

Duomenų struktūros ir algoritmai

13 paskaita

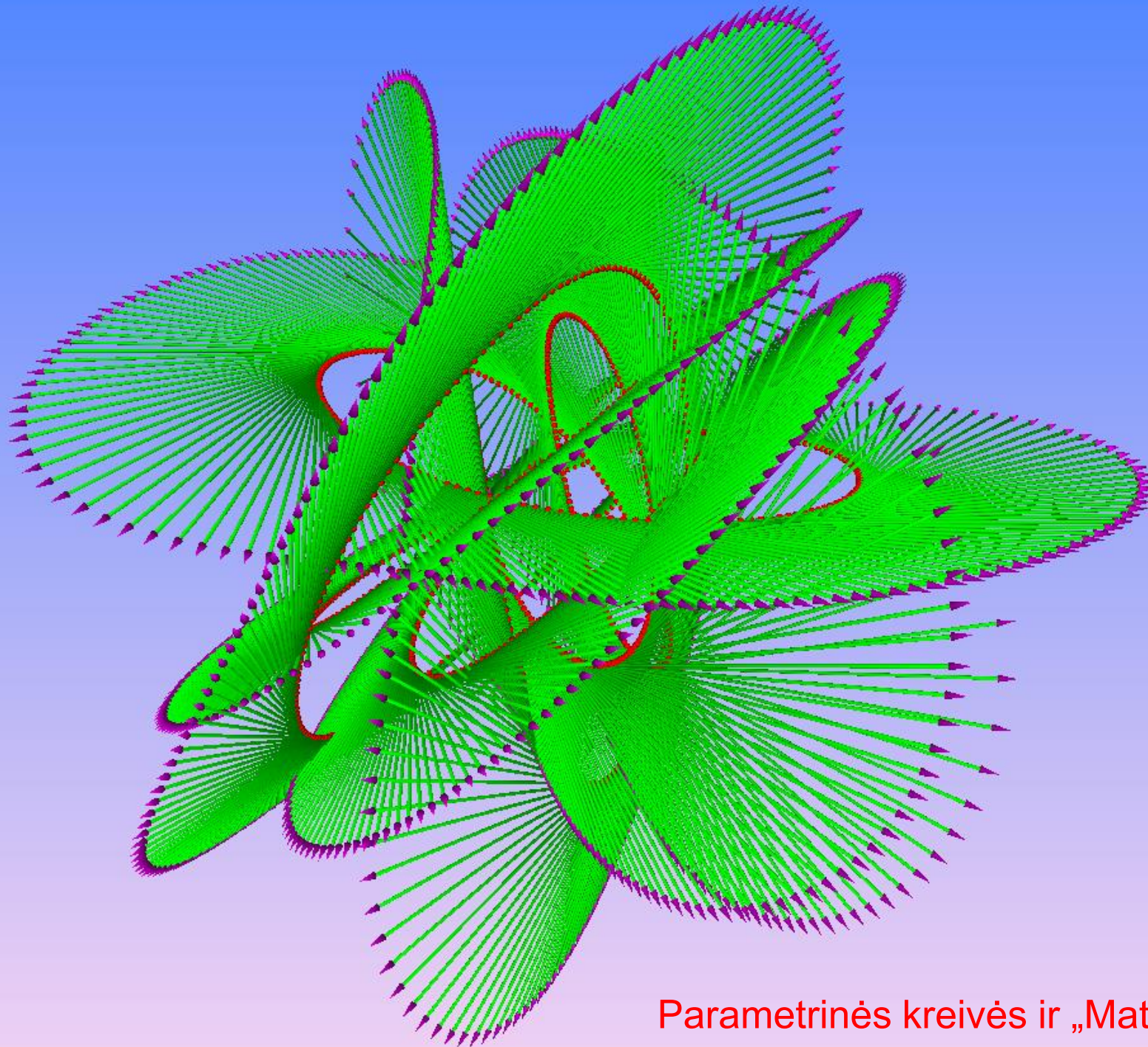
2020-05-06

Svarbi informacija

- Gegužės 13 d. paskaitos nebus (dėl konferencijos Rumunijoje).
- Gegužės 20 d. paskaita skirta Jūsų pristatymams.
- Gegužės 27 d. paskaita skirta Jūsų 3D modelių pristatymui ir 5 gražiausių išrinkimui (+0,25 balo premija 5 išrinktiesiems modelių autoriams).
- Atsiskaityti praktines užduotis galima iki gegužės galo.
- **Egzaminas birželio 19 d. 12 val. nuotoliniu būdu**
- **Egzamino trukmė – 2 val.**

