Fotorealistyczna Grafika Komputerowa

Punkt, wektor, promień, prymityw, przecięcie

Adam Błaszczyk 239636
Antonina Matuszek 239687

Chapter 1

Indeks Hierarchiczny

1.1 Hierarchia klas

Klasy występujące w programie:

Chapter 2

Indeks Klas

2.1 Lista Klas

Lista klas: :

Chapter 3

Dokumentacja Klas

3.1 Referencja klasy plane.Plane

Metody publiczne

- def **__init__** (self, normal_vector, d)
- def **__str__** (self) # Wyświetla atrybuty płaszczyzny
- def **get_intersection** (self, ray) # Zwraca punkt przecięcia płaszczyzny z promieniem

Atrybuty publiczne

- **normal_vector** # Wektor normalny
- **a** # Pierwsza współrzędna wektora normalnego
- **b** # Druga współrzędna wektora normalnego
- **c** # Trzecia współrzędna wektora normalnego
- **d** # Przesunięcie, wzdłuż normalnej, płaszczyzny od środka układu współrzędnych

Klasa zaimplementowana w pliku:

- plane.py

3.2 Referencja klasy ray.Ray

Metody publiczne

- def **__init__** (self, origin=Vec3(0, 0, 0), direction=Vec3(1, 1, 1), length=math.inf)
- def **__str__** (self) # Wyświetla atrybuty promienia
- def **is_point_on_ray** (self, point) # Sprawdza, czy punkt leży na promieniu
- def **set_direction** (self, new_direction) # Ustawia nowy wektor kierunkowy
- def **get_plane_intersection** (self, plane)
- def **get_sphere_intersections** (self, sphere)

Atrybuty publiczne

- **origin** # Początek promienia
- **direction** # Wektor kierunkowy
- **length** # Długość

Klasa zaimplementowana w pliku:

- ray.py

3.3 Referencja klasy sphere.Sphere

Metody publiczne

- def **__init__** (self, centre=Vec3(0, 0, 0), radius=math.inf)
- def **get_centre** (self) # Zwraca środek sfery
- def **get_radius** (self) # Zwraca promień sfery
- def **surface_area** (self) # Zwraca pole powierzchni sfery
- def **get_volume** (self) # Zwraca objętość sfery
- def **__str__** (self) # Wyświetla atrybuty sfery
- def **get_ray_intersections** (self, ray) # Zwraca punkty przecięcia sfery z promieniem

Atrybuty publiczne

- **centre** # Środek sfery
- **radius** # Promień sfery
- **area** # Pole powierzchni sfery
- **volume** # Objętość sfery

Klasa zaimplementowana w pliku:

- sphere.py

3.4 Referencja klasy tests.Test

Testy jednostkowe programu.

3.5 Referencja klasy vector.Vec2

Metody publiczne

- def **__init__** (self, x, y)
- def **__add__** (self, other) # Dodawanie
- def **__iadd__** (self, other) # Dodawanie w miejscu
- def **__sub__** (self, other) # Odejmowanie
- def **__isub__** (self, other) # Odejmowanie w miejscu
- def **__eq__** (self, other) # Porównanie
- def **__abs__** (self) # Wartość bezwzględna
- def **__ne__** (self, other) # Porównanie
- def **__neg__** (self) # Negacja
- def **__pos__** (self) # +x
- def **__str__** (self) # Wyświetla atrybuty wektora
- def **length** (self) # Zwraca długość wektora
- def **distance** (self, other) # Zwraca odległość między wektorem, a punktem
- def **__truediv__** (self, other) # Dzielenie
- def **__itruediv__** (self, other) # Dzielenie w miejscu
- def **__mul__** (self, other) # Mnożenie
- def **__imul__** (self, other) # Mnożenie w miejscu
- def **__rmul__** (self, other) # Mnożenie

Atrybuty publiczne

- **x** # Współrzędna x wektora
- **y** # Współrzędna y wektora

Klasa zaimplementowana w pliku:

- vector.py

3.6 Referencja klasy vector.Vec3

Metody publiczne

- def **__init__** (self, x, y, z)
- def **__add__** (self, other) # Dodawanie
- def **__iadd__** (self, other) # Dodawanie w miejscu
- def **__sub__** (self, other) # Odejmowanie
- def **__isub__** (self, other) # Odejmowanie w miejscu
- def **__eq__** (self, other) # Porównanie
- def **__abs__** (self) # Wartość bezwzględna
- def **__ne__** (self, other) # Porównanie
- def **__neg__** (self) # Negacja
- def **__pos__** (self) # +x
- def **__str__** (self) # Wyświetla atrybuty wektora
- def **length** (self) # Zwraca długość wektora
- def **distance** (self, other) # Zwraca odległość między wektorem, a punktem
- def **is_point_on_ray** (self, ray)
- def **__truediv__** (self, other) # Dzielenie
- def **__itruediv__** (self, other) # Dzielenie w miejscu
- def **__mul__** (self, other) # Mnożenie
- def **__imul__** (self, other) # Mnożenie w miejscu
- def **__rmul__** (self, other) # Mnożenie
- def **cross** (self, other) # Zwraca iloczyn wektorowy

Atrybuty publiczne

- **x** # Współrzędna x wektora
- **y** # Współrzędna y wektora
- **z** # Współrzędna z wektora

Klasa zaimplementowana w pliku:

- vector.py

Index