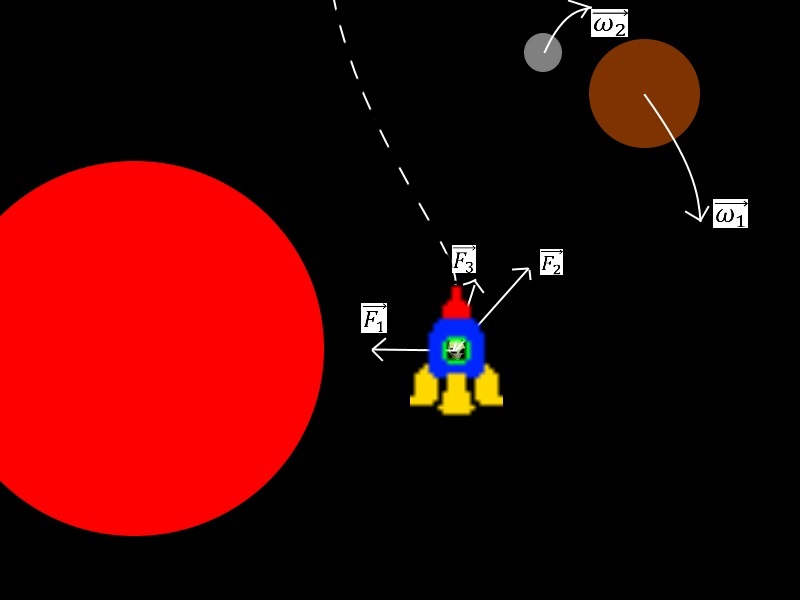
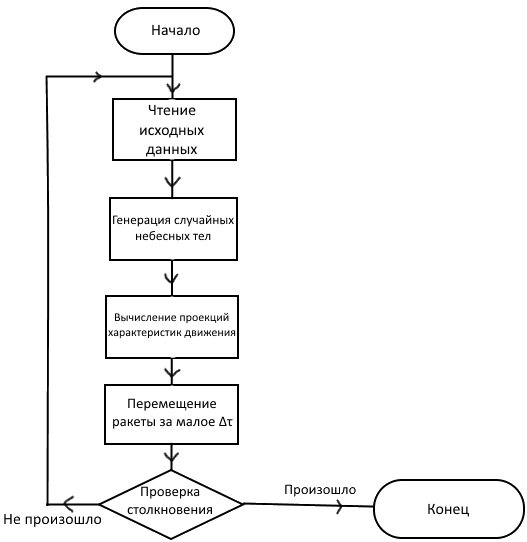
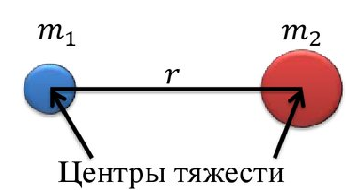
1. Постановка задачи  
   Проект — 2D игра, в которой космический корабль будет лететь сквозь недры космоса, встречая на своём пути различные препятствия в виде планет, звёзд, астероидов и даже (возможно) чёрных дыр, которые будут физически взаимодействовать с данным космическим кораблём (закон всемирного тяготения) и между собой (законы Кеплера — спутники планет). При этом для насыщенности игры будет отсутствовать или очень сильно сокращено взаимодействие между самими препятствиями (для того, чтобы размеры корабля были сравнимы с расстояниями между небесными телами). Управление — в двумерной системе координат, кроме того, объём топлива, доступный для маневрирования, будет ограничен.
2. Уточнение исходных и выходных данных  
   В качестве входных данных нас интересует получение сигнала с клавиатуры о нажатии клавиш ("←", "→", "↑", "↓"), которым будут соответствовать изменение горизонтальной координаты ракеты (считаем, что изменение горизонтальной скорости происходит моментально, когда это касается входных данных) и изменение вертикальной скорости - входные данные необходимо получать всё время работы программы. Выходными данными будет вывод на экран изображения космического корабля и небесных тел, влияющих на его движение. Кроме того, так как запас топлива ограничен, шкала его состояния также будет выведена на экран.
3. Выбор метода решения
   1. **Анализ используемой структуры данных**В качестве переменных нас интересуют как минимум координаты ракеты. Кроме того, так как планируется учитывание гравитационного взаимодействия ракеты и небесных тел, будут введены переменные, отображающие проекции скорости ракеты на горизонтальную и вертикальную оси. К тому же необходим двумерный массив, содержащий координаты небесных тел, их массы и радиусы. Из указанных переменных и массивов все, кроме скорости, должны быть целочисленными (так как координаты определяются целым числом - модули базисных векторов равны одному пикселю). Скорости для повышения точности определим типом данных double (так как таких переменных будет только две, экономия памяти не будет заметна, поэтому можно не использовать float).
   2. **Выбор метода решения**

Проверяем есть ли входные данные, получаем их, если есть. Просчитываем проекции трёх основных характеристик кинематики на вертикальную и горизонтальную оси — координаты, проекции вектора скорости и проекции вектора ускорения на основании входных данных (если таковые присутствуют) и влияния случайно появляющихся на экране небесных тел. Соответственно, генерирование этих самых случайных характеристик небесных тел и хранение их в двумерном массиве. Программа повторяется циклически пока не произойдёт столкновение ракеты и любого небесного тела

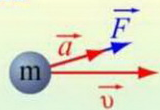
1. Составление алгоритма  
   
2. Математическая модель
   1. Закон всемирного тяготения

— модуль силы, действующей на тела,

m1, m2 — массы объектов, сила гравитационного взаимодействия между которыми рассматриваются,

r — расстояние между телами,

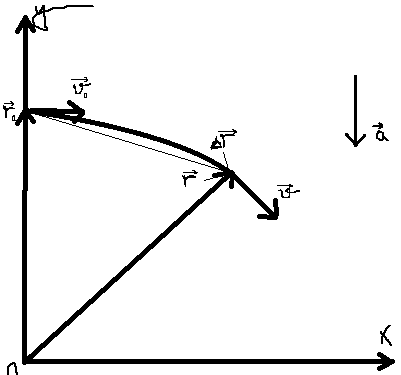
G — гравитационная постоянная, равная 6.67 × 10-11 м3 кг-1 с-2.

* 1. Второй закон Ньютона

— модуль силы, действующей на тело,

m — масса тела,

— модуль ускорения, с которым движется тело под действием данной силы.

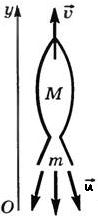
* 1. Преобразования Галилея

x,y,x0,y0 — координаты в данный момент времени и начальные координаты соответственно

— проекции начальной скорости на координатные оси

— проекции ускорения на координатные оси

— время, за которое это изменение произошло

* 1. Закон сохранения импульса

— модули начальной и конечной скоростей ракеты

*—* модуль скорости вылета топлива

— массы ракеты и выпущенного топлива соответственно

1. Листинг программы