Проект Pygame «Тетрис»

Автор проекта: Карпова Алиса

Идея создания проекта:

Данный проект - реализация приближенной к классической версии компьютерной игры «Тетрис». На главном экране игры изображено клетчатое поле шириной 10 и высотой 18 клеток (в оригинальной версии 10 х 20). Сверху начинают падать случайные геометрические фигуры, состоящие из четырех квадратов. Во время полета фигуры игрок может поворачивать её на 90 градусов по часовой стрелке, перемещать по горизонтали, а также ускорять её перемещение по вертикали. Пользователь может управлять фигурой до тех пор, пока она не столкнулась с нижней частью поля, либо с другой фигурой по вертикали. Если во время игры полностью заполняется один или сразу несколько горизонтальных рядов поля, эти ряды исчезают, и всё, что находилось выше, смещается вниз на соответствующее количество клеток. За каждый полностью собранный ряд начисляются очки (количество начисляемых очков зависит от того, сколько рядов за ход заполнил пользователь). После заполнения каждых 10 рядов происходит увеличение уровня и увеличивается скорость падения фигур. Игра оканчивается тогда, когда сверху не может начать падать следующая фигура.

Реализация проекта:

Все компоненты игры и их взаимодействие друг с другом реализованы с помощью классов. Код содержит 7 классов: Tetris (главный из них; регулирует работу программы), GameBoard (отображает игровое поле и устанавливает взаимодействие между его объектами), Preview (отображает следующие фигуры), Score (регулирует подсчёт очков), Timer (устанавливает задержку времени для отображения перемещения и вращения фигур), Block (создает блоки-спрайты, из которых состоят фигуры), Shapes (создает фигуры из блоков).

Приемы, использованные при реализации:

* Установление связи между классами с помощью метода, принадлежащего одному из них (пример из программы: изначально объект класса Preview не имеет прямого доступа к методу get\_next\_shapes, генерирующему список следующих фигур, т.к. он принадлежит классу Tetris; для решения проблемы внутри одного из методов класса Tetris создается объект класса Preview, которому в качестве аргумента к одному из методов передается метод get\_next\_shapes)
* Установление связи между главным экраном игры (display\_surface) и другими поверхностями с помощью конструкции: pygame.display.get\_surface()
* Создание двухмерной матрицы field\_data для хранения местоположения всех спрайтов игрового поля и упрощения проверки объектов на столкновения

Необходимые для запуска модули / библиотеки:

* Требующие установки (Pygame)
* Встроенные (sys, os, random)

Скриншоты:

