## Проверочная работа №3

Выполните предложенное ниже задание согласно своему варианту, составьте **отчет** с **блок-схемами** или **словесным** описанием всех алгоритмов, **кодом** и **результатами работы**.

### Варианты

**Вариант 1**

Разработайте игру “Сапер”. Идея игры такая: на игровом поле 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 8 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:

1. стартовый экран - содержит наименование игры, поле ввода своего имени, а также кнопку “Начать игру”. Кнопка “Начать игру” активна, если введено имя пользователя.
2. экран игры - интерфейс должен содержать:
   1. блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс.
   2. блок с заработанными баллами (количество оставшихся секунд по таймеру) - количество заработанных игроком баллов за текущую игру.
   3. блок с руководством по игре.
3. экран окончания игры - содержит результат игрока и кнопку, позволяющую начать заново.

**Вариант 2**

Вам необходимо разработать онлайн игру (аркаду) – «Гонки». Участнику необходимо реализовать ограниченное по размерам игровое поле, на котором в нижней части передвигается машина. В верхней части поля располагаются 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, поле ввода своего имени, а также кнопку “Начать игру”. Имя не может быть пустым. Кнопка “Начать игру” активна, если только введено имя пользователя.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. блок с именем игрока - если игрок на стартовом экране в имени ввел “terter” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
4. игровое поле - в зависимости от выбранной экспертами игры. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит результат игрока.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле, перечень элементов, для которых может сохранять анимацию, описывается экспертами отдельно (например, перемещение фона в режиме паузы). Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, как только доходит до значения 00:00 игра заканчивается
3. Режима теста - режим в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры
4. Реализация логики работы игрового поля, например, изменение количества собранных объектов, уменьшение количества “жизней”, уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - конец игры, проигрыш, выигрыш при наступлении определенных действий.

**Вариант 3**

С помощью чистого js реализуйте игру “Рыбки”.

**Общее описание игры.**

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. Задача, набрать максимальное количество баллов за минуту.

**Минимальный функционал программы:**

1. ввод имени игроков перед началом игры
2. по клику на рыбку она исчезает, а игроку начисляются очки
3. рыбки двигаются
4. игра заканчивается с окончанием времени

**Дополнительный функционал программы:**

1. присутствуют несколько видов рыб разных размеров и видов, при клике на каждую начисляется разное количество очков
2. присутствуют анимации появления и исчезновения рыб, анимации таймера, элементов интерфейса
3. после окончания игры появляется таблица рекордов, в которой показан текущий результат игрока и результаты лучших 10 попыток.

**Вариант 4**

С помощью чистого js реализуйте проект “Todo - список дел”. Составьте отчет по выполненной работе.

“Todo” очень 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.

**Минимальный функционал программы:**

1. присутствует разработанный дизайн и адаптивная верстка
2. можно создавать блокноты и списки дел в них
3. можно отмечать выполненные дела, удалять и редактировать заметки и блокноты
4. все данные сохраняются после обновления страницы

**Дополнительный функционал программы:**

1. можно окрашивать блокноты в разные цвета
2. можно осуществлять поиск по заметкам
3. устанавливать приоритет дел в списке
4. сортировать дела по приоритету, названию, дате
5. присутствует анимация интерфейса

**Вариант 5**

Требуется разработать игру «Space pipes».

**Сюжет:**

*При столкновении с космическим телом на Международной космической станции «Марс-2» в секторе «А» произошел процесс неконтролируемой разгерметизации.*

*Требуется срочно восстановить разрушенную систему подачи воздуха. Для этого в сектор «А» 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 соответственно. Клетки поля заполняются случайным образом изображениями фрагментов трубы. При каждом щелчке игрока по изображению фрагмент поворачивается на 90 градусов Игра заканчивается либо с построением правильного пути, либо с окончанием времени таймера.

