

Правила игры

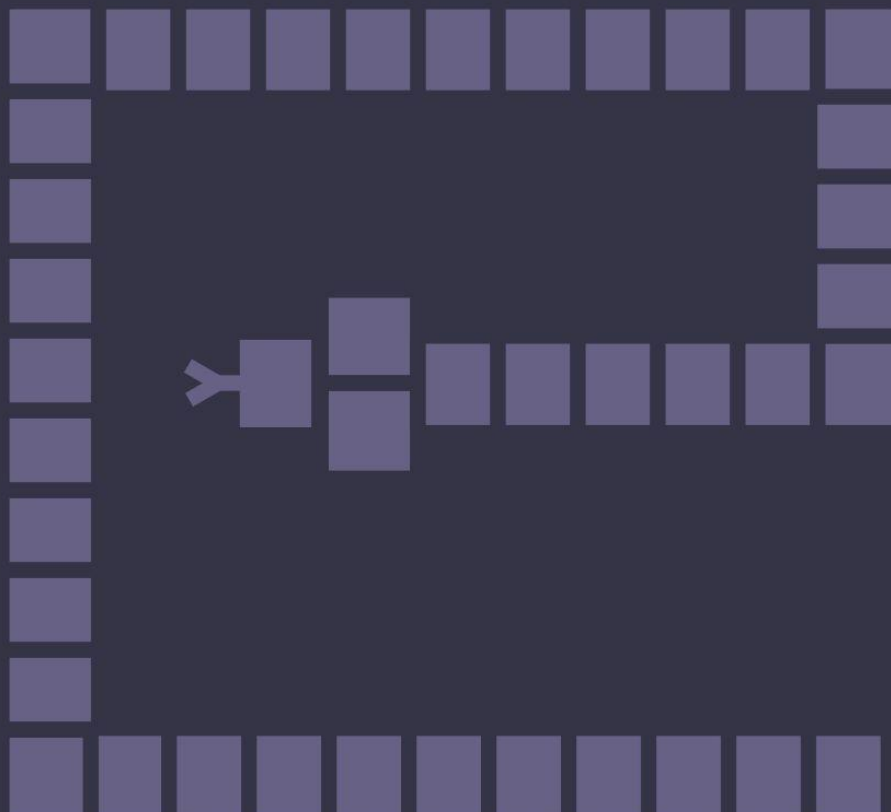


#DEVELOBEAR

**#DEVELOBEAR**

#DEVELOBEAR

V Турнир по программированию НПО Криста и ЯрГУ



# Описание

Как и все турниры `develobear`, текущий не будет исключением. Участникам необходимо разработать программу, которая на игровом поле будет соревноваться с программами других участников.

В этом году на поле одновременно будут сражаться по 4 алгоритма.

Среда разработки предоставлена прямо в браузере, а язык программирования будет необычным и новым для всех участников.

