

Используя классово-объектную структуру, написать на любом объектно-ориентированном языке программирования следующую программу:

1. Игра «Пятнанадцать»
2. Игра «Реверси»
3. Игра «Пять в ряд»
4. Игра «Поддавки»
5. Игра «Пазл»
6. Игра «Питон»
7. Игра «Сапер»
8. Игра «Аквариум»
9. Игра «Морской бой»
10. Игра «Двойной Теннис»

Номер варианта соответствует последней цифре в номере зачетной книжки.

Способ оформления работы: код программы и исполняемый файл программы на дискете; электронный текстовый документ с **подробным** описанием всех классов и объектов, а также алгоритма работы программы; распечатанный вариант этого документа.

Основные требования к работе:

Все используемые в программе элементы должны быть описаны соответствующими объектами. При корректно составленной объектной структуре работы, мелкие недочеты в реализации не считаются.

Подробное описание заданий:

1. Игра «Пятнанадцать»

Правила игры:

На игровом поле размером 16 клеток (4 на 4) случайным образом размещается 15 фишек с номерами от 1 до 15. Таким образом, шестнадцатая клетка остается пустой. Игрок может, нажатием на смежную с пустой клеткой по вертикали или горизонтали фишку, переместить ее на пустую клетку. Соответственно, ранее занимаемая этой фишкой клетка становится пустой. Цель игры – путем таких перемещений выстроить фишки по номерам, начиная с верхнего левого угла. Если игроку это удалось, игра завершается сообщением «победа». Если получился вырожденный случай (фишки 14 и 15 оказались поменяны местами, то есть выиграть невозможно), то игра завершается сообщением «Не повезло».

Обязательно наличие счетчика ходов и возможности отмены последних сделанных ходов, вплоть до первого хода.

Также предусмотреть вариант игры на время, по истечении которого игра завершается проигрышем «Время вышло».

Подсказки по выполнению:

Массив фишек можно задать от 0 до 15, где число 0 будет означать пустую клетку.

Алгоритм заполнения поля фишками в случайном порядке: массив фишек задается связным списком, из которого случайным образом выбирается элемент, который размещается на поле (последовательно) и одновременно удаляется из списка. Процесс заиклиивается до тех пор, пока в списке не останется 0 элементов.

Процесс проверки на смежность: если у выделенной клетки с пустой клеткой один индекс совпадает, а второй различается на единицу, значит они смежны. Например, клетки (4,2) и (4,3). Если пользоваться одномерной нумерацией, то тогда первый индекс получается из номера элемента деленного нацело на 4 и +1, а второй индекс – остаток от деления на 4.

Например, элемент, стоящий на 9 месте в одномерном массиве, в двумерном будет располагаться в клетке $9 \div 4 + 1 = 3$, $9 \bmod 4 = 1$, то есть в клетке (3,1).

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Финальное расположение

1	2	3	4
5	6	7	8
9	10	11	12
13	15	14	

Вырожденный случай

2. Игра «Реверси»

Правила игры:

Играют двое игроков на поле размером 8 на 8 клеток. Один играет фишками белого цвета, другой – черного. Игроки выставляют свои фишки по очереди из начальной позиции по следующим правилам:

1. Если между выставленной игроком фишкой и любой фишкой его цвета на поле, по горизонтали, вертикали или диагонали окажутся фишки противника, эти фишки обращаются в цвет игрока, и ход считается результативным.
2. Каждая выставленная фишка должна совершать результативный ход (обращать хотя бы одну фишку противника).
3. Если результативных ходов нет, ход передается противнику.
4. Игра заканчивается, когда заняты все 64 клетки поля.
5. Победителем считается тот, фишек чьего цвета на поле больше.

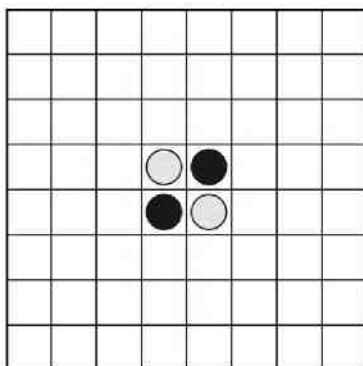
На экране должен отображаться текущий счет игры после каждого хода; должно сообщаться, чей сейчас ход, также можно отобразить время игры. По окончании игры показывается, кто победил и финальный счет.