При реализации игры необходимо предусмотреть три варианта экрана:

1. стартовый экран - содержит наименование игры, поле ввода своего имени, а также кнопку “Начать игру”. Кнопка “Начать игру” активна, если введено имя пользователя.
2. экран игры - интерфейс должен содержать:
   1. блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс.
   2. блок с заработанными баллами (количество оставшихся секунд по таймеру) - количество заработанных игроком баллов за текущую игру.
   3. блок с руководством по игре.
3. экран окончания игры - содержит результат игрока и кнопку, позволяющую начать заново.

**Вариант 6**

Разработайте электронный вариант игры “Быки и коровы”.

“Быки и коровы” — логическая игра, в ходе которой за несколько попыток один из игроков должен определить, что задумал другой игрок.

Каждый из игроков задумывает и записывает тайное 4-значное число с неповторяющимися цифрами. Игрок, который начинает игру по жребию, делает первую попытку отгадать число. Попытка — это 4-значное число с неповторяющимися цифрами, сообщаемое противнику. Противник сообщает в ответ, сколько цифр угадано без совпадения с их позициями в тайном числе (то есть количество коров) и сколько угадано вплоть до позиции в тайном числе (то есть количество быков). Например:

Задумано тайное число «3219».

Попытка: «2310».

Результат: две «коровы» (две цифры: «2» и «3» — угаданы на неверных позициях) и один «бык» (одна цифра «1» угадана вплоть до позиции).

Игроки делают попытки угадать по очереди. Побеждает тот, кто угадает число первым, при условии, что он не начинал игру. Если же отгадавший начинал игру — его противнику предоставляется последний шанс угадать последовательность.

При игре против компьютера игрок вводит комбинации одну за другой, пока не отгадает всю последовательность.

При реализации игры необходимо предусмотреть три варианта экрана:

* 1. стартовый экран - содержит наименование игры, поле ввода своего имени, а также кнопку “Начать игру”. Кнопка “Начать игру” активна, если введено имя пользователя.
  2. экран игры - интерфейс должен содержать:
  3. блок с таймером - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс.
  4. блок с заработанными баллами (количество оставшихся секунд по таймеру) - количество заработанных игроком баллов за текущую игру.
  5. блок с руководством по игре.
  6. экран окончания игры - содержит результат игрока и кнопку, позволяющую начать заново.

**Вариант 7**

К вам обратилась компания по разработке игр для веб-сайтов. Компания

просит помочь в разработке веб-игры.

**Описание игры:**

При открытии игры должен быть отображен **экран входа в игру**. На

данном экране присутствует поле для указания вашего имени и кнопка входа.

Если имя пользователя не указано, то кнопка должна быть не активна.

Когда имя пользователя будет заполнено кнопка должна стать активной.

При клике на кнопку входа экран должен смениться на игровой.

На **игровом экране** находиться имя пользователя, которое должно

корректно отображаться, секундомер (начинает считать с 00:00), который

должен быть запущен при старте игры, счетчик жизней (при старте у игрока

есть 3 жизни) и счетчик пойманных предметов.

На игровом поле есть корзина, в которую нужно ловить падающие

фрукты.

При старте игры корзина должна находиться снизу по центру.

В процессе игры должны появляться фрукты (1 фрукт каждую секунду).

Фрукты должны падать вниз с разной скоростью.

Игрок должен иметь возможность управлять корзиной с помощью

стрелок влево и вправо.

Корзина не должна выходить за пределы игровой зоны.

Если фрукт достигает земли (нижняя граница экрана), то у пользователя

отнимаются жизни.

Если удерживать «пробел», то все фрукты, присутствующие на экране

будут считаться собранными. Этот процесс может быть активен не дольше 4

секунд. После этого нужно 5 секунд на восстановление.

Постарайтесь отобразить процесс «автоматического сбора».

Должна быть возможность поставить игру на паузу нажав кнопку ESC.

Повторное нажатие должно продолжить игру. Во время паузы все

интерактивные действия (анимация, секундомер, фрукты, корзина) должны

быть приостановлены.

