



Računalna grafika – laboratorijske vježbe 5

Damir Horvat

Fakultet organizacije i informatike, Varaždin



Sadržaj

1 Stožac

Modeli stošca

Opis implementacije

2 Valjak

Modeli stošca

Opis implementacije

3 Kugla

Modeli kugle

Opis implementacije

4 Žičani model parametarske plohe



Primjer 1. – Stožac

- U klasu Persp dodajte metodu za crtanje stošca

`stozac(r, h, n)`

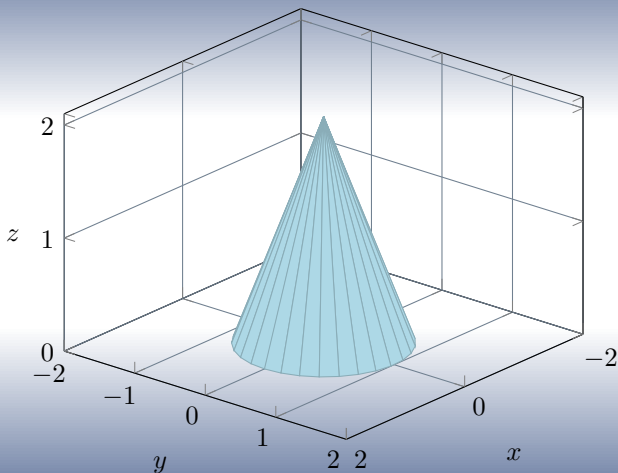
gdje je r polumjer baze, h visina stošca, a n broj segmenata (u ovom slučaju trokuta) i linija koje čine plašt. Bazu nacrtajte u xy -ravnini, a vrh stošca na koordinati $(0, 0, h)$.

- Napravite animaciju u kojoj kamera kruži oko stošca i svaki puta nakon što prijeđe puni kut pomakne se na veću ili manju visinu ovisno o trenutnom smjeru promjene visine.
- Visinu kamere ograničite unutar nekog segmenta, npr. $[0.2, 10]$. Visina kamere se naizmjenično mijenja unutar zadanih granica.
- Dodatak:** pomoću slidera omogućite mijenjanje polumjera baze stošca r , visine stošca h , broja segmenata n i udaljenost kamere od stošca.



Stožac

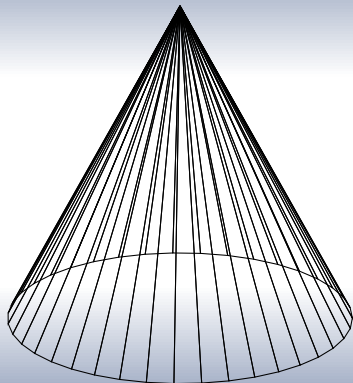
Stožac u 3D sceni





Stožac

Žičani model stošca





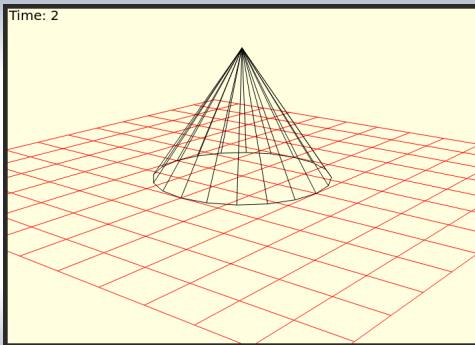
Ideja implementacije žičanog modela stošca

- Baza stošca leži u xy -ravnini. Točke ruba baze leže na kružnici polumjera r sa središtem u ishodištu.
- Sve točke na rubu baze su oblika $(r \cos \varphi, r \sin \varphi, 0)$ za $\varphi \in [0, 2\pi)$.
- Vrh stošca je u točki $(0, 0, h)$.
- Trebamo izgenerirati određeni broj točaka na rubu baze stošca, tj. ukupno n takvih točaka. Postavimo parametar φ na vrijednost 0 i u svakom sljedećem koraku povećamo njegovu vrijednost za $\frac{2\pi}{n}$. Na taj način nakon provedenih n koraka dobivamo ukupno n točaka na rubu baze stošca.
- Svake dvije uzastopne točke na rubu baze trebamo spojiti linijom i na taj način dobivamo pravilni n -terokut koji je aproksimacija kružnice.
- Točku $(0, 0, h)$ spojimo linijom sa svakom od n točaka na rubu baze stošca.



