение местоположения – запрещено. Задача игрока – «потопить» все «корабли» противника до полной потери собственных. Игрок может сделать только один ход подряд, следующий ход возможен после хода компьютера.

Стартовый экран - содержит наименование игры, поле ввода своего имени, а также кнопку “Начать игру”. Имя не может быть пустым. Кнопка “Начать игру” активна, если только введено имя пользователя.

Основной экран игры - интерфейс должен содержать:

* блок с таймером - содержит таймер продолжительности игры с указанием минут, секунд в формате мм:сс. Таймер должен обновляться без перезагрузки страницы. В этом же блоке располагается кнопка “Пауза”
* блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
* игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.

Экран окончания игры - содержит сообщение о том, кто и с каким счетом выиграл, и переход на стартовый экран.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация расстановки кораблей игрока
4. Реализация расстановки кораблей компьютера
5. Реализация игрового процесса (выбора ячеек игроком и компьютером, и обработка событий, связанных с попаданием/промахом).
6. Реализовать анимацию «подбитых / затопленных кораблей».
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 5.

Необходимо разработать браузерную игру «Сбор фруктов».

Игровое поле представляет собой сад с падающими фруктами. В определённой области через заданный промежуток времени появляется какой-либо фрукт. Игрок должен набрать максимальное количество баллов, кликая на такие же фрукты. После клика по нужному фрукту, он исчезает, а игрок получает очки. Если игрок кликнул не на тот фрукт, то фрукт исчезает, а игрок теряет очки. Каждый фрукт движется со случайной скоростью сверху вниз. Если по фрукту не кликнули, и он достиг нижней части игрового поля, то он исчезает. Одновременно на одном игровом поле может находиться не более N фруктов (задаётся через специальное поле).

Стартовый экран - содержит наименование игры, поля ввода имени игрока, ввода максимально возможного одновременного количества фруктов на игровом поле, ввода значения таймера обратного отсчёта (со значениями по умолчанию), а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.

На стартовом экране представлена инструкция к игре.

Основной экран игры - интерфейс должен содержать:

* блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс. Таймер должен обновляться без перезагрузки страницы. В этом же блоке располагается кнопка “Пауза”.
* блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
* блок с именем игрока.
* блок, в котором отображается фрукт, по которым нужно кликать в данный момент времени.
* игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.

Экран окончания игры - содержит набранное количество очков и переход на стартовый экран.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация логики появления, «падения» и исчезновения 1 фрукта.
4. Реализация логики появления, «падения» и исчезновения до N фруктов (с разной скоростью).
5. Реализация логики клика по нужному и ошибочному фрукту.
6. Реализация статистики (сколько каких фруктов упало / было собрано).
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 6.

Необходимо разработать онлайн-игру (аркаду) – «Гонки».

Описание игры

Участнику необходимо реализовать ограниченное по размерам игровое поле, на котором в нижней части передвигается машина (влево или вправо). В верхней части поля появляются и двигаются вниз объекты, которые являются препятствиями или бонусами в игре. Скорость перемещения объектов увеличивается со временем игры. Задача игрока – продержаться в игре как можно больше времени.

Стартовый экран - содержит наименование игры, поля ввода своего имени, ввода количества бонусов для сбора, ввода количества жизней, ввода значения таймера обратного отсчёта (со значениями по умолчанию), а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.

Основной экран игры - интерфейс должен содержать:

* блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс. Таймер должен обновляться без перезагрузки страницы. В этом же блоке располагается кнопка “Пауза”
* блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
* блок с именем игрока - если игрок на стартовом экране в имени ввел “tester” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
* игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.

Экран окончания игры - содержит результат игрока и переход на стартовый экран.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация логики появления этих объектов (смены фона)
4. Реализация статистики (сколько бонусов / препятствий и какого типа было собрано).
5. Обработка соответствующих событий - уменьшение количества “жизней”, конец игры, проигрыш, выигрыш при наступлении определенных действий (сбор нужного количества бонусов).
6. Реализация режима теста – режима, в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры.
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле. Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры также возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 7.

Вам необходимо разработать онлайн игру «Космический симулятор».

Цель игры – сбить как можно больше пролетающих метеоритов, не допустив при этом столкновения с кораблем. Поле имеет прямоугольную форму, корабль располагается внизу и может двигаться влево или вправо. Метеориты появляются сверху (в случайном порядке) и могут двигаться под произвольным углом к нижней границе игрового поля. Корабль имеет функцию выстрела, которая уничтожает первый метеорит на текущей вертикали. Количество выстрелов неограниченно. Если метеорит попадает в корабль – игра заканчивается. За сбитый выстрелом метеорит начисляются очки, за «пропущенный» метеорит (объект дошел до нижней горизонтали) очки вычитаются.

