

1. Obravnavaj met žogice iz višine H v smeri vektorja $\mathbf{v}_0 = \{v_{0x}, v_{0y}\}$. Upoštevaj gravitacijski pospešek $g = 9.81 \text{ m/s}^2$. Pomagaj si z naslednjimi koraki:

- Obravnavaj problem v ravnini, kjer je os x smer meta, os y pa predstavlja višino. Sestavi vektorsko (2D) funkcijo $\mathbf{v}[t_]$, ki za dan čas t izračuna vektor hitrosti pri metu. Izberi naslednje parametre:

$\mathbf{v}_0 = \{10, 3\}$

$g = 9.81$

$H = 10$

- Vemo, da velja $\vec{x}(t) = \vec{x}(0) + \vec{v}(0)t + \frac{at^2}{2}$. Če veš, da je $\vec{x} = \mathbf{x}_0 = \{0, H\}$, sestavi še vektorsko funkcijo $\mathbf{X}[t_]$, ki vrne položaj žogice ob času t
- Sestavi funkcijo `SlikaTocke[t_]`, ki nariše žogo kot točko. Velikost točke naj bo 3% slike.
- Sestavi funkcijo vektorja hitrosti v odvisnosti od časa $\mathbf{V}[t_]$.
- Sestavi funkcijo `SlikaVektorja[t_, k_]`, ki nariše vektor hitrosti iz točke pomnožen s skalarjem k . Uporabi grafični objekt `Arrow`. Debelina črte naj bo 0,5% velikosti slike.
- Nariši sliko žoge in vektorja po eni sekundi. Na sliki naj se nariše koordinatni sistem, os x nastavi na interval $[-1, 25]$, os y pa na interval $[-5, 15]$. Velikosti enot na obeh oseh naj bosta enaki (Namig: preveri delovanje naslednjih opcij v objektu `Graphics`: `Axes`, `PlotRange` in `AspectRatio`).
- S pomočjo funkcije `Manipulate` naredi animacijo leta žoge. Čas t premikaj o 0 do 5s.
- Izračunaj čas padca žoge. Izračunaj še vrednost v smeri x , kjer žoga pade.
- Izračunaj čas doseganja največje višine leta ter maksimalno višino.
- Uporabi izračuna iz zgornjih dveh točk tako, da bo območje izrisa čimbolj primerno (na osi x od -1 do malce več kot lokacija padca; na osi y pa naj bo od nekaj pod ničlo do malce več kot maksimalna višina leta; čas animacije naj bo od 0 do trenutka padca).
- Kakšna je absolutna vrednost hitrosti v trenutku, ko žogica zadene tla?

2. Ravnino podamo v obliki `Ravnina[n_, v_]`, kjer je \mathbf{n} normala na ravnino in je \mathbf{v} točka na ravnini. Npr. ravnina, ki gre skozi točko $(1, 1, 1)$ in je pravokotna na glavno diagonalo je `r111 = Ravnina[{-1,-1,-1}, {1,1,1}]`.

- Sestavi funkcijo `Slika[Ravnina[n_, v_]]`, ki pripravi 3D grafični objekt. Oglej si grafični objekt `Hyperplane`.
- Sestavi funkcijo `Format[r_Ravnina]`, ki nariše ravnino. Pozor: če definiramo funkcijo `Format` na objektih, se ti objekti "sami" izrišejo.
- Sestavi še priročni funkciji `Normala[Ravnina[n_, v_]]` in `Tocka[Ravnina[n_, v_]]`, ki vrneta točko in normalo ravnine.
- Definiraj koordinatne ravnine `rx`, `ry`, `rz`, ki gredo skozi točko 0 in so njihove normale enotski vektorji koordinatnih osi.
- Sestavi funkcijo `SlikaNormale[Ravnina[n_, v_]]`, ki vrne grafični objekt slike normale. Uporabi grafični objekt `Arrow`.

- Nariši vse štiri ravnine `rx`, `ry`, `rz`, `r111` na isti sliki skupaj z normalami.
- Sestavi funkcijo `Enacba[Ravnina[n_, v_]]`, ki vrne enačbo ravnine.
- Sestavi funkcijo `ResiSistem[sistem_List]`, ki za dan sistem treh enačb v spremenljivkah x , y in z vrne točko rešitve. Predpostavi, da je sistem takšen, da vrne samo eno rešitev.
- Sestavi funkcijo `Presecisca[ravnine_List]`, ki za dan seznam ravnin poišče vse podmnožice velikosti 3 in potem poišče vse preseke treh ravnin iz teh podmnožic. Uporabi funkcijo `Subset` ter funkciji iz prejšnjih dveh nalog.
- Sestavi funkcijo `Vsebuje[Ravnina[n_, v_], x_, y_, z_, eps_:0.000001]`, ki s toleranco 10^{-6} natančno ugotovi, ali je točka v ravnini. Uporabi definicijo ravnine $\vec{n} \cdot (\vec{r}_0 - \vec{r}) = 0$, kjer za vrednost na levi strani namesto enačenja z nič dopuščaš odmik za ϵ .
- Sestavi funkcijo `Trikotnik[r_Ravnina, tocke_List]`, ki za dano ravnino in seznam točk nariše trikotnik na točkah, ki so vsebovane v ravnini. Predpostavimo, da bodo izmed točk podanih v seznamu točke, vedno natanko tri na ravnini - katere tri pa moramo izračunati. Izračunaj še vsa presečišča štirih ravnin (uporabi zgornjo funkcijo) ter trikotnik za ravnino `rx`.
- Sestavi funkcijo `SlikaTrikotnikov[r1_, r2_, r3_, r4_]`, ki sestavi sliko vseh štirih trikotnikov, ki so definirani s preseki ravnin.

3. Izberi si neko 3D telo in ga nariši s pomočjo točk in poligonov. Enostavni primeri so kocke, kvadri. Bolj napredni so piramide, prizme, prerezane piramide... Bodite izvirni in narišite nekaj lepega. Narišite tudi normalo nad eno ploskvijo, ki gleda ven iz telesa iz težišča ploskve.