13 paskaitos tikslas

- Apžvelgti modulio `add.py` naujas funkcijas (1.2a versija)
- Plačiau susipažinti su kūrybine užduotimi ir naudinga medžiaga 3D modelio kūrimui:
 - Parametrinės kreivės:
 - Apskritimas;
 - Elipsė;
 - Beziė kreivės.
 - Parametriniai paviršiai:
 - Sfera, cilindras, toras, sukinytis ir kt.
 - Beziė paviršiai.
 - Paviršių dalijimo algoritmai:
 - Catmullio ir Clarko paviršių dalijimo algoritmas.



Parametrinés kreivés ir „Math art“

1.2a modulio off.py funkcijos

def stretch(M,s): # funkcija labai panaši į zoom(M,s), tik čia s parametras turi 3 reikšmes $s=[a,b,c]$, pagal kurias atitinkamai keičiamas mastelis X, Y ir Z ašių kryptimis.

def curve(P,min_t,max_t,grid_t,k,r,RGB,isConnected):

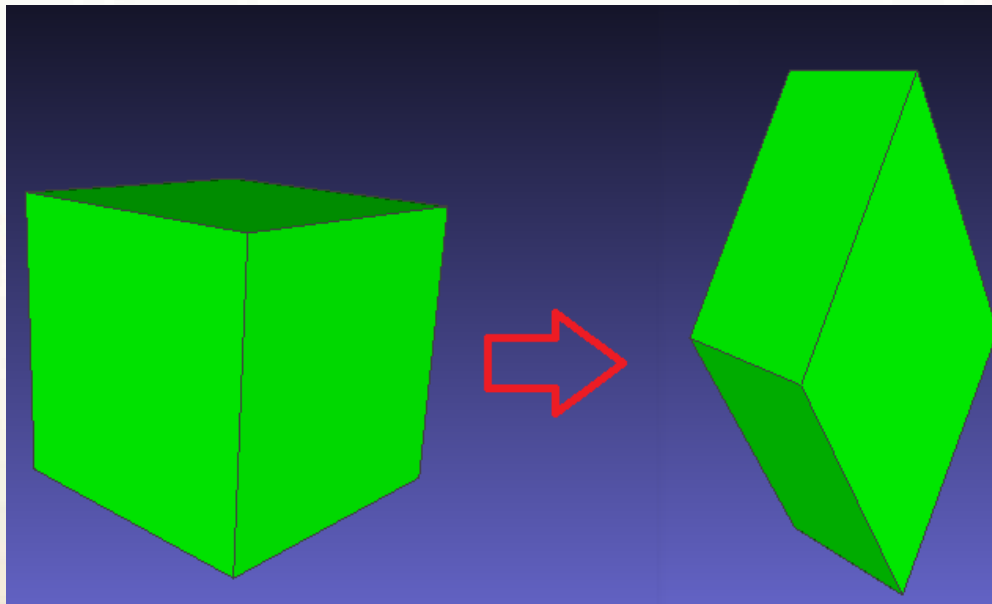
pagrindiniame sluoksnyje funkcija grąžina parametrinę kreivę, kur $P=P(t)$ – parametrinės kreivės lygtis, $t \in [\min_t, \max_t]$,
grid_t – parametrinės kreivės detalumas t parametro atžvilgiu,
k – parametrinę kreivę sudarančių apskritimų detalumas,
r – parametrinę kreivę sudarančių apskritimų spindulys (gali būti ir kintantis spindulys $r=r(t)$), RGB – parametrinės kreivės spalva,
isConnected – parametrai priskiriamos reikšmės true arba false atitinkamai jei kreivės galai sutampa ir jei nesutampa.

