### Астероиды

#### Тема лабораторной работы

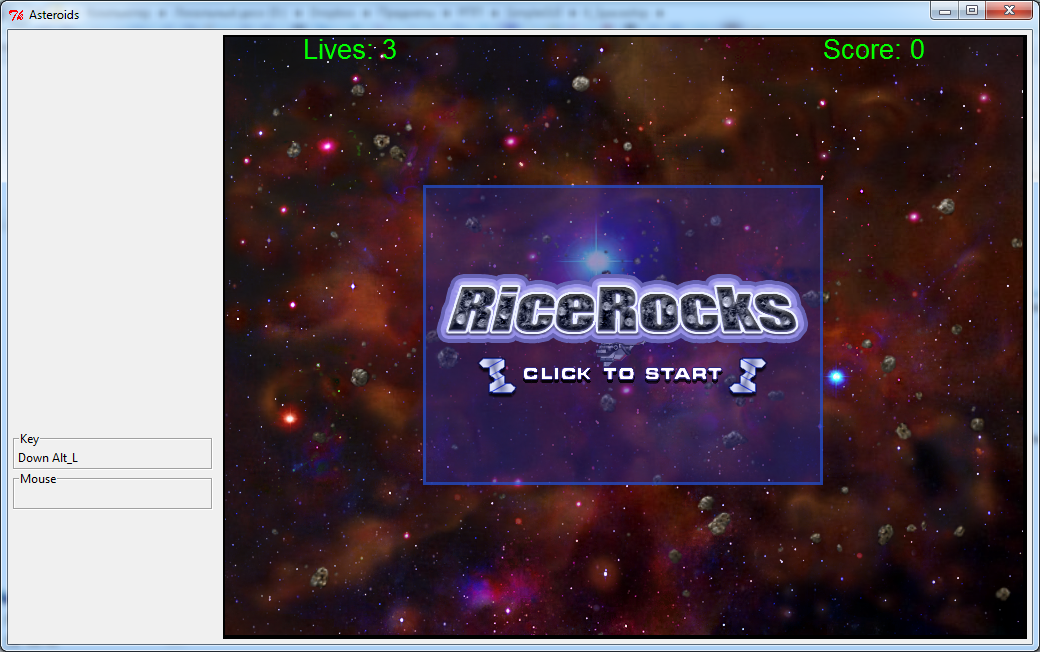
Тайловая анимация изображений, сложное взаимодействие объектов.

#### Цель лабораторной работы

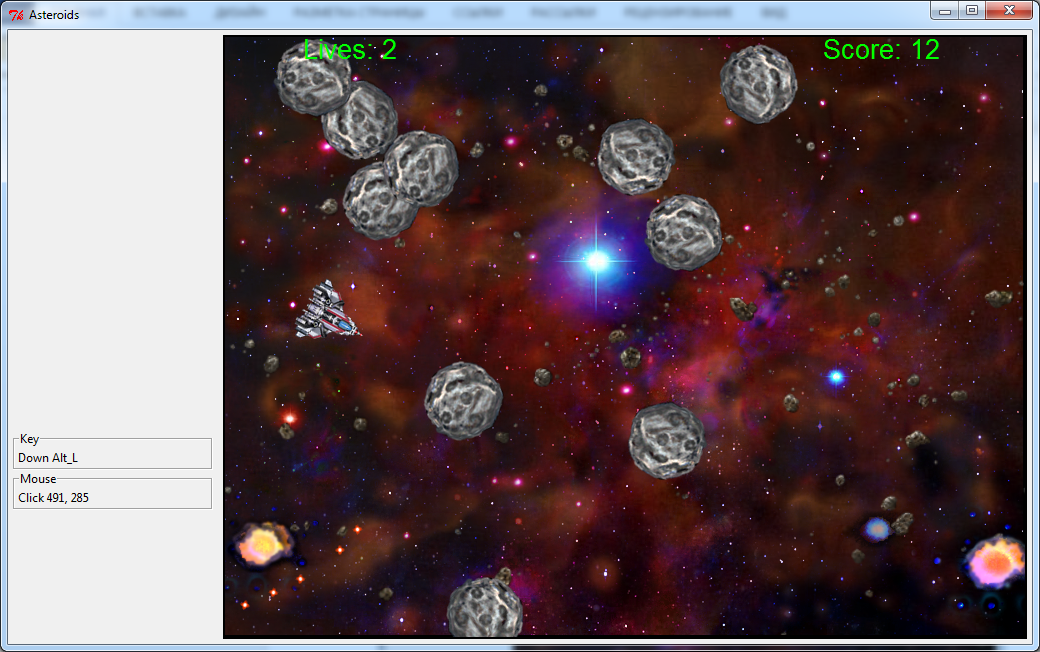
Научиться проектировать и строить сложные, многофайловые приложения по заданию, принимать инженерные и проектировочные решения.

Необходимо написать двумерную аркадную игру «Астероиды» по следующим требованиям:

1. При старте игры на экране отображается (см. рисунки):
   1. Начальное количество жизней (3);
   2. Начальное количество очков (0);
   3. неподвижный фон;
   4. динамический фон – полупрозрачное изображение астероидов, которое движется вдоль экрана влево с постоянной скоростью;
   5. заставка игры – по центру.
2. При щелчке на заставке игра начинается. В центре игровой области появляется корабль. Также начинают генерироваться астероиды в произвольных местах игрового поля с определенной частотой.
3. Корабль управляется игроком с клавиатуры и может:
   1. вращаться относительно своего центра;
   2. стрелять ракетами по направлению ориентации корабля;
   3. ускоряться по направлению ориентации корабля;
4. Игровое поле имеет тороидальную геометрию, то есть при пересечении любым объектом, например, левой границы экрана он появляется с правой стороны, сохраняя направление и величину линейной и угловой скорости.
5. При ускорении изображение корабля должно меняться на изображение корабля с включенными двигателями. При завершении ускорения изображение должно меняться на исходное.
6. При выключенных двигателях корабля, он испытывает сопротивление среды и постепенно замедляется до полной остановки.
7. Астероиды при появлении имеют случайную линейную и угловую скорость.
8. Ракеты должны появляться из «носа» корабля и лететь в ту сторону, в которую ориентировано изображение корабля с постоянной скоростью.
9. Ракеты имеют ограниченное время жизни. По его истечении ракета исчезает с экрана. Это время жизни нужно выбрать исходя из отношения размеров экрана к скорости полета ракеты таким образом, чтобы ракета не могла пролететь все игровое поле даже по малому измерению.
10. При столкновении ракеты и астероида астероид и ракета исчезают с экрана, на месте астероида показывается анимация взрыва, игроку начисляется одно очко.
11. При столкновении корабля и астероида, астероид исчезает с экрана, на его месте показывается анимация взрыва, корабль возрождается неподвижный в середине игрового поля, количество жизней игрока уменьшается на единицу.
12. Если на момент смерти игрока, количество жизней равнялось 0, игра считается оконченной, на экране снова демонстрируется заставка (все астероиды, ракеты и корабль не прорисовываются), счет и количество жизней остаются на экране.
13. Астероиды не сталкиваются, а свободно проходят друг поверх друга.



Начальное окно игры «Астероиды»



Игровой процесс в игре «Астероиды». Обратите внимание на анимацию взрыва снизу слева и справа.

Указания к выполнению лабораторной:

1. Все необходимые параметры игры выбираются самостоятельно исходя из интереса игры.
2. Код программы надлежит разбить на несколько (2 – 4) файлов исходя из логики и соображений читаемости. В частности, рекомендуется определения классов выносить в отдельные файлы и импортировать их в основную программу.
3. Следует обратить внимание, что при показе заставки начинает игру только щелчок мыши на прямоугольнике заставки. Щелчок по игровому полю вне заставки должен игнорироваться.
4. Рекомендуемая схема управления игрой:
   1. <Space> - выстрел ракетой;
   2. стрелки влево, вправо – вращение корабля;
   3. стрелка вверх – включение двигателя (ускорение) корабля.
5. Все параметры игры должны быть оформлены как глобальные переменные в определенном месте программы и должным образом прокомментированы.
6. Перед началом написания кода программы следует провести объектно-ориентированное проектирование, разработать иерархию классов, выделить вспомогательные функции и функции-обработчики событий. Все функции, классы и методы должны быть тщательно продокументированы. Также для них должны быть выбраны короткие, но понятные имена, поясняющие их суть.

Дополнительные задания:

1. Привести код программы в соответствие с PEP8 – соглашением о стиле кода в языке Python.
2. Любые дополнения и усложнения механики и физики игры оцениваются в соответствии со сложностью реализации и соответствием задумке игры.

### 