**федеральное государственное автономное образовательное учреждение высшего образования**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**Факультет информационных технологий**

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 4**

**Дисциплина:** Физическое моделирование компьютерных игр\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Тема:** Моделирование движения по винтовой линии\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Выполнил(а): студент(ка) группы \_221-3710\_\_**

\_\_\_\_\_\_Пельт Сергей Александрович\_\_\_\_\_

(Фамилия И.О.)

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил: \_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_***

(Фамилия И.О., степень, звание) **(Оценка)**

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва2025**

**Моделирование движения по винтовой линии**

В лабораторной работе реализовано движение по винтовой траектории, сочетающее поступательное движение вдоль оси X с круговым движением в плоскости YZ. Ускорение изменяется линейно во времени, что усложняет поведение и делает его более реалистичным.

Использованные формулы:

1. **a(t) = A + B·t** — линейно изменяющееся ускорение;
2. **v(t) = v₀ + A·t + 0.5·B·t²** — скорость при таком ускорении;
3. **s(t) = v₀·t + 0.5·A·t² + (1/6)·B·t³** — путь при ускорении с линейной зависимостью;
4. **θ += ω·Δt**, где ω — угловая скорость;
5. **(y, z) = (R·cos(θ), R·sin(θ))** — положение на окружности;
6. **Δs = (vₓ + R·ω)·Δt** — длина участка винтовой линии за кадр.

Таким образом, реализовано трёхмерное винтовое движение объекта, в котором поступательное ускоренное движение по оси X сочетается с равномерным вращением вокруг неё, формируя винтовое движение.

**Листинг кода:**

void Update()

{

if (move)

{

// Увеличиваем общее прошедшее время

time += Time.deltaTime;

if (time < t)

{

pos = Vector2.zero;

linSpeed = 0;

acceleration = 0;

speed = 0;

}

else

{

float elapsedTime = time - t;

// Линейно изменяющееся ускорение: a(t) = A + B·t

acceleration = A + B \* elapsedTime;

// Линейная скорость вдоль оси X: v(t) = v0 + A·t + 0.5·B·t²

linSpeed = startLinSpeed + A \* elapsedTime + 0.5f \* B \* elapsedTime \* elapsedTime;

// Положение по X: s(t) = v0·t + 0.5·A·t² + (1/6)·B·t³

pos.x = startLinSpeed \* elapsedTime + 0.5f \* A \* elapsedTime \* elapsedTime + 1f / 6f \* B \* Mathf.Pow(elapsedTime, 3);

// Угловая скорость вокруг оси (по окружности): ω(t) = ω₀ + A·t + 0.5·B·t²

speed = startSpeed + A \* elapsedTime + 0.5f \* B \* elapsedTime \* elapsedTime;

// Изменяем угол на текущем кадре: θ += ω·Δt

angle += speed \* Time.deltaTime;

// Координаты на окружности в вертикальной плоскости (движение по окружности)

pos.y = radius \* Mathf.Cos(angle);

pos.z = radius \* Mathf.Sin(angle);

// Пройденное расстояние вдоль винтовой траектории: Δs = (vₓ + ω·R) · Δt

distance += (linSpeed + speed \* radius) \* Time.deltaTime;

transform.position = pos;

output.UpdateOutput(distance, linSpeed, acceleration, pos, time);

}

}

}

Ссылка на билд: <https://github.com/Sergey-Pelt/Physics>