

DU 6

March 24, 2023

Vizualizujte pomocou OpenGL pripad sikmeho vrhu v 3D. V podstate je to to iste, ako pre dvojrozmerny pripad, len je tu este jeden uhol naviac - azimutalny uhol, natocenie v roviny $x - y$, uhol, ktorý zvierá priemet vektoru rychlosti do roviny $x - y$ s osou x . Vstupne parametre teda budu pociatocna vyska $z_0 > 0$, velkost pociatocnej rychlosti $v_0 > 0$, uhol nameru $\alpha \in (0^\circ, 90^\circ)$ a azimutalny uhol $\phi \in (0^\circ, 360^\circ)$. Rovnice pre suradnice a rychlosti vyzeraju nasledovne:

$$v_{0xy} = v_0 \cos(\alpha) \quad (1)$$

$$v_{0x} = v_{0xy} \cos(\phi) \quad (2)$$

$$v_{0y} = v_{0xy} \sin(\phi) \quad (3)$$

$$v_{0z} = v_0 \sin(\alpha) \quad (4)$$

$$x(t) = v_{0x}t \quad (5)$$

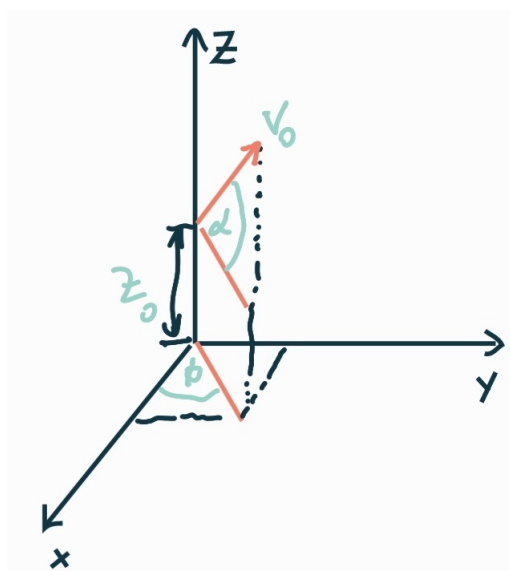
$$y(t) = v_{0y}t \quad (6)$$

$$z(t) = z_0 + v_{0z}t - \frac{1}{2}gt^2 \quad (7)$$

$$v_x(t) = v_{0x} \quad (8)$$

$$v_y(t) = v_{0y} \quad (9)$$

$$v_z(t) = v_{0z} - gt \quad (10)$$



Pri pohybe bodu zobrazte aj trajektoriu (celu parabolicku krivku) a aj vektor rychlosti v mieste aktualnej pozicie hmotneho bodu (ako v DU4). V tomto pripade byva vizualizacia trochu neprehladna, takže zobrazte aj osi x , y a z , rovinu zeme $z = 0$ a aj pociatocnu vysku z_0 v pociatku suradnicoveho systemu (napríklad uzky kvadrom vysky z_0). Zaroven pri pohybe zobrazte v suradnici $(x(t), y(t), 0)$ nejaky malý tmavý disk, nieco ako tien letiaceho objektu, kvôli predstave, ako vysoko nad zemou sa objekt nachadza. Pohyb bodu zastavte v momente dopadu na zem. Dobré je v tomto pripade pouzivat funkciu *gluLookAt()*, ktoru som vysvetloval.

Mozete napríklad nastaviť kameru tak, aby sa objekt pohyboval vždy pozdĺž obrazovky zľava do prava (vlastne to bude vyzerat, ako predchádzajúci 2D prípad), alebo sa kamera môže postupne otáčať za letiacim bodom. Zadávanie vstupných parametrov môžete spraviť aj interaktívne, že pomocou sipiek budete meniť α a ϕ , ďalšími dvoma klavesami nastavíte z_0 a potom ďalšími v_0 (prípadne podľa toho, ako dlho podržíte klaves bude narastať v_0 a po jeho pustení bude bod vrhnutý). V tomto prípade nech sa potom zobrazuje počiatočný smer a veľkosť vektora rýchlosti v mieste $(0, 0, z_0)$, Nech je jasné, ktorým smerom je natočený a aký je veľký.

Este jedna vec k 3D zobrazeniu. Objekty sa vám teraz zobrazujú v poradi, v akom ich zadávate do scény. Neberie sa ešte do úvahy z -ová súradnica, program nekontroluje, ktorý objekt je bližšie ku kamere. Treba to zapnúť nasledovne:

Vo funkcii `main()` na začiatku, kde inicializujete veci z OpenGL:

```
glutInitDisplayMode(GLUT_DEPTH | GLUT_DOUBLE);  
glEnable(GL_DEPTH_TEST);
```

Vo funkcii, kde kreslite scénu (zadáвате jednotlivé objekty) úplne na začiatku musí byť:

```
glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
```