

# Spis treści

<b>1</b>	<b>Matematyka</b>	<b>2</b>
1.1	Koordinaty . . . . .	2
1.1.1	Kartezjańskie . . . . .	2
1.1.2	Biegunowe (2D) . . . . .	2
1.1.3	Sferyczne (3D) . . . . .	2
1.2	Operacje na punktach . . . . .	2
1.2.1	Skalowanie . . . . .	2
1.2.2	Obroty . . . . .	2
<b>2</b>	<b>Atrybuty</b>	<b>2</b>
2.1	Na klasę . . . . .	3
2.2	Na pole . . . . .	3

# 1 Matematyka

- skalar  $\alpha$  - liczba - punkt  $\bar{p}$  - pozycja - wektor  $\vec{v}$  - kierunek i długość

## 1.1 Koordynaty

### 1.1.1 Kartezjańskie

$$P(x, y, z)$$

### 1.1.2 Biegunowe (2D)

$$P(r, \alpha)P(x, y) = P(r * \cos \alpha, r * \sin \alpha)$$

### 1.1.3 Sferyczne (3D)

$$P(r, \alpha, \beta)P(x, y, z) = P(r * \cos \alpha * \cos \beta, r * \sin \alpha, r * \cos \alpha \sin \beta)$$

## 1.2 Operacje na punktach

### 1.2.1 Skalowanie

$p = (x, y, z, 1)$  Punkt 3ds =  $[sx, sy, sz]$  wektor skalip' =  $(x * sx, y * sy, z * sz)$  Punkt przeskalowany

Macierze:

$$[x', y', z', 1] = [x, y, z, 1] \begin{bmatrix} sx & 0 & 0 & 0 \\ 0 & sy & 0 & 0 \\ 0 & 0 & sz & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### 1.2.2 Obroty

Wokół X:

$$p' = (x, y * \cos \alpha - z * \sin \alpha, y * \sin \alpha + z \cos \alpha)$$

Macierz:

$$[x', y', z', 1] = [x, y, z, 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Wokół Y

$$p' = (z \sin \alpha + x \cos \alpha, y, z \cos \alpha - x \sin \alpha)$$

Macierz

$$[x', y', z', 1] = [x, y, z, 1] \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ \sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Wokół Z

$$p' = (x \cos \alpha - y \sin \alpha, x \sin \alpha + y \cos \alpha, z)$$

Macierz

$$[x', y', z', 1] = [x, y, z, 1] \begin{bmatrix} \cos \alpha & \sin \alpha & 0 & 0 \\ -\sin \alpha & \cos \alpha & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

## 2 Atrybuty

```
1 [Attribute(Params)]
2 public class Class {}
```

## 2.1 Na klasę

- AddComponentMenu("menu/name")

## 2.2 Na pole

- Range(float from, float to)