Подсказки по выполнению:

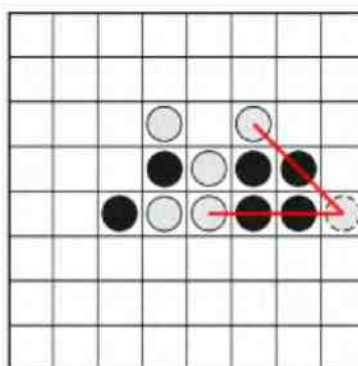
Для проверки того, какие фишки после хода обращать, а какие нет, можно воспользоваться следующим алгоритмом:

От выставленной фишки есть восемь направлений проверки.

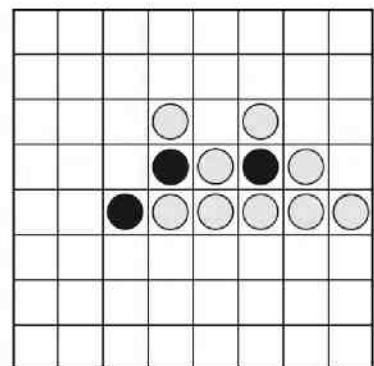
1. Выбрать первое направление.
2. Последовательно проверять клетки данного направления, перемещаясь по клеткам, занятым фишками противника.
3. Завершать проверку направления, если встретилось пустое поле, либо край доски, либо фишка того же цвета, что и выставленная. В последнем случае передавать флаг о том, что фишки в данном направлении надо обратить, и пройти по данному направлению еще раз, обращая фишки противника.
4. Выбрать следующее направление.
5. Выполнять пункты 2-4, пока не будут проверены все направления.



Начальное положение



Белые ставят шапку



Шапки обращены

3. Игра «Пять в ряд»

Правила игры:

Играют двое, на поле 13x13 клеток. Один играет крестиками, другой ноликами. Игроки по очереди делают ходы, выставляя свой значок на поле. Победит тот, кто сможет построить непрерывную цепь из пяти своих значков в одну линию по горизонтали, вертикали или диагонали.

Если одному из игроков это удалось, то победная цепь выделяется (жирным шрифтом, мигает, либо зачеркивается), и на экране отображается имя победителя. Желательно наличие счетчика ходов и времени игры.

Подсказки по выполнению:

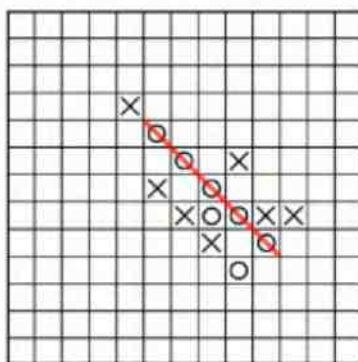
Крестик и нолик – элементы одного класса!

Для проверки того, построена цепь или нет, можно воспользоваться следующим алгоритмом:

От выставленного значка есть четыре пары направлений проверки в две противоположных стороны: вверх и вниз, влево и вправо, по диагонали влево-вверх и вправо-вниз; по диагонали вправо-вверх и влево-вниз.

1. Выбрать первую пару направлений
2. Выбрать, с какой стороны начать.
3. Сделать счетчик своих элементов равным единице.
4. Последовательно проверять клетки в выбранную сторону, с каждой клеткой, занятой своим значком, увеличивая счетчик. Завершить проверку направления, если встретился значок противника, либо край доски, либо пустое поле.
5. Если счетчик достиг значения 5, значит, цепь построена. Завершить программу.
6. Выполнить пункты 3 - 5 в противоположную сторону из данной пары.
7. Выбрать следующую пару направлений
8. Перейти к пункту 2.

Таким образом, если алгоритм не дал победного результата, значит игра продолжается.



«Нолики» победили

4. Игра «Поддавки».

Правила игры:

Играют два игрока на стандартной шахматной доске. Один играет белыми шашками, другой черными. Расстановка стандартная для русских шашек (первые три линии с каждой стороны, на черных клетках). Ходят по очереди только по черным клеткам. Рубить обязательно. Срубленная шашка убирается с доски. Выигрывает тот игрок, у которого не осталось шашек на доске.