Когда жизни будут равны нулю, то игра должна закончиться и должен

отобразиться экран с результатами.

На **экране с результатами** необходимо отобразить время, которое игрок

продержался в игре и кол-во собранных фруктов. Если игрок продержался

дольше 10 секунд, то должно отобразиться сообщение о выигрыше, иначе о

проигрыше.

На экране с результатами есть кнопка «Играть сначала» при клике на

которую игра должна перезапуститься.

**Вариант 8**

Вам необходимо разработать онлайн игру Арканоид. Цель игры – очистить поле от прямоугольных блоков, попадая по ним шариком, который отскакивает от «ракетки», который можно передвигать вправо и влево. Если шарик «упал на пол», уменьшается количество жизней. Когда жизни заканчиваются, заканчивается и игра..

**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.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. блок с именем игрока - если игрок на стартовом экране в имени ввел “terter” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
4. игровое поле - в зависимости от выбранной экспертами игры. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит таблицу результатов с первыми 10 лучшими результатами. Если игрок по баллам не вошел в 10 лучших, то вместо 10 места показывается его результат с указанием места в таблице рекордов.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле, перечень элементов, для которых может сохранять анимация описывается экспертами отдельно (например, перемещение фона в режиме паузы). Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, как только доходит до значения 00:00 игра заканчивается
3. Режима теста - режим в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры
4. Реализация логики работы игрового поля, например, изменение количества собранных объектов, уменьшение количества “жизней”, уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - конец игры, проигрыш, выигрыш при наступлении определенных действий.

**Вариант 9**

Вам необходимо разработать онлайн игру “Сбей самолет”. По экрану летят вражеские самолеты. Цель играющего – сбить их. Пусковая установка находится в нижней строке экрана. Пусковую установку можно перемещать по строке вперед и назад. Вражеские самолеты при полете сбрасывают бомбы, при попадании которых в пусковую установку ей наносится ущерб (уменьшается "здоровье" установки). Необходимо сбить максимальное количество самолетов за выделенное время.

**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.

**Экран игры** - интерфейс должен содержать:

1. 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»
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. 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.
4. игровое поле - в соответствии с описанием игры. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит результат игрока.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. 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 так же возможно по нажатию на клавишу «пробел» или по нажатию на кнопку «Пауза» на игровом поле
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, как только доходит до значения 00:00 игра заканчивается
3. Сохранение итоговых результатов в базу данных на стороне сервера. Участнику предоставляется готовая структура базы данных(дамп таблицы), с которой он должен работать. Изменять структуру нельзя.
4. Режима теста - режим в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры
5. Реализация логики работы игрового поля, например, изменение количества сбитых самолетов, уменьшение количества «здоровья пусковой установки», уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - конец игры, проигрыш, выигрыш при наступлении определенных действий.

**Вариант 10**

Вам необходимо разработать онлайн игру «StarTrek».

Цель игры – провести космический корабль через поле метеоритов. Если метеорит попал в корабль - уменьшается количество жизней. Когда жизни заканчиваются, заканчивается и игра.

**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.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. 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.
4. игровое поле - в зависимости от выбранной экспертами игры. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит результат игрока.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле, перечень элементов, для которых может сохранять анимацию, описывается экспертами отдельно (например, перемещение фона в режиме паузы). Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, как только доходит до значения 00:00 игра заканчивается
3. Таблица рекордов - показывает 10 лучших результатов игры. Если игрок не вошел в 10 лучших его результат показывает вместо 10 результата, с указанием его места в таблице рекордов.
4. Режима теста - режим в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры
5. Реализация логики работы игрового поля, например, изменение количества собранных объектов, уменьшение количества “жизней”, уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - конец игры, проигрыш, выигрыш при наступлении определенных действий.

**Вариант 11**

В данном задании участнику необходимо разработать игру «Пенальти», для футбольных болельщиков. Цель игры – забить гол в ворота соперника. 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.