stretch(M,s) funkcija

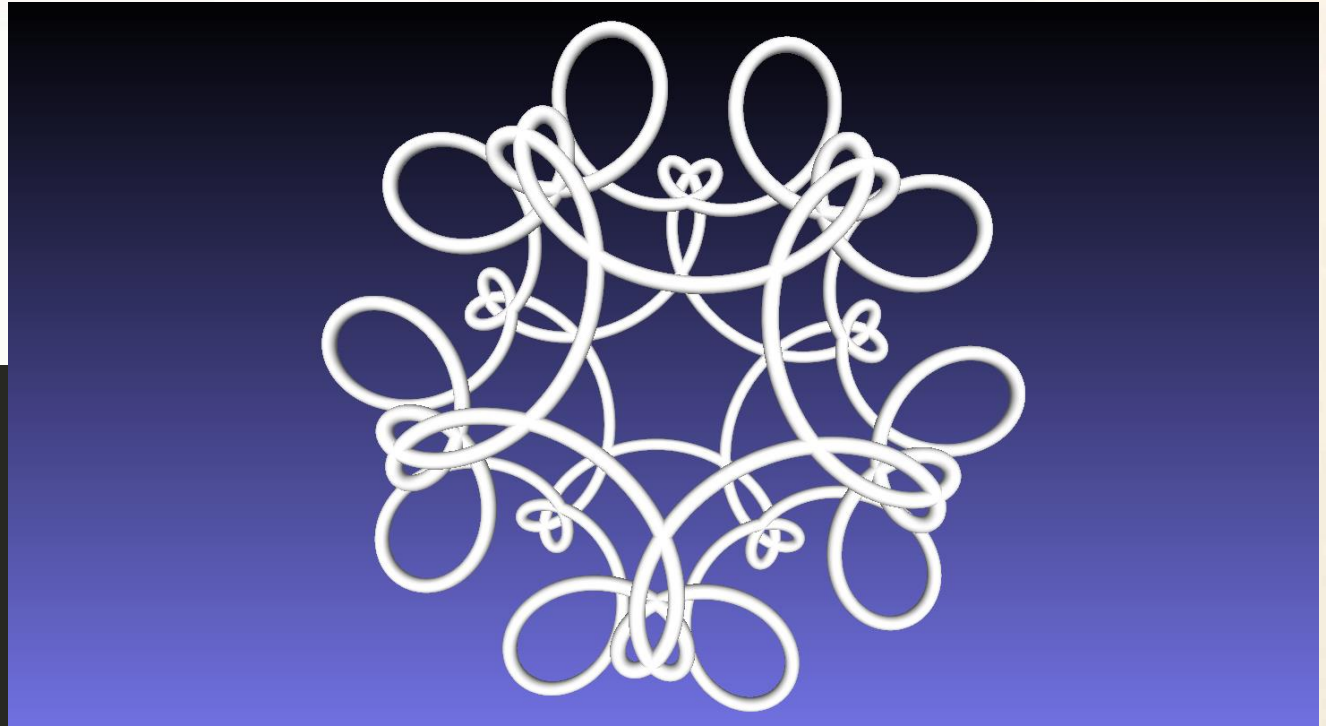
```
1 import add
2 add.cube([0,0,0],5,[0,255,0])
3 add.off('kubas.off')
```



```
1 import add
2 import math
3 add.cube([0,0,0],5,[0,255,0])
4 K = add.layer()
5 K = add.rotateX(K,math.pi/4,[0,0,0])
6 K = add.stretch(K,[1,2,1])
7 add.mesh(K)
8 add.off('kubas.off')
```

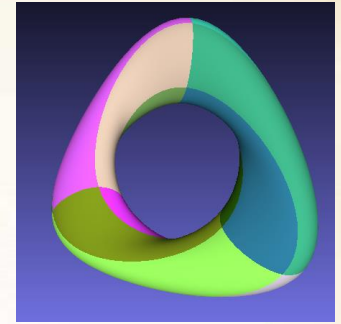


curve(P,min_t,max_t,grid_t,k,r,RGB,isConnected) funkcija



```
1 import add
2 import math
3 a1 = 5
4 b1 = 10
5 s1 = 0.4
6 v1 = -1
7 a2 = 2.1
8 b2 = 2.1
9 s2 = 5.2
10 v2 = 0.2
11 def P(t):
12     x = a1*math.cos(s1*t)*math.sin(v1*t)+b1*math.sin(s1*t)*math.cos(v1*t)+
13         a2*math.cos(s2*t)*math.sin(v2*t)+b2*math.sin(s2*t)*math.cos(v2*t)
14     y = 5*math.sin(t)
15     z = -a1*math.sin(s1*t)*math.sin(v1*t)+b1*math.cos(s1*t)*math.cos(v1*t)
16         -a2*math.sin(s2*t)*math.sin(v2*t)+b2*math.cos(s2*t)*math.cos(v2*t)
17     return [x,y,z]
18 add.curve(P,0,10*math.pi,3000,20,0.25,[255,255,255],True)
19 add.off('kreive1.off')
```