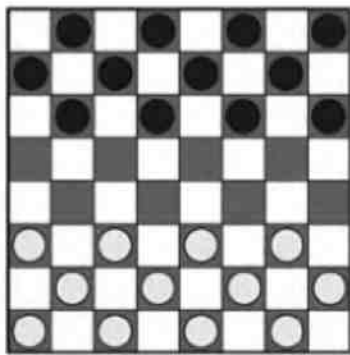
Ход: Игрок перемещает свою шашку на соседнюю клетку по диагонали, не занятую своей шашкой либо шашкой противника. Ходить можно только вперед (от своего края доски). Если шашка достигла противоположного края доски, она снимается с доски (дамок нет).

Рубка: Если в момент хода, рядом с шашкой игрока находится шашка противника, и позади нее в том же направлении есть свободная клетка, то шашка игрока «перепрыгивает» через шашку противника, «срубая» ее, и перемещаясь на следующую клетку после нее клетку. Если из нового положения ЭТОЙ ШАШКЕ опять есть что срубить, то рубка повторяется. Рубить можно и вперед и назад. Если в процессе рубки нескольких шашек подряд, шашка окажется на противоположном краю доски, она не убирается, а продолжает рубить в обратном направлении.

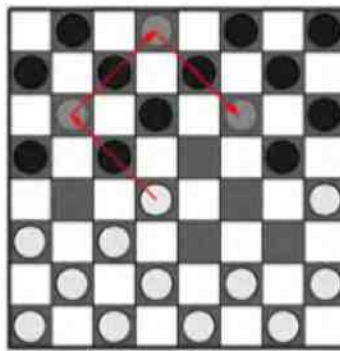
Если в момент хода имеется несколько вариантов рубки, то игрок выбирает любой из них.

Рубка считается сделанным ходом, и по ее окончании ход переходит к противнику.

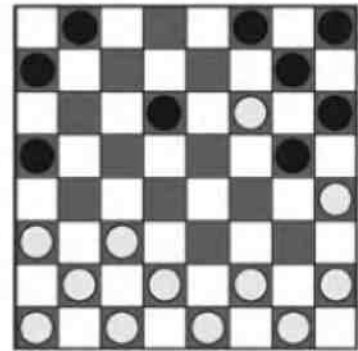
Программа должна следить за соблюдением правил игры.



Начальное положение



Белые обязаны рубить



Три шашки срублены

5. Игра «Паззл»

Правила игры:

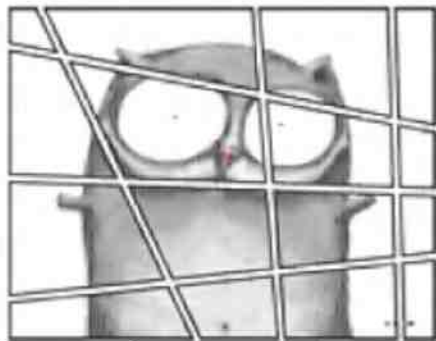
На игровое поле загружается картинка (любая по желанию игрока). Затем она разделяется прямыми непересекающимися линиями на некоторое количество четырехугольных кусочков (желательно сделать несколько уровней сложности с разным числом кусков). Данные кусочки перемешиваются случайным образом, образуя хаотичную кучу в правой части экрана. Задача игрока – перемещая кусочки из правой части экрана в центр, и прикладывая друг к другу, собрать первоначальную картинку. В любой момент можно переместить любой кусочек либо группу объединенных кусочков. После получения первоначальной картинки, игра заканчивается.

Подсказки по выполнению:

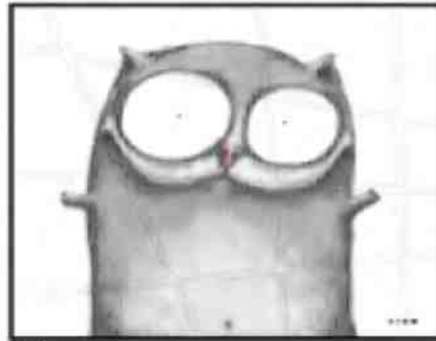
Для хранения каждого кусочка можно использовать отдельный компонент из визуальной библиотеки. Места разрезов на каждом кусочке можно закодировать определенным кодом, так, чтобы по обе стороны от разреза были одинаковые коды. Тогда проверка на правильность соединения сводится к проверке кодов сторон.

Еще один алгоритм проверки правильности соединения заключается во внесении всех кусочков в массив, и проверке номеров элементов данного массива. Если элементы смежны (один индекс имеет то же значение, а второй отличается на единицу, например (4,5) и (4,6)), то значит соединение верно.