При реализации игры необходимо предусмотреть три варианта экрана:

* Стартовый экран - содержит наименование игры, поле ввода своего имени, поле с вводом значения таймера, а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.
* Основной экран игры - интерфейс должен содержать:
  + блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс.
  + блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
  + блок с именем игрока - если игрок на стартовом экране в имени ввел “tester” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
  + игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.
* Экран окончания игры - содержит результат игрока, и кнопку, позволяющую начать заново.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация появления и движения метеоритов.
4. Реализация движения корабля и его выстрелов.
5. Реализация статистики «сбито / пропущено».
6. Реализация режима теста – режима, в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры.
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 8.

Реализуйте игру “Рыбки”.

Существует аквариум, в котором плавают рыбки. Каждую секунду добавляются новые рыбки, до определенного максимума заполняемости аквариума. При клике на рыбку, она исчезает, игроку начисляются некоторое количество баллов (в зависимости от размера рыбки – чем она меньше, тем больше баллов). Задача - набрать максимальное количество баллов за заданный период времени MM:SS.

* Стартовый экран - содержит наименование игры, поле ввода своего имени, поле с вводом значения таймера, а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.
* Основной экран игры - интерфейс должен содержать:
* блок с таймером - содержит таймер обратного отсчета с указанием минут и секунд в формате мм:сс, кнопку «Пауза»
* игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.
* Экран окончания игры - содержит результат игрока, и кнопку, позволяющую начать заново.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация логики появления рыбок разных размеров.
4. Реализация логики движения рыбок (влево-вправо, вверх-вниз).
5. Реализация логики по обработке клика на рыбку (она исчезает, а игроку начисляются очки).
6. Реализация поддержки таблицы игроков с запоминанием результатов.
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 9.

Необходимо разработать игру «Пенальти».

Цель игры – забить гол в ворота соперника (мяч находится внизу игрового поля, в середине). Игрок выбирает область ворот, в которую будет осуществлен удар, после чего появляется показатель сила удара, меняющийся от максимального к минимальному каждые 300 мс. Во время повторного клика сила удара фиксируется, и мяч перемещается в указанную область. В створе ворот должен размещаться вратарь команды противника, который перемещается от левой границы ворот к правой каждые 1000 мс. Если при ударе по воротам мяч попадает во вратаря или в зону вокруг него, равную половине ширины вратаря, вратарь ловит мяч и пенальти считается проигранным. В обратном случае, гол и победа засчитывается за игроком. Время на пенальти ограничено таймером. В случае трех голов подряд в игре должен появляться бонус – ускорение мяча при ударе в два раза. С каждым голом скорость перемещения вратаря уменьшается на 10 мс.

При реализации игры необходимо предусмотреть три варианта экрана:

* Стартовый экран - содержит наименование игры, поле ввода своего имени, поле с вводом значения таймера, а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.
* Основной экран игры - интерфейс должен содержать:
  + блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс.
  + блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
  + блок с именем игрока - если игрок на стартовом экране в имени ввел “tester” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
  + игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.
* Экран окончания игры - содержит результат игрока, и кнопку, позволяющую начать заново.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация перемещения вратаря.
4. Реализация выбора траектории мяча и его перемещения.
5. Реализация оставшейся игровой логики (см. описание).
6. Реализация режима теста – режима, в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры.
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.

## Вариант 10.

Необходимо разработать онлайн игру «Арканоид».

Цель игры – очистить поле от появляющихся сверху в виде стопки прямоугольных блоков разных размеров, попадая по ним шариком, который отскакивает от «ракетки», которую можно передвигать вправо и влево (за это начисляются баллы). При падении на поверхность (ракетку, стену, плиту) шарик отражается под углом в 90 градусов. Уничтожение плиты происходит только при попадании шариком на её нижнюю поверхность (в противном случае происходит отражение). Если шарик «упал на пол», уменьшается количество попыток (когда они заканчиваются, заканчивается и игра).

При реализации игры необходимо предусмотреть три варианта экрана:

* Стартовый экран - содержит наименование игры, поле ввода своего имени, поле с вводом значения таймера, а также кнопку “Начать игру”. Поля не могут быть пустыми. Кнопка “Начать игру” активна, если только заполнены поля.
* Основной экран игры - интерфейс должен содержать:
  + блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс.
  + блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
  + блок с именем игрока - если игрок на стартовом экране в имени ввел “tester” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
  + игровое поле - процесс игры должен включать в себя взаимодействие пользователя и компьютера и протекать без обновления страницы.
* Экран окончания игры - содержит результат игрока, и кнопку, позволяющую начать заново.

Лабораторные работы по заданию:

1. Реализация экранов игры.
2. Реализация изображений возможных объектов на основном экране игры.
3. Реализация появления и движения стопки плит разных размеров.
4. Реализация движения «ракетки»
5. Реализация движения шарика и обработки его действий.
6. Реализация режима теста – режима, в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры.
7. Реализовать таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение MM:SS. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
8. Пауза игрового процесса (по нажатию на кнопку “Пауза”) - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.