Stožac

Screenshot animacije stošca



► [Link na animaciju](#)



Primjer 2. – Valjak

- U klasu Persp dodajte metodu za crtanje valjka

`valjak(r, h, n)`

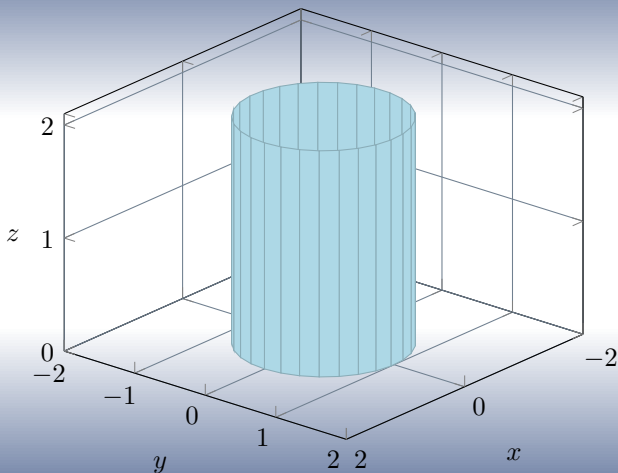
gdje je r polumjer baze, h visina valjka, a n broj segmenata (u ovom slučaju pravokutnika) i linija koje čine plašt. Donju bazu nacrtajte u xy -ravnini.

- Napravite animaciju u kojoj kamera kruži oko stošca i svaki puta nakon što prijeđe puni kut pomakne se na veću ili manju visinu ovisno o trenutnom smjeru promjene visine.
- Visinu kamere ograničite unutar nekog segmenta, npr. $[0.2, 10]$. Visina kamere se naizmjenično mijenja unutar zadanih granica.
- Dodatak:** pomoću slidera omogućite mijenjanje polumjera baze valjka r , visine valjka h , broja segmenata n i udaljenost kamere od valjka.



Valjak

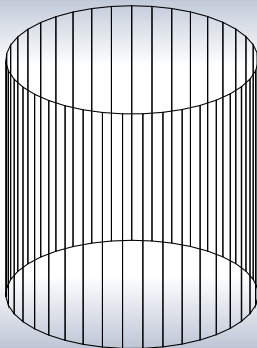
Valjak u 3D sceni





Valjak

Žičani model valjka





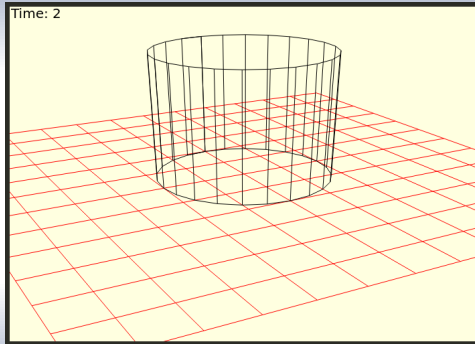
Ideja implementacije žičanog modela valjka

- Donja baza valjka leži u ravnini $z = 0$, a gornja baza u ravnini $z = h$.
- Sve točke na rubu donje baze su oblika $(r \cos \varphi, r \sin \varphi, 0)$, a na rubu gornje baze su oblika $(r \cos \varphi, r \sin \varphi, h)$ za $\varphi \in [0, 2\pi)$.
- Trebamo izgenerirati n točaka na rubu donje baze i n točaka na rubu gornje baze valjka. Postavimo parametar φ na vrijednost 0 i u svakom sljedećem koraku povećamo njegovu vrijednost za $\frac{2\pi}{n}$. Na taj način nakon provedenih n koraka dobivamo ukupno $2n$ traženih točaka.
- Svake dvije uzastopne točke na rubu donje baze trebamo spojiti linijom. Svake dvije uzastopne točke na rubu gornje baze trebamo spojiti linijom.
- Svaku točku donje baze treba spojiti linijom s odgovarajućom točkom na gornjoj bazi. Odgovarajuće točke su one točke koje su dobivene za istu vrijednost parametra φ .



Valjak

Screenshot animacije valjka



► [Link na animaciju](#)



Primjer 3. – Kugla

- U klasu Persp dodajte metodu za crtanje kugle

`kugla(r, m, n)`

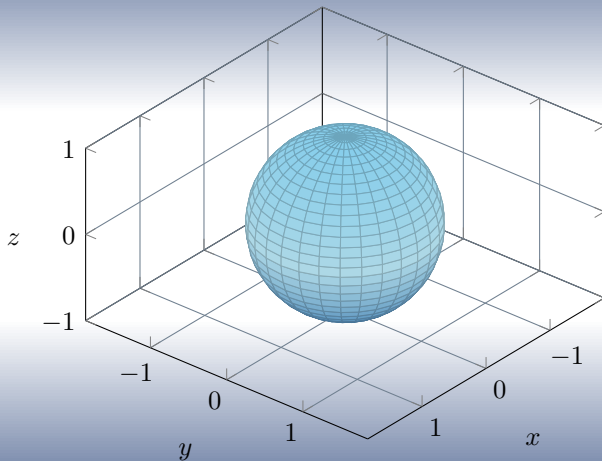
gdje je r polumjer kugle, m broj meridijana, a n broj paralela. Središte kugle je u ishodištu, a ekvator u xy -ravnini.