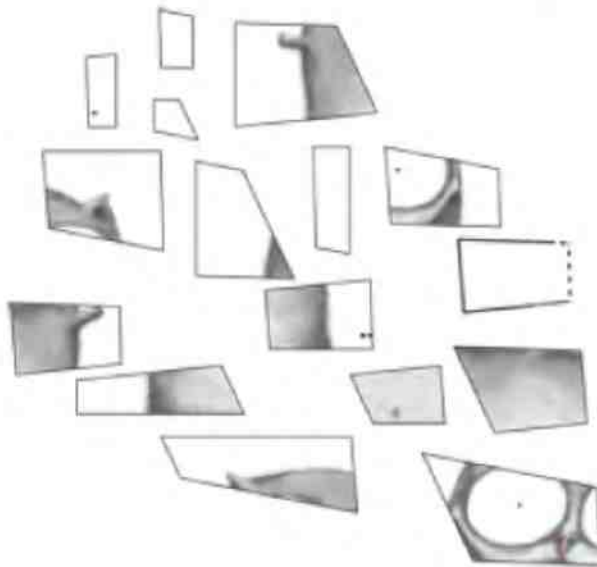
Если два кусочка соединены верно (расстояние между сторонами соединения на экране меньше какой-то величины, они «склеиваются» - вместо двух объектов остается один, содержащий в себе изображения обоих кусочков, либо просто два объекта «привязываются друг к другу и перемещаются вместе).



Картинка разбивается на части



Собранная картинка



Разрозненные кусочки картинки

6. Игра «Питон»

Правила игры:

По замкнутому прямоугольному полю, состоящему из невидимых клеток, перемещается питон. Питон состоит из головы, размером в одну клетку, и хвоста, первоначально размером в пять клеток. Сразу после появления, голова питона начинает движение прямо (в сторону от хвоста), с постоянной

скоростью. Хвост движется строго за головой, перемещаясь по тем клеткам, где была голова. Если игрок нажимает левую кнопку мыши (или клавишу «влево»), питон поворачивает на 90 градусов против часовой стрелки; при нажатии правой кнопки мыши (клавиши «вправо») – питон поворачивает на 90 градусов по часовой стрелке, и продолжает движение в новом направлении.

Каждые несколько секунд длина питона увеличивается на одну клетку.

На экране в случайных местах появляются «кролики» - случайные цифры от 1 до 9. Если питон «наедет» головой на кролика, то считается, что он его съел – счет игры увеличивается на величину значения съеденной цифры. Иногда на экране появляются мангусты – значки с буквой «М». Если питон «съест» мангуста, либо собственный хвост, либо границу экрана он умирает и игра заканчивается.

Кроме игрового поля на экране должен отображаться текущий счет, время игры, и текущая длина питона.

Подсказки по выполнению:

Суть движения питона состоит в том, что каждую единицу времени его голова перемещается на клетку вперед; там, где она находилась, рисуется элемент тела, а последний элемент хвоста стирается. Остальное тело не меняется и не перерисовывается. Процесс роста реализуется так же, только последний элемент хвоста стирать не надо – получится, что голова на одну единицу сместилась, а хвост остался на месте, то есть питон вырос на 1 клетку.

Движение против часовой стрелки: вправо, вверх, влево, вниз. По часовой: вправо, вниз, влево, вверх. При нажатии кнопки поворота, направление движения меняется циклично.

Не забудьте проверить, чтобы ваши кролики и мангусты не появлялись на уже занятых клетках!

7. Игра «Сапер»

Правила игры:

На квадратном поле 16х16 клеток случайным образом разбросаны 40 мин. Вокруг каждой мины находятся цифры – каждая цифра точно показывает, сколько вокруг нее мин в клетках, смежных по горизонтали, вертикали и диагонали. Если вокруг клетки мин нет, она пуста.

В начале игры поле закрыто. Игрок начинает проверять произвольные клетки, открывая их содержимое. Если он думает, что под данной клеткой – мина, он может не открывать ее, а маркировать флажком. Если игрок откроет пустую клетку (вокруг которой ни одной мины), то программа автоматически открывает все пустые клетки, смежные с данной, а также (не все!) не занятые минами клетки, смежные с пустыми.

