**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 № 11**

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

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

**Тема:** Цилиндрические зубчатые передачи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

**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**

**Цилиндрические зубчатые передачи**

В лабораторной работе реализовано моделирование механического взаимодействия между шестернями, включая проверку согласованности модулей зацепления и расчёт передаточного отношения. В первом задании осуществляется вычисление угловой скорости и частоты вращения для пары шестерён, во втором — симуляция сложной передачи с несколькими шестернями, включая редуктор.

Особое внимание уделено корректной передаче угловой скорости между шестернями в зацеплении, а также соблюдению физически реалистичного направления вращения при смене оси. Реализована поддержка как параллельных, так и соосных соединений шестерён.

Используемые формулы:

* Модуль зацепления: **m = d / Z**
* Передаточное отношение: **u = Z2 / Z1**
* Угловая скорость ведомой шестерни: **w2 = -w1 / u,**
* Частота вращения:  
  **f = |w| / 360**
* Для нескольких шестерён:  
  **w[i] = -w[i-1] / (Z[i] / Z[i-1]),**  
  если шестерни не на одной оси, и  
  **w[i] = w[i-1],**  
  если находятся на одной оси

**Листинг кода:**

bool ValidateModule() // проверка модулей

{

if (isTask1)

{

if (m == 0 && startU == 0)

{

try

{

float m1 = d1 / Z1; // модуль первой шестерни

float m2 = d2 / Z2; // модуль второй шестерни

if (m1 == 0f || !Mathf.Approximately(m1, m2))

{

return false;

}

else

{

return true;

}

}

catch

{

return false;

}

}

else

{

return true;

}

}

else

{

try

{

for (int i = 0; i < gears.Count - 1; i++)

{

float m1 = gears[i].d / gears[i].Z; // модуль текущей шестерни

float m2 = gears[i + 1].d / gears[i + 1].Z; // модуль следующей шестерни

if (m1 == 0f || !Mathf.Approximately(m1, m2))

{

return false;

}

}

return true;

}

catch

{

return false;

}

}

}  
void CalculateRotations() // расчёт скорости вращения шестерней

{

if (isTask1)

{

if (u == 0f)

{

u = (float)Z2 / Z1; // передаточное отношение u = Z2 / Z1

}

w2 = -w1 / u; // угловая скорость второй шестерни w2 = -w1 / u

frequency = Mathf.Abs(w2 / 360); // частота оборотов = |w2| / 360

output.UpdateOutput(Mathf.Abs(w2), frequency);

}

else

{

gears[0].w = w12;

for (int i = 1; i < gears.Count; i++)

{

if (gears[i].sameAxis)

{

gears[i].w = gears[i - 1].w; // шестерни на одной оси имеют одинаковую угловую скорость

}

else

{

float ratio = (float)gears[i].Z / gears[i - 1].Z; // передаточное число = Z текущей / Z предыдущей

gears[i].w = -gears[i - 1].w / ratio; // угловая скорость следующей = -предыдущая / передаточное число

}

}

w22 = gears[gears.Count - 1].w;

output.UpdateOutput(Mathf.Abs(w22));

}

}

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