3D modelio kūrimas



UŽDUOTIS:

- Sukurti 3D modelį naudojant tik [pirminį programos tekstą](#).
- Modelio failo [formatas](#) – „[OFF](#)“.
- Modelį galima kurti grupėje iki 3 studentų.

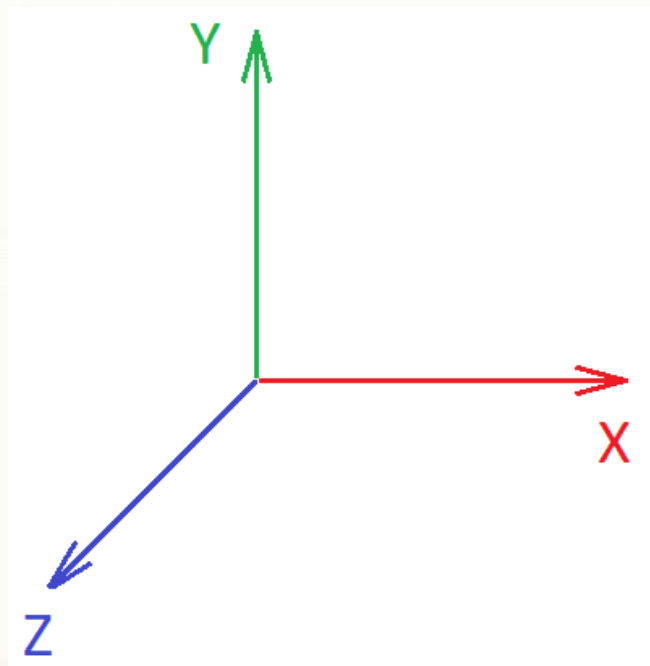
REIKALAVIMAI:

- Gautas 3D modelis turi būti sudarytas bent iš 1000 daugiakampių.
- Sukurtame algoritme privalo būti bent 1 parametras, nuo kurio priklausytų 3D modelio forma.
- Negalima naudoti 3D modeliavimui skirtų programų.
- Keičiant parametrus sukurtas 3D modelis negali sutapti su kursiojų modeliais arba paskaitų metu nagrinėjamais pavyzdžiais.

Vertinimo strategija

- Jei sukurtas 3D modelis tenkins reikalavimus, už jį bus skirta **po 1 balą kiekvienam** šio modelio **autoriui** (atsiskaitymas pratybų metu). Taip pat bus suteikta galimybė dalyvauti gražiausio 3D modelio konkurse.
- Sukurtų 3D modelių pristatymas, gražiausio modelio rinkimas – gegužės 27 d. (per teorijos paskaitą).
- Jūsų pačių išrinkti 5 geriausi modeliai atskirai bus įvertinti **0.25** premija. Jei keli autoriai, premija dalinama lygiomis dalimis.