Задача игрока – открыть все клетки поля, не занятые минами, и маркировать все мины. Если игрок попытается открыть заминированную клетку, он проигрывает.

На экране должно отображаться время игры и количество немаркированных мин.

Программа должна позволять игроку варьировать начальный размер поля и количество мин на нем.

Подсказки по выполнению:

Проще всего заполнить игровое поле, сперва случайным образом раскидав по нему мины, а потом посчитав для каждой клетки количество мин в смежных клетках.



Все поле открыто



Игрок нарвался на мину

8. Игра «Аквариум»

Правила игры:

Экран представляет собой аквариум, заполненный водой. В аквариуме живут карпы. Каждый карп умеет перемещаться внутри аквариума, разворачиваться возле стенок аквариума, и сканировать пространство перед собой в некотором секторе. Карпов в аквариуме много, и они плавают стаями. Также в аквариуме живет щука. Она движется быстрее любого карпа, но хуже видит (меньше радиус сканирования). Увидев карпа, щука начинает преследование, настигнув – съедает. Съеденный карп удаляется из аквариума (а его объект уничтожается). Карпы, увидев щуку, начинают движение в противоположную сторону (убегают). Программа завершает работу, когда щука съест последнего карпа.

Подсказки по выполнению:

Аквариум – это класс-контейнер; содержит в себе клас «рыба», к которому относятся и карп и щука. Чтобы создать стаю, карпов можно объединить в связный список. При съедании карпа, его объект уничтожается и удаляется из списка.

Описать движение рыб можно просто: движется по прямой, потом по случайному событию меняет направление движения в случайную сторону. Для организации процесса «сканирования» пространства, можно координаты всех рыб хранить в отдельном массиве и брать информацию о местоположении оттуда. Если положение какого-нибудь карпа находится в радиусе зрения щуки, она начинает преследование – вместо движения по прямой начинает движение кратчайшим путем к текущему положению карпа.

9. Игра «Морской бой»

Правила игры:

Играет один человек против компьютера.

На двух полях 10x10 клеток случайным образом расставляются корабли: 1 четырехпалубный, 2 трехпалубных, 3 двухпалубных, 4 однопалубных. Между любыми двумя кораблями на поле обязана быть хотя бы одна свободная клетка.

Поле, принадлежащее компьютеру, закрыто.

Игрок и компьютер начинают по очереди делать ходы, «стреляя» по клеткам поля противника. Если на «простреленной» клетке находится часть корабля, то на экране мигает надпись «ранен», и эта клетка выделяется оранжевым цветом. Если все палубы корабля пробиты, мигает надпись «убит», клетки корабля закрашиваются красным, а клетки вокруг корабля зачеркиваются крестом (по правилам там не может находиться другой корабль).

Компьютер делает ход случайным образом. Однако если он «ранил» корабль игрока, то он начинает обстреливать область вокруг этой клетки, пока корабль не будет потоплен. На поле игрока раненные и убитые корабли отображаются так же, как и на поле компьютера. Кроме того, ни компьютер, ни игрок не могут два раза стрелять в одну и ту же клетку, и в клетки вокруг потопленных кораблей, зачеркнутые крестом.

Выигрывает тот, кто первым потопит все корабли противника.

Подсказки к выполнению:

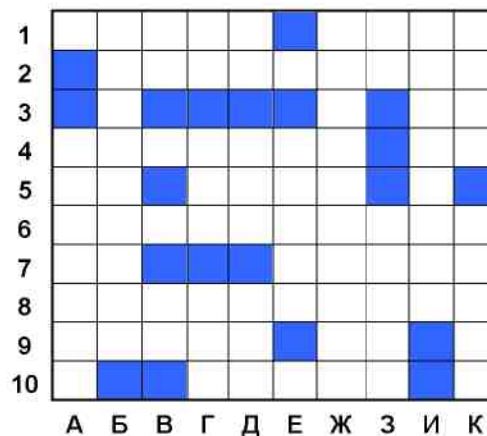
Все корабли – экземпляры одного класса, отличающиеся только количеством палуб.