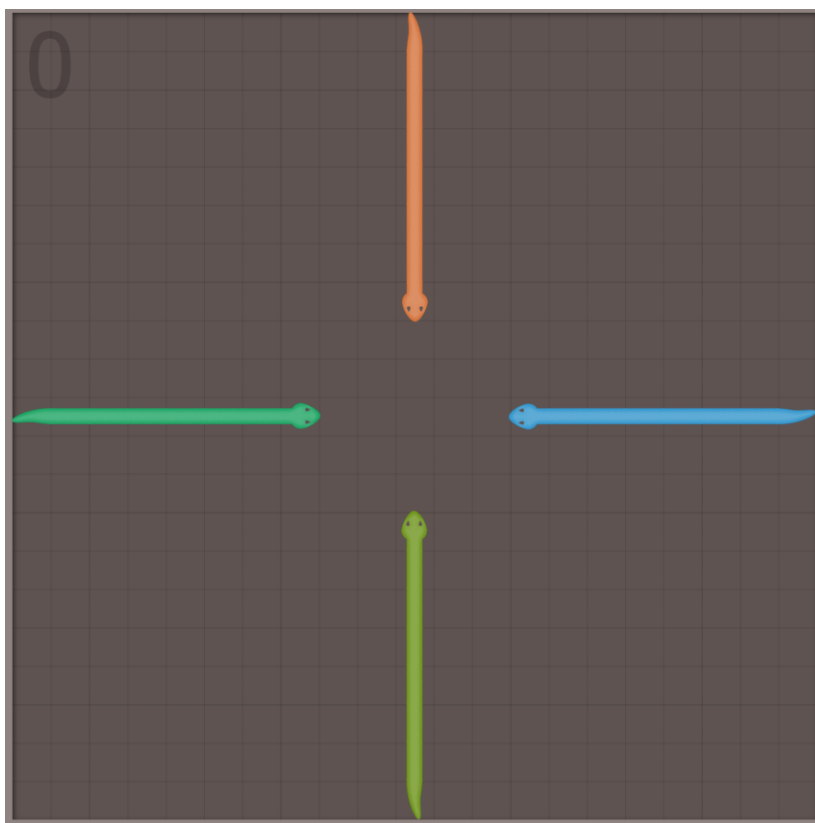
Личный кабинет со средой программирования размещается по адресу:

**<https://snake.develobear.ru>**

## Правила игры

Игра достаточно простая. На поле боя появляются 4 змеи. Участники должны написать программу, которая будет управлять одной змеей. Задача - съесть других змей на поле боя. Кусать можно только за хвост. Размер поля 21x21. Изначальная длина змеи 8 клеток. При откусывании хвоста у соперника змея увеличивается в размере. И наоборот, если хвост змеи был укушен, её размер сокращается.

По результатам раунда змеи ранжируются. Побеждает самая длинная змея или та, которая съела всех других. Второе место занимает вторая по длине змея или та, которая была съедена последней и т. д. Один раунд состоит из 500 шагов. На каждом шаге все 4 змеи принимают решение, куда им сходить. Однако змеи ходят не одновременно, а по очереди. Очередность хода змеи в рамках одного шага каждый раз выбирается случайным образом. Например, если на первом шаге оранжевая змея ходила первой, то на втором шаге она, может быть, будет ходить второй или четвертой.



# Правила турнира

Сам турнир проходит в два этапа.

Первый этап групповой, второй проходит по принципу плей-офф.

Групповая часть турнира состоит из туров. Перед каждым туром все участники случайным образом разбиваются на группы по 4. Далее внутри каждой группы проходит матч. Он состоит из нескольких десятков раундов. Раунд, как описано ранее, - это непосредственно сражение на поле боя 4-х змей, состоящее из 500 шагов. После завершения всех раундов (матча) происходит назначение очков в каждой группе: за первое место +2, за второе +1, за третье -1, за четвертое -2.

Далее начинается следующий тур, где всё повторяется. Случайным образом формируются команды по 4 участника, и проходят новые матчи.