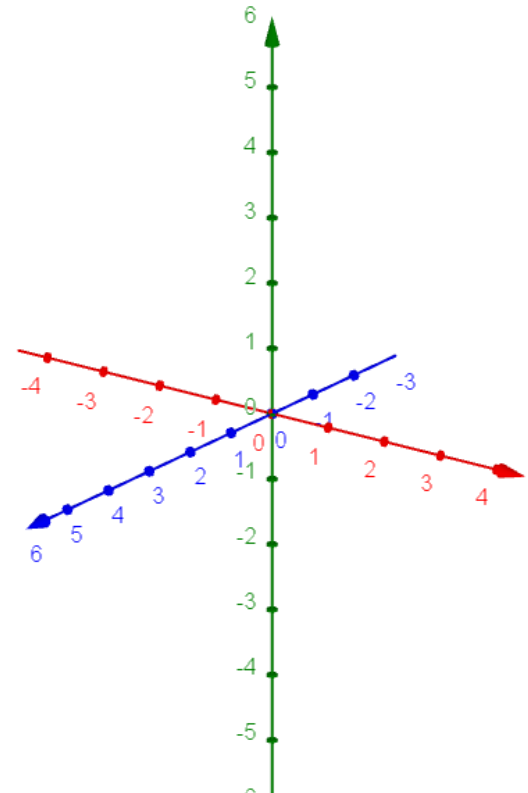
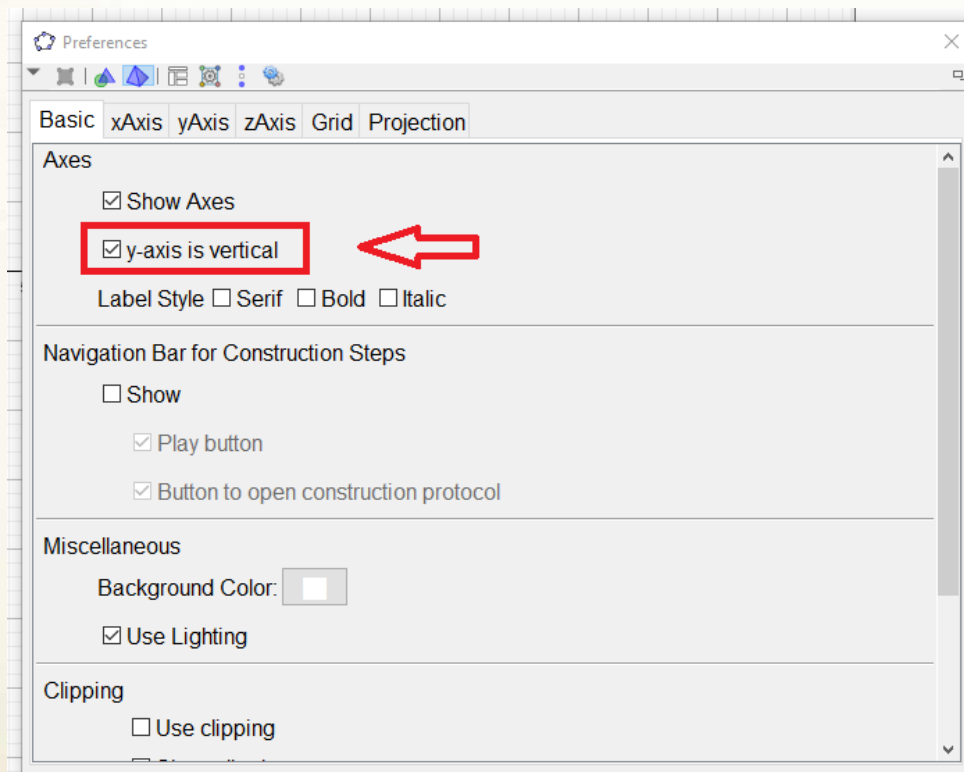
Naudosime šį Dekarto koordinačių ašių išdėstymą



Pagal susitarimą **X ašis** visada žymima **raudona spalva**,
Y ašis – **žalia** ir **Z ašis** – **mėlyna spalva**.

Ašių sukeitimas Geogebra programoje

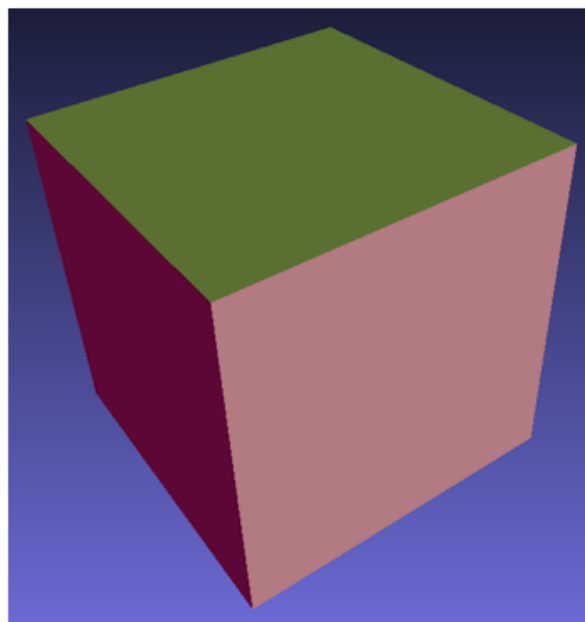
Options → Advanced → Preferences - 3D Graphics → Basic