Примерный алгоритм случайной расстановки кораблей на поле:

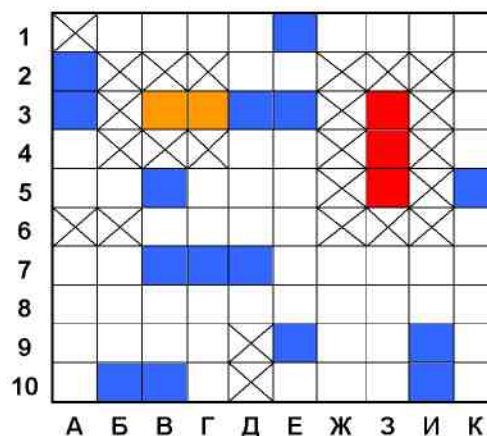
Расставлять корабли надо по размеру, в убывающем порядке, начиная с четырехпалубного. Текущее расположение может храниться в матрице 10x10, где занятые кораблями клетки, а также клетки вокруг (на них тоже нельзя размещать другие корабли), обозначены единицей, свободные клетки – нулем. Для каждого корабля выполняется следующий алгоритм (описан на примере четырехпалубного):

1. Начать проверять клетки поля в горизонтальном направлении с первой горизонтали.
2. Если данная клетка свободна, то проверить следующие за ней три клетки (итого 4 клетки под 4 палубы).

3. Если там тоже свободно, то значит сюда теоретически можно поставить корабль – сохранить координаты данной клетки в нумерованном списке, с пометкой, что это горизонтальное расположение.
 4. Перейти к следующей клетке этой горизонтали. Повторять пункты 2 и 3 до седьмой клетки включительно (далее проверять бессмысленно, так как корабль просто не войдет).
 5. Перейти на следующую горизонталь.
 6. Выполнить аналогичный алгоритм для вертикалей.
 7. Из получившегося списка разрешенных координат выбрать случайным образом одну координату.
 8. Разместить корабль, начиная с выбранной координаты, в соответствующем направлении.
 9. Внести изменения в матрицу занятых клеток, добавив туда клетки, занятые кораблем, и клетки вокруг корабля.
 10. Обнулить связный список
- И так для всех кораблей.



Пример начальной расстановки кораблей



Трехпалубный подбит, четырехпалубный ранен

10. Игра «Двойной Теннис».

Правила игры:

Играют двое игроков. Экран представляет собой игровое поле, ограниченное слева и справа. Ракетки игроков расположены сверху и снизу экрана, и выглядят по-разному.

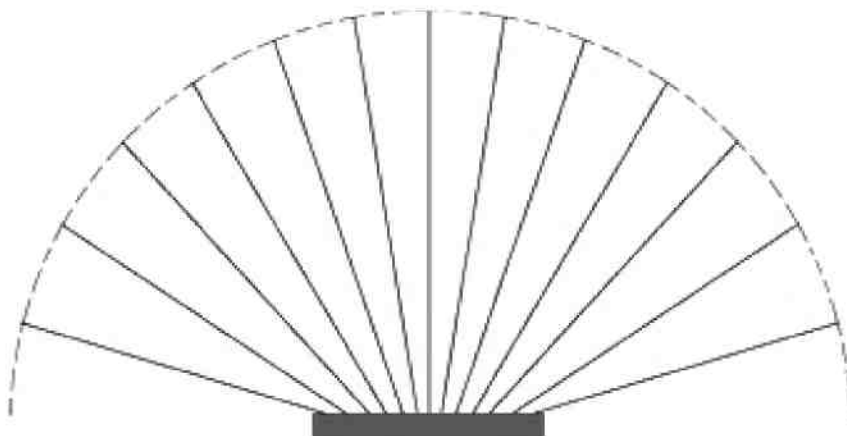
В начале игры ракетки находятся посередине экрана, и на каждой ракетке находится по мячу разного цвета, которые после нажатия клавиши «старт» одновременно вылетают под определенным углом (см. ниже) в сторону противника.

Перемещая свою ракетку в горизонтальной плоскости, каждый игрок должен подставлять свою ракетку под летящие в его сторону мячи, отбивая их. Если мяч пролетел мимо ракетки, игроку-сопернику начисляется призовое очко, а мячи и ракетки возвращаются на исходные позиции. Игра ведется до заранее заданного количества очков.

Скорость перемещения ракеток постоянна, скорость перемещения мячей изначально одинакова и постепенно растет.

От стен и друг от друга мячи отражаются по правилам оптики – под тем же углом, что и упали.

Угол отражения мяча от ракетки зависит не от угла падения, а от места касания ракетки (см. рисунок):



Мяч отразится от ракетки в соответствующем направлении