- Napravite animaciju u kojoj kamera kruži oko kugle i svaki puta nakon što prijeđe puni kut pomakne se na veću ili manju visinu ovisno o trenutnom smjeru promjene visine.
- Visinu kamere ograničite unutar nekog segmenta, npr. $[0.2, 7]$. Visina kamere se naizmjenično mijenja unutar zadanih granica.
- **Dodatak:** pomoću slidera omogućite mijenjanje polumjera kugle r , broja meridijana m , broja paralela n i udaljenost kamere od kugle.



Kugla

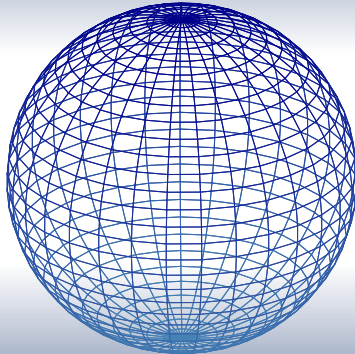
Kugla u 3D sceni





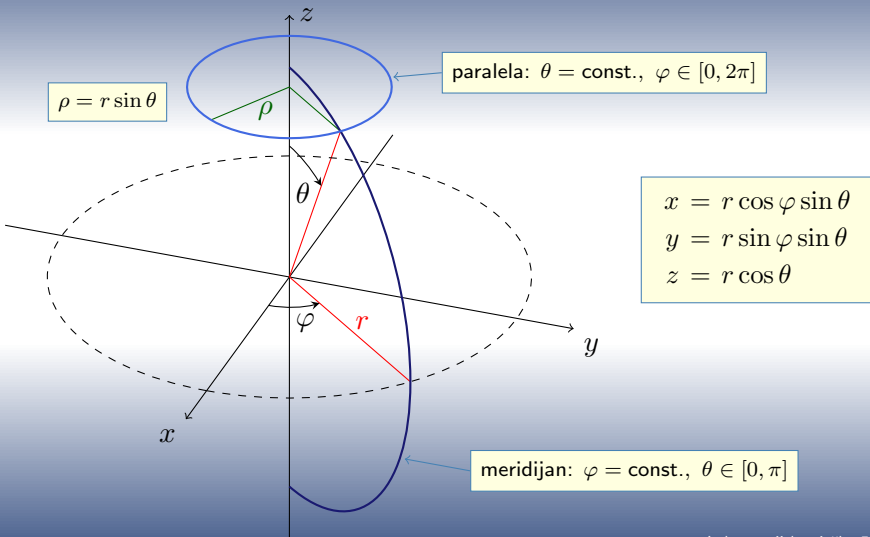
Kugla

Žičani model kugle





Parametarske jednadžbe sfere (rubu kugle)





Ideja implementacije žičanog modela kugle

- Trebamo nacrtati m meridijana. S vanjskom petljom mijenjamo vrijednost parametra φ s pomakom $\frac{2\pi}{m}$ unutar intervala $[0, 2\pi)$. Za svaku takvu fiksnu vrijednost parametra φ crtamo meridijan tako da s unutarnjom petljom s određenim pomakom mijenjamo vrijednost parametra θ unutar segmenta $[0, \pi]$ generirajući točke

$$(r \cos \varphi \sin \theta, r \sin \varphi \sin \theta, r \cos \theta)$$

pri čemu susjedne točke spajamo linijom.



Ideja implementacije žičanog modela kugle

- Trebamo nacrtati n paralela. S vanjskom petljom mijenjamo vrijednost parametra θ s pomakom $\frac{\pi}{n+1}$ unutar intervala $\langle 0, \pi \rangle$. Za svaku takvu fiksnu vrijednost parametra θ crtamo paralelu tako da s unutarnjom petljom s određenim pomakom mijenjamo vrijednost parametra φ unutar segmenta $[0, 2\pi]$ generirajući točke

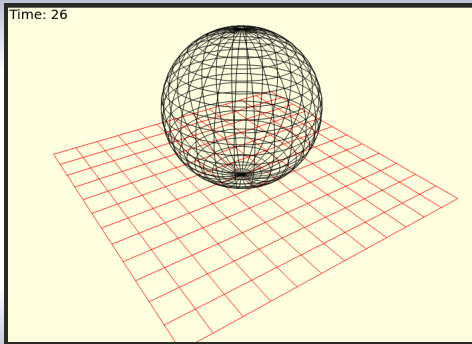
$$(r \cos \varphi \sin \theta, r \sin \varphi \sin \theta, r \cos \theta)$$

pri čemu susjedne točke spajamo linijom.



