

# DU 4

March 13, 2023

Vizualizujte pomocou OpenGL pripad sikmeho vrhu, ktorý sme preberali na cviceni. Vstupne hodnoty budu veľkosť počiatočnej rýchlosti hmotného bodu  $v_0 > 0$ , uhol, ktorý zvierá vektor počiatočnej rýchlosti s vodorovnou rovinou  $\alpha \in (0^\circ, 90^\circ)$  a počiatočná výška, z ktorej je hmotný bod vrhnutý  $h_0 > 0$ . Ide o pohyb v 2D, čiže budeme potrebovať dve zložky vektorov polohy a rýchlosti:

$$x(t) = v_0 \cos(\alpha)t \quad (1)$$

$$y(t) = h_0 + v_0 \sin(\alpha)t - \frac{1}{2}gt^2 \quad (2)$$

$$v_x(t) = v_0 \cos(\alpha) \quad (3)$$

$$v_y(t) = v_0 \sin(\alpha) - gt \quad (4)$$

Takže z vypočítaných hodnôt  $x(t)$  a  $y(t)$  urobte posun hmotného bodu cez  $glTranslatef(x(t), y(t), 0)$ . Vizualizáciu ukončíte pri dopade hmotného bodu na zem ( $y = 0$ ). Zároveň vykreslite v každom momente vektor rýchlosti  $\vec{v} = (v_x(t), v_y(t))$  do miesta, kde sa nachádza hmotný bod (bude mať smer dotyčnice k trajektorii) a vykreslite aj krivku trajektorie od začiatku pohybu do aktuálneho miesta, kde sa hmotný bod práve nachádza. To môžete spraviť uložením všetkých dvojíc hodnôt  $x(t)$  a  $y(t)$  do pola a vykreslením čiary medzi jednotlivými týmito bodmi v každom kroku.

Pri tejto úlohe je dôležité nastaviť si správne rozmery scény, aby bolo celý pohyb dobre vidieť. Môžete to spraviť vypočítaním maximálnej výšky a maximálnej vzdialenosti, kam hmotný bod doletí:

$$v_{0y} = v_0 \sin(\alpha) \quad (5)$$

$$v_{0x} = v_0 \cos(\alpha) \quad (6)$$

$$HH = h_0 + \frac{v_{0y}^2}{2g} \quad (7)$$

$$LL = v_{0x} \frac{v_{0y} + \sqrt{v_{0y}^2 + 2hg}}{g} \quad (8)$$

Z týchto hodnôt môžete potom nastaviť rozmery okna a zobrazovanej scény napríklad takto:

Vo funkcii `main()`:

```
LL = maximalny x-ovy rozmer sceny
HH = maximalny y-ovy rozmer sceny

if (LL >= HH) {
    XNPix = 1080;
    YNPix = XNPix * HH/LL;
}
else {
    YNPix = 640;
```

```
        XNPix = YNPix * LL/HH;  
    }  
    glutInitWindowSize(XNPix, YNPix);
```

-----

Vo funkcii obsluhaResize():

```
gluOrtho2D(0.0, LL*1.1, 0.0, HH*1.1);
```