**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.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. 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.
4. игровое поле - в зависимости от выбранной экспертами игры. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит результат игрока.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле, перечень элементов, для которых может сохранять анимация описывается экспертами отдельно (например, перемещение фона в режиме паузы). Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, как только доходит до значения 00:00 игра заканчивается
3. Режима теста - режим в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры
4. Реализация логики работы игрового поля, например, изменение количества собранных объектов, уменьшение количества “жизней”, уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - конец игры, проигрыш, выигрыш при наступлении определенных действий.

**Вариант 12**

Вам необходимо разработать онлайн игру «SpaceShip» (космический симулятор).

Цель игры – сбить как можно больше пролетающих мимо метеоритов, не допустив при этом столкновения с кораблем. Поле имеет прямоугольную форму, корабль может двигаться влево и вправо. Метеориты 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.

**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.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. 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.
4. игровое поле - в зависимости от выбранной экспертами игры. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит результат игрока.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле, перечень элементов, для которых может сохранять анимацию, описывается экспертами отдельно (например, перемещение фона в режиме паузы). Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, как только доходит до значения 00:00 игра заканчивается
3. Реализация логики работы игрового поля, например, изменение количества собранных объектов, уменьшение количества “жизней”, уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - конец игры, проигрыш, выигрыш при наступлении определенных действий.

**Вариант 13**

Вам 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). 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.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит сообщение о том кто и с каким счетом выиграл.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле, перечень элементов, для которых может сохранять анимация описывается экспертами отдельно (например, перемещение фона в режиме паузы). Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
2. Таймер продолжительности игрового процесса - начинает счет с началом игры, останавливается, когда заканчивается игра.
3. Реализация логики работы игрового поля, например, расстановка «кораблей», способ выбора целей, и т.д.

**Вариант 14**

Вам необходимо разработать онлайн игру (аркаду) – «Гонки». Участнику необходимо реализовать ограниченное по размерам игровое поле, на котором в нижней части передвигается машина. В верхней части поля располагаются 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, поле ввода своего имени, а также кнопку “Начать игру”. Имя не может быть пустым. Кнопка “Начать игру” активна, если только введено имя пользователя.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. блок с именем игрока - если игрок на стартовом экране в имени ввел “terter” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
4. игровое поле - в зависимости от выбранной экспертами игры. Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит результат игрока.

Реализуйте логику, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле, перечень элементов, для которых может сохранять анимацию, описывается экспертами отдельно (например, перемещение фона в режиме паузы). Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, как только доходит до значения 00:00 игра заканчивается
3. Режима теста - режим в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры
4. Реализация логики работы игрового поля, например, изменение количества собранных объектов, уменьшение количества “жизней”, уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - конец игры, проигрыш, выигрыш при наступлении определенных действий.

**Вариант 15**

Вам необходимо разработать браузерную игру.

**Описание игры**

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. Каждыйфрукт 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 10 фруктов.

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

На стартовом экране представлена инструкция к игре.

**Экран игры** - интерфейс должен содержать:

1. 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”
2. блок с заработанными баллами - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
3. блок с именем игрока.
4. блок, в котором отображается фрукт, по которым нужно кликать в данный момент времени
5. игровое поле - Процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы

**Экран окончания игры** - содержит набранное количество очков.

Реализация логики, в состав которой должны быть включены следующие функции:

1. Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле. Возобновление игры также возможно по нажатию на кнопку “Пауза” на игровом поле.
2. Таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение 00:30. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.

### Критерии оценивания

3: Изучен теоретический материал 5-6 главы. Выполнено задание (возможно реализация не всех функций). Составлен отчет о работе;

4: Изучен теоретический материал 5-6 главы. Выполнено задание (возможно с незначительными погрешностями). Составлен отчет о работе;

5: Изучен теоретический материал 5-6 главы. Выполнено задания. Составлен отчет о работе. Присутствует единый стиль оформления кода, комментарии, улучшен интерфейс программы.

### 