Kugla

Screenshot animacije kugle



► [Link na animaciju](#)

Dodatak



Napomena

- Crtanje žičanog modela plohe koja je zadana parametarskim jednadžbama

$$x = x(u, v), \quad y = y(u, v), \quad z = z(u, v)$$

obavlja se na isti način kako je opisano kod žičanog modela sfere (ruba kugle) koja je također bila zadana u parametarskom obliku.

- U žičanom modelu parametarske plohe treba nacrtati njezine parametarske u -crte i v -crte s određenim pomakom uz pretpostavku da se radi o plohi koja ima “lijepu sliku”.



Napomena

- **Parametarske u -crte.** S vanjskom petljom mijenjamo vrijednost parametra v s određenim pomakom unutar segmenta $[v_{\min}, v_{\max}]$. Za svaku takvu fiksnu vrijednost parametra v crtamo u -crtu tako da s unutarnjom petljom s određenim pomakom mijenjamo vrijednost parametra u unutar segmenta $[u_{\min}, u_{\max}]$ generirajući točke

$$(x(u, v), y(u, v), z(u, v))$$

pri čemu susjedne točke spajamo linijom.



Napomena

- **Parametarske v -crte.** S vanjskom petljom mijenjamo vrijednost parametra u s određenim pomakom unutar segmenta $[u_{\min}, u_{\max}]$. Za svaku takvu fiksnu vrijednost parametra u crtamo v -crtu tako da s unutarnjom petljom s određenim pomakom mijenjamo vrijednost parametra v unutar segmenta $[v_{\min}, v_{\max}]$ generirajući točke

$$(x(u, v), y(u, v), z(u, v))$$

pri čemu susjedne točke spajamo linijom.



Zadatak

Zadatak – Parametarske jednadžbe plohe

- Napravite žičani model plohe koja je zadana parametarskim jednadžbama

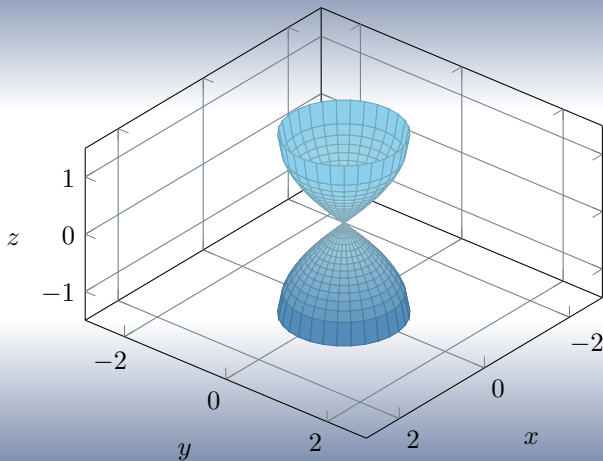
$$x = u \sin v, \quad y = u \cos v, \quad z = \arcsin u$$

pri čemu je $u \in [-1, 1]$, $v \in [0, 2\pi]$.

- Napravite animaciju u kojoj kamera kruži oko plohe i svaki puta nakon što prijeđe puni kut pomakne se na veću ili manju visinu ovisno o trenutnom smjeru promjene visine.
- Visinu kamere ograničite unutar nekog segmenta, npr. $[0.2, 7]$. Visina kamere se naizmjenično mijenja unutar zadanih granica.
- **Dodatak:** pomoću slidera omogućite mijenjanje gustoće parametarskih u -crta i v -crta te udaljenosti kamere od plohe.



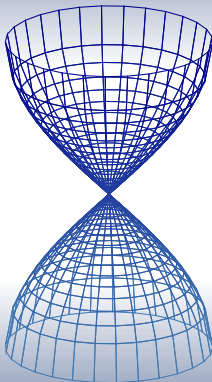
Promatrana ploha u 3D sceni





Žičani model parametarske plohe

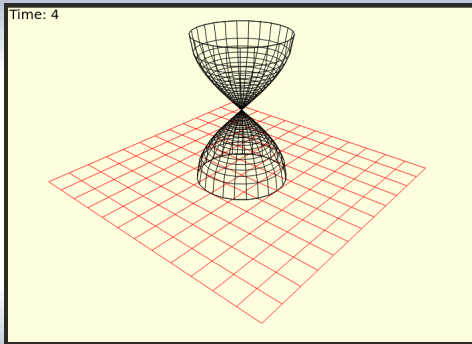
Žičani model promatrane plohe





Žičani model parametarske plohe

Screenshot animacije plohe



► [Link na animaciju](#)