

Лабораторная работа 8

Потоки и анимация

Часть 1. Анимация

В этом задании нужно моделировать движение объекта с учетом гравитации. Будем считать, что сила тяготения сонаправлена с осью ОУ в tkinter (т.е. направлена вниз). Объектом должен быть шарик, который движется внутри прямоугольника, испытывая абсолютно упругие соударения с его стенками.

Радиус шарика, его цвет, начальные координаты и начальные компоненты скорости по осям ОХ и ОУ задаются в интерактивном режиме. Должна быть панель (окно), в которой возможно изменять характеристики шарика и применять сделанные изменения.

Начальные координаты должны задаваться относительно одного из углов прямоугольника (например, верхнего левого).

Также следует предусмотреть панель (окно), в которой возможно выполнять настройки "окружения шарика": задавать размеры прямоугольника и ускорение свободного падения.

Предусмотрите возможность включать и выключать отображение траектории шарика (более точно, траекторию его центра как материальной точки).

Описание шарика и его возможностей (в т.ч. движения) должно быть сделано с помощью класса. Можете попробовать организовать анимацию при помощи потоков (один шарик = 1 поток).

Дальнейшая доработка проекта:

1. Считайте, что при каждом соударении со стенками шарик теряет некоторое количество энергии (что выражается в уменьшении по абсолютной величине компонент его скорости).
2. Замените область движения шарика с прямоугольника на какую-либо кривую (например, окружность или внутренность гиперболы; допустимы ступенчатые функции, многоугольники и т. п.), либо добавьте некоторое препятствие (например, ступень, наклонную плоскость и т.п.)
3. Предусмотрите возможность добавления любого количества шариков, которые будут двигаться, оставляя траектории разного цвета.
4. Предусмотрите возможность "паузы" (временной остановки) движения объектов.
5. Придумайте формат для хранения настроек шарика и области его движения (своего рода файл конфигурации), так чтобы можно было загружать этот файл и использовать эти настройки при исполнении вашей программы.
6. Можете добавить обработку соударений шариков друг с другом в соответствии с законами физики.

Полезные ресурсы:

- <https://scipython.com/blog/the-double-pendulum/>
- <https://scipython.com/blog/the-spring-pendulum/>
- <https://scipython.com/blog/chaotic-balls/>
- <https://scipython.com/blog/molecular-dynamics-modelling-of-effusion-into-a-vacuum/>
- https://www.academia.edu/18519946/Computational_Physics_with_Python

Часть 2. Потоки

Разработайте приложение, в котором одновременно будут запущены несколько потоков. Описание того, что эти потоки должны делать, приведено ниже. Приложение должно быть настраиваемым, т. е. пользователь должен иметь возможность выбрать количество потоков, настроить цвет, шрифт, размер и/или другие параметры для выводимой

информации. Постарайтесь разобраться с GIL в Python, синхронизируйте между собой работу потоков.

Варианты заданий

- 1 Каждый поток должен выводить по одному слову из четверостишия. Вывод должен происходить в случайно выбранной точке. Если слова могут перекрыться, попробуйте снова выбрать случайную точку. Каждое четверостишие должно выводиться своим цветом и шрифтом.
- 2 Каждый поток закрашивает определенный участок (например, прямоугольник) некоторым (определенным заранее) цветом. Закрашивание должно происходить в случайно выбранном направлении, "постепенно", начиная со случайно выбранной точки. Если потоки "сталкиваются", то они должны делить область, которую каждый из них закрасил бы в отдельности, на равные части.
- 3 Каждый поток рисует некоторый многоугольник (выпуклый). Одним действием при этом считается рисование линии. Начинается рисование в случайно выбранной точке. Желательно объединять многоугольники, если они пересекаются или вкладываются друг в друга.
- 4 Каждый поток имитирует движение простой геометрической фигуры (небольшого закрашенного квадрата, например). Столкновения между фигурами между собой не учитывать, учитывать только законы отражения (от "стенок"). Фигура при создании потока должна появиться в случайно выбранной точке. При отражении цвет поменяйте на дополнительный.
- 5 Каждый поток вычисляет среднее арифметическое чисел в заданном диапазоне (числа из этого диапазона выбираются случайно) "нарастающим итогом" (результат постоянно обновляется на экране).
- 6 Каждый поток заполняет некоторыми числами матрицу. Заполняемый элемент выбирается случайным образом. Элементы матрицы сопоставляются точкам некоторой области на экране, в момент заполнения эти точки должны "подсвечиваться" (разными цветами для разных потоков).
- 7 Каждый поток изображает закрашенный круг некоторого диаметра. Закраска круга - градиентная (одним тоном) от центра к краям. Центр круга и его диаметр выбирается случайным образом. Желательно при перекрытии кругов смешивать цвета.
- 8 В некоторой области экрана движется небольшой объект (например, шарик). Начальное положение и скорость объекта выбираются произвольно. Каждый поток рисует отрезок прямой (длина и положение в пространстве выбираются произвольно). Этот отрезок существует на экране в течение некоторого времени. Если за это время шарик успеет соприкоснуться с этим отрезком, он должен отскочить от него (согласно закону отражения).
- 9 Каждый поток изображает некоторую закрашенную фигуру, которая расширяется с течением времени. При перекрытии фигуры объединяются, а цвета медленно (по мере расширения) перемешиваются. Точка, в которой расположен центр фигуры, выбирается случайным образом.
- 10 Каждый поток изображает некоторую закрашенную фигуру, которая расширяется с течением времени. При столкновении с другой фигурой цвет меняется на дополнительный, и фигура начинает сжиматься до точки. После этого вновь происходит расширение.
- 11 Некоторая область экрана делится на клетки. Каждый поток ставит в произвольно выбранную клетку некоторую фигуру или букву, свойственную именно этому потоку.

Если клетка занята, происходит попытка выбрать другую клетку. Если занято некоторое (заранее заданное число) клеток, расположенных рядом, то они очищаются.

- 12 Некоторая область экрана делится на клетки. Каждый поток, начиная с произвольно выбранной клетки, перемещается к следующей по некоторому правилу, допускающему не менее 4-х вариантов движения из внутренней клетки (подобно шахматной фигуре, можно генерировать разные правила для разных потоков). Вариант движения выбирается случайным образом. Если выбранная вновь клетка уже была пройдена этим потоком, делается попытка выбрать другую клетку. Если поток оказался в "тупике" (из текущей клетки никуда нельзя попасть), он уничтожается с выводом соответствующего сообщения. Если при этом все клетки были пройдены, область закрашивается одним цветом (вообще, визуализацию процесса продумайте самостоятельно).