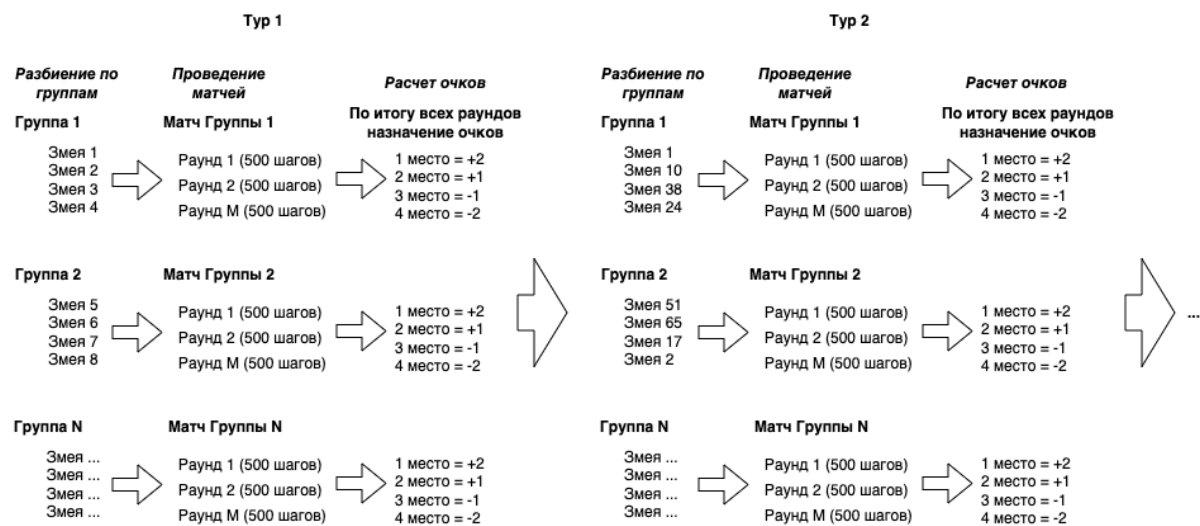
Команда выбывает, если набирает -10 очков.

После 25-го тура каждые 3 тура снимаются по 3 очка с каждой команды.

Это будет происходить до тех пор, пока не останутся 32 команды, которые пройдут в плей-офф.

В плей-офф схема проведения всем известна. Формируются группы по 4 команды. В каждой группе играется матч по несколько десятков раундов. После этого определяются по два победителя в каждой группе стадии, которые выходят в следующую стадию плей-офф.

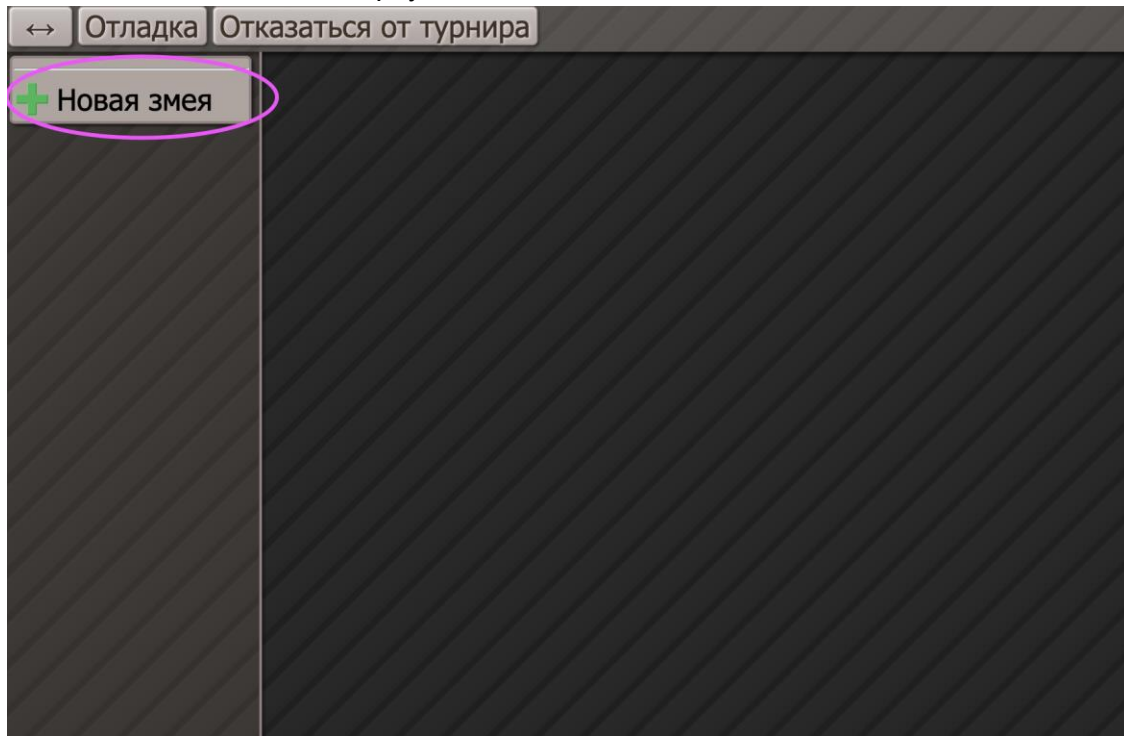
## Групповая стадия



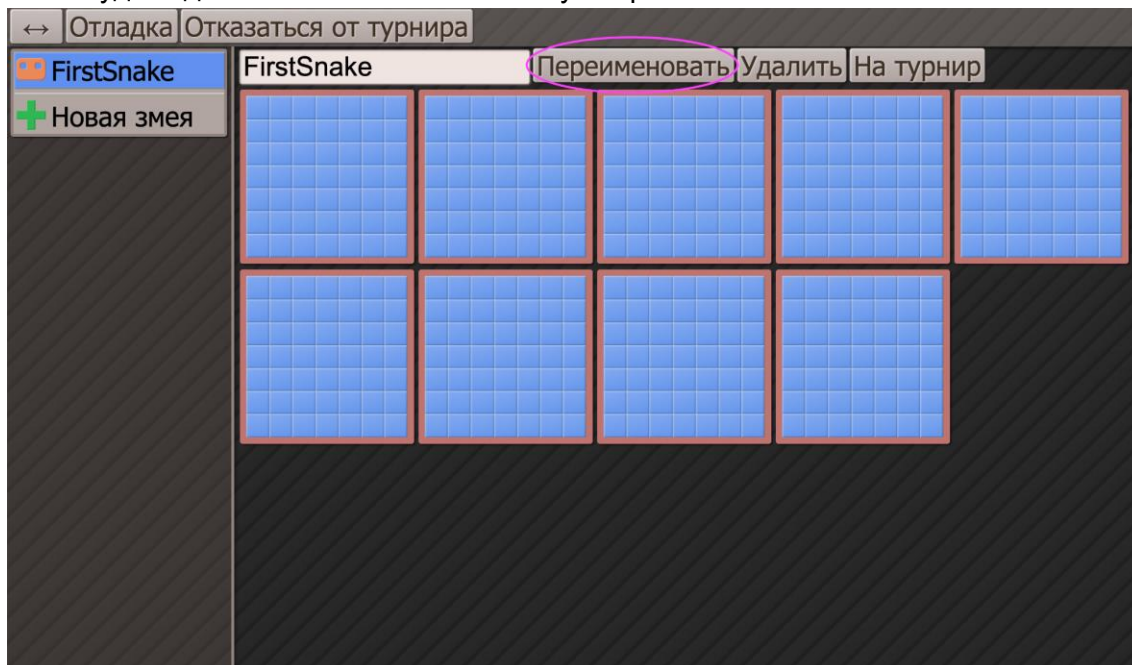
# Среда разработки и язык программирования

В этом году решено не использовать языки общего назначения. Вам будет предоставлен специально разработанный для решения именно этой задачи декларативный язык программирования, который мы назвали ViPER - Visual Pattern Enhanced Recitation.

После входа в личный кабинет вы окажетесь в среде разработки. Первое, что нужно сделать, - создать свою первую змею!



Не забудьте дать ей имя и нажать кнопку “Переименовать”.



Поведение змеи программируется с помощью схем (паттернов) размером 7x7 клеток. Составление паттернов поведения змеи делается очень наглядно и просто. У каждой змеи есть девять программируемых схем/паттернов поведения. Схема 1 имеет более высокий приоритет, чем схема 2 и т. д. То есть, если змея исполняет программу, записанную в первой схеме, до остальных она не дойдет. Если же реальная обстановка не соответствует программе, записанной в схеме 1, то может быть использована схема 2 и т. д.

Схема считается совпавшей с фрагментом поля, если при наложении под всеми элементами схемы оказались такие же элементы на поле. Если это произошло, то направление своей головы на карте покажет направление движения на следующем шаге. Если совпадения нет или подобный ход невозможен, то изучаются все повороты и зеркальные отражения данной схемы. Если совпадения все равно не произошло, то берется следующая схема и процедура запускается снова. Если же сравнение всех схем не принесло результата, то направление следующего хода выбирается случайно.

Существуют группы "И"/"ИЛИ" для элементов схемы. Для удобства каждая группа имеет свой цвет (жёлтый, синий, белый, зелёный). Логические группы зависят от типа самой схемы: если она поведения "И", то доступны 4 группы "ИЛИ", и наоборот, если схема "ИЛИ", то доступны 4 группы "И". Использование группы "И" требует, чтобы одновременно совпали все элементы одного соответствующего цвета. При использовании групп "ИЛИ" должен совпасть хотя бы один элемент из группы. Также существует элемент "Кроме", который означает, что элемент считается совпавшим, если он любой, кроме указанного в соответствующем квадрате. Тип самой схемы ("И"/"ИЛИ") дает общее указание на то, как именно должны соединяться различные цветовые группы.

Группа "Без группы" всегда применяется к другим "цветным" группам по правилу "И", все примитивы, которые находятся внутри группы "Без группы", тоже всегда объединяются по правилу "И" вне зависимости от типа схемы.

## Примеры

Рассмотрим тип схемы "И".

На каждой схеме обязательно должна присутствовать "Своя голова". Это точка отсчёта и применения схемы. "Своя голова" на схеме всегда присутствует в единственном экземпляре и в группе под названием "Без группы".

В схеме "И" элементы поля внутри каждой группы объединяются по "ИЛИ".

Приведённый ниже паттерн будет считаться совпавшим, если на поле боя соответствует хотя бы одно из трёх расположений хвоста противника относительно головы вашей змеи. Потому что хвосты входят в группу "ИЛИ".

Однако элементы, которые находятся в группе "Без группы", всегда внутри себя требуют полного соответствия, т. е. объединяются по "И".

Если записать это в виде правил булевой логики, то схемы "И" вычисляются так:

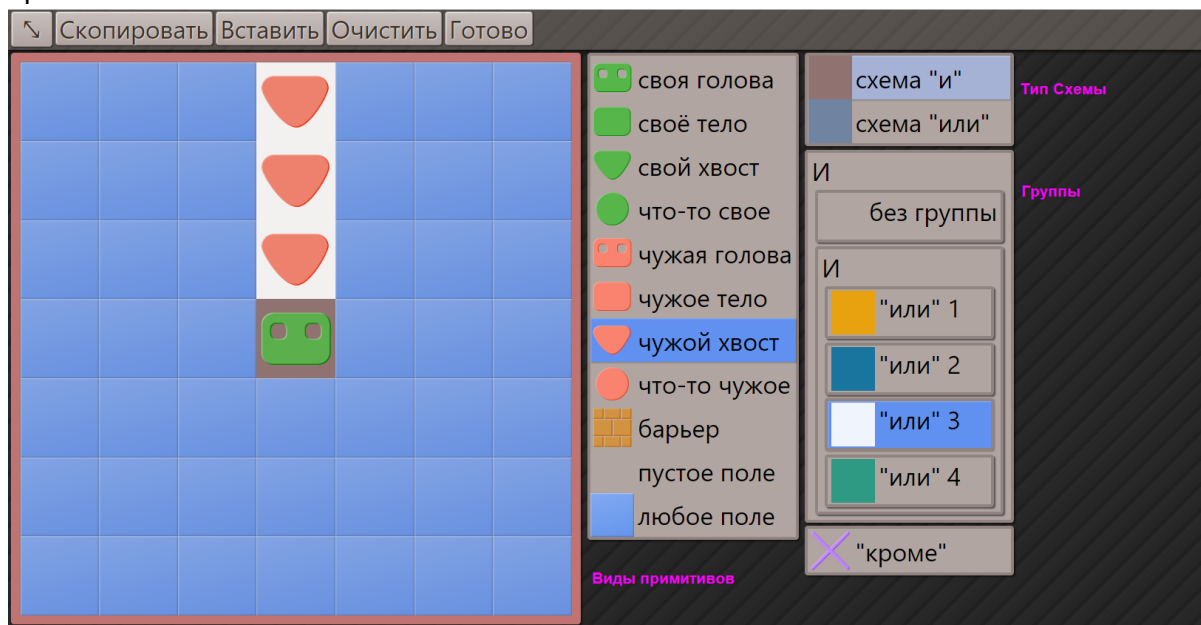
(полное совпадение серого/"Без группы")

И (любой элемент из белой группы)

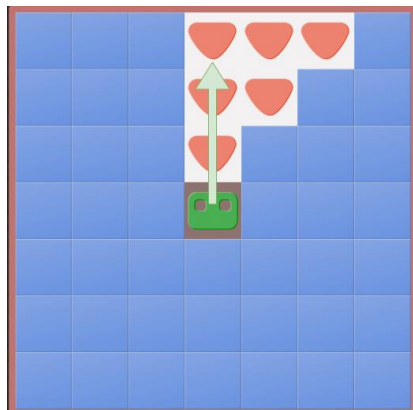
И (любой элемент из жёлтой группы)

И (любой элемент из синей группы)  
И (любой элемент из зелёной группы)

Приведённый ниже паттерн заставляет змейку двигаться в направлении хвоста противника.

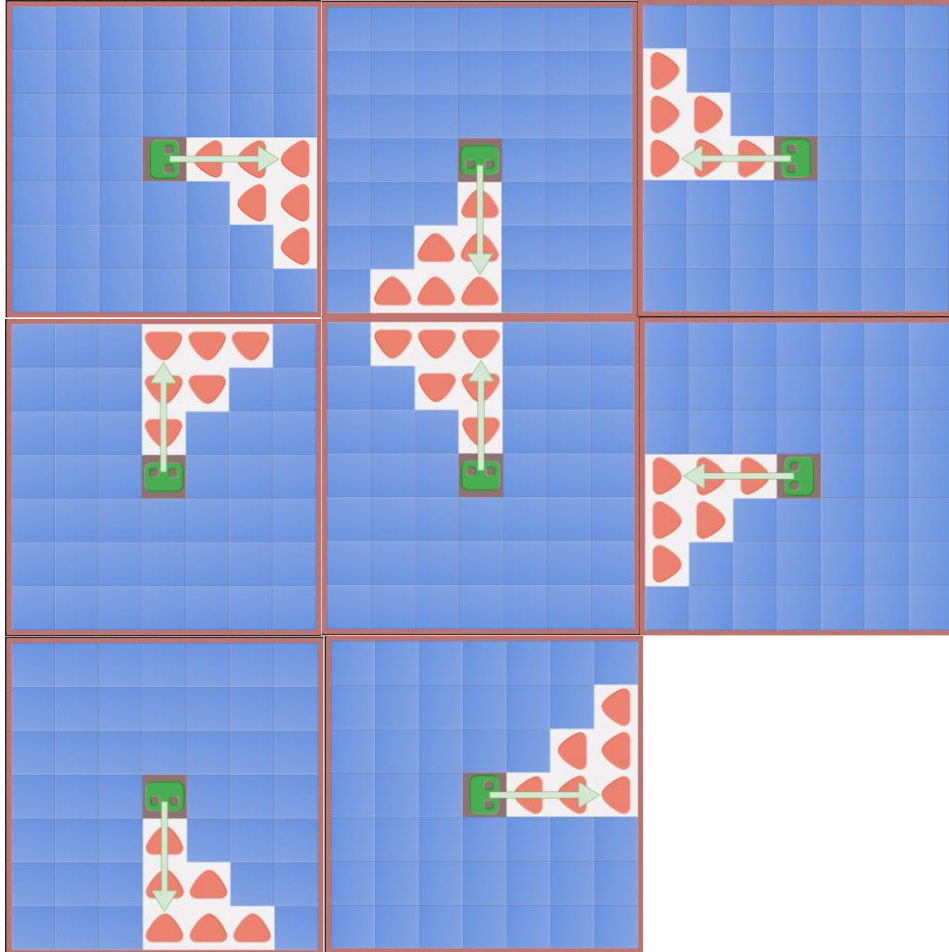


Однако это очень узкий случай, и правильнее было бы двигаться за хвостом, когда он не непосредственно напротив головы, а где-то неподалеку от головы. Тогда паттерн можно было бы записать следующим образом:

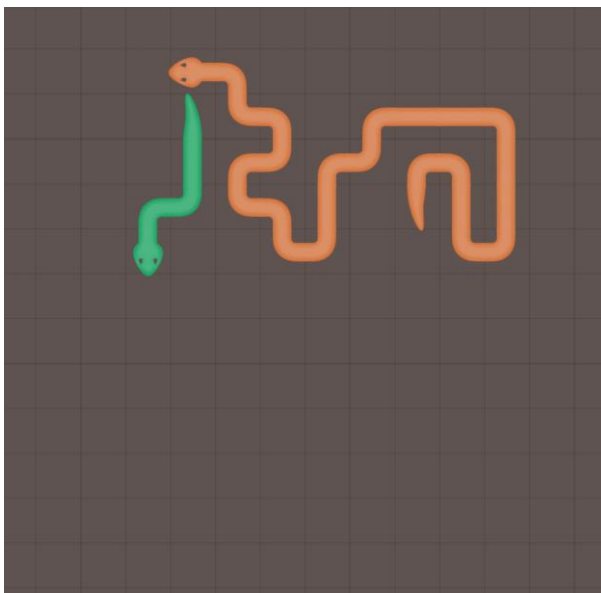


При этом вам не требуется разрисовывать все возможные положения хвостов, т. е. справа, снизу, слева, т. к. паттерн автоматически попытается применяться во всех возможных позициях. Т. е. он будет поворачиваться на 90 % относительно головы, зеркально отражаться и пробовать применяться к игровому полю.

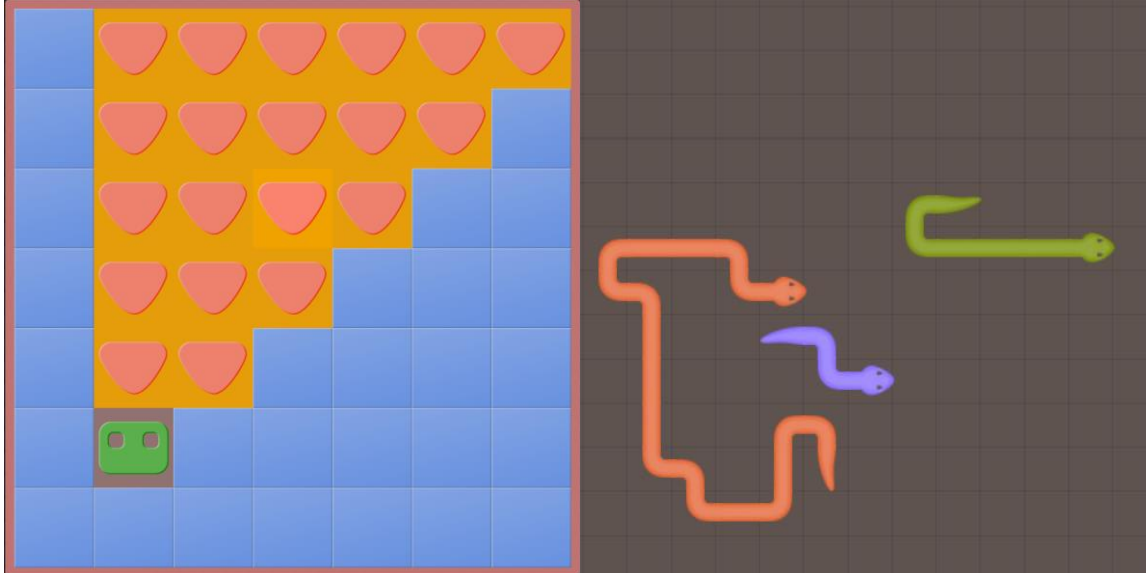
Однако последовательность применения случайна, вы не сможете контролировать, какая из модификаций паттерна применится первой.



Рассмотрим игровую ситуацию, в которой этот паттерн применится. У оранжевой змеи сработает именно этот паттерн, и, если ей повезет сходить в следующем шаге первой, она съест хвост зеленой змеи.



Несмотря на то что паттерны применяются последовательно, нужно иметь в виду, что мы не контролируем, как и в какой последовательности варианты одного паттерна будут применяться. Рассмотрим применение паттерна “схема И” к игровой ситуации.



Логичным было бы предположить, что оранжевая змея должна была пойти съесть хвост фиолетовой змеи. Однако это может быть не так, т. к. данный паттерн может успешно применяться как к хвосту фиолетовой змеи, так и к хвосту зеленой змеи. Поэтому оранжевая змея может пойти и к хвосту зеленой змеи, а не фиолетовой. Мы не можем контролировать последовательность применения вариантов паттерна. Чтобы избежать такой путаницы, возможно, потребуется два паттерна: один с коротким обзором, другой с дальним.

## Рассмотрим тип схемы “ИЛИ”

Если записать эту схему формулой булевой логики, то получится

(полное совпадение серого/“Без группы”)

И

(

(любой элемент из белой группы)

ИЛИ (любой элемент из жёлтой группы)

ИЛИ (любой элемент из синей группы)

ИЛИ (любой элемент из зелёной группы)

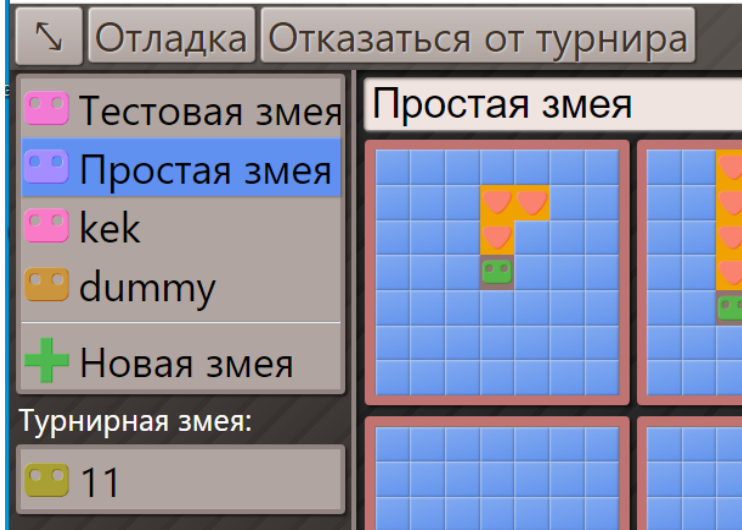
)

Т. е. группа “Без группы” имеет абсолютно то же поведение, что в схеме “И”, что в схеме “ИЛИ”.

## Отладка

Отладить один паттерн нельзя. Можно отладить работу всей змеи сразу. Чтобы отладить змею, необходимо нажать на кнопку “Отладка”.





Змеям перед каждым следующим шагом присваивается приоритет хода. После того как сходили все 4 змеи, их приоритеты случайным образом меняются. В режиме отладки остановка происходит всегда на первой в списке змее, несмотря на то, какой она по счёту ходит.

*Поэтому визуально иногда вы можете заметить ситуацию, когда при следующем шаге какая-то из змей может сделать два хода. Это связано с тем, что на предыдущем шаге она ходила 4-ой, а на текущем, например, первой.*

Доступна работа с клавиатуры, стрелки - шаг вперед, шаг назад, пробел - старт/стоп отладки.