Skaitmeninių modelių formatai

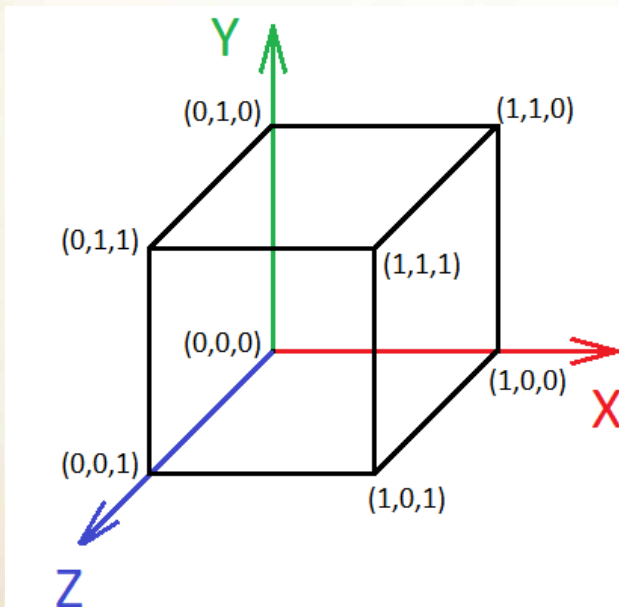
*.ply, *.stl, *.obj, *.qobj, *.off, *.ptx, *.vmi, *.bre, *.dae,
*.ctm, *.pts, *.apts, *.xyz, *.pcl, *.gts, *.pdb, *.tri, *.asc,
*.x3d, *.x3dv, *.wrl, *.bw.

```
cube.off - Notepad
File Edit Format View Help
OFF
8 6 0] - (8 viršūnės, 6 sienos)
0 0 0
0 0 1
0 1 0
0 1 1
1 0 0
1 0 1
1 1 0
1 1 1
4 0 1 3 2 0 0 205
4 2 3 7 6 255 127 36
4 4 6 7 5 110 139 61
4 0 4 5 1 139 10 80
4 1 5 7 3 255 185 15
4 0 2 6 4 205 140 149
keturkampių viršūnės
(taškų numeriai pradedant nuo 0)
daugiakampių kampų skaičius
(keturkampiai, nes 4)
```

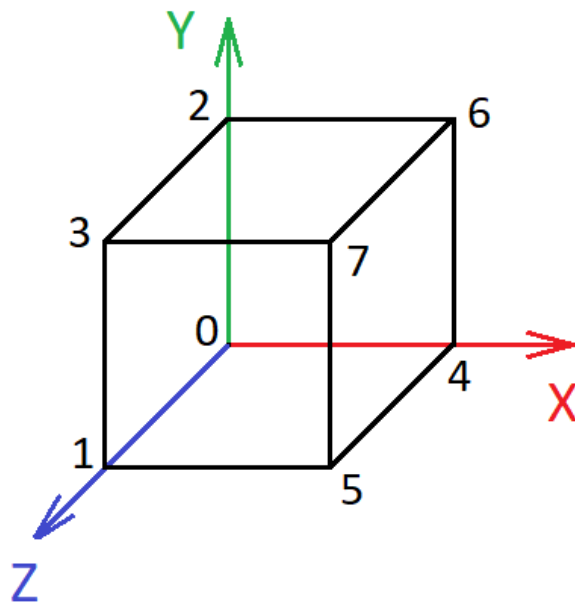


Kubo sudarymas off formatu

1 etapas: apskaičiuojamos kubo viršūnių koordinatės.



2 etapas: viršūnės indeksuojamos pradedant 0.

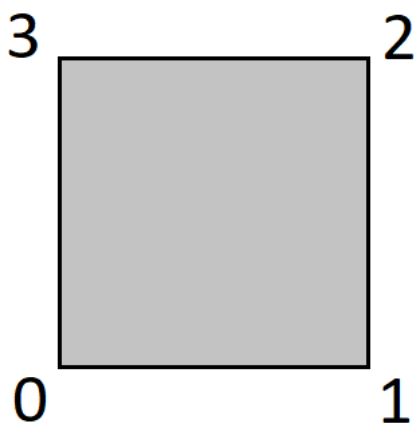


3 etapas: sukuriamas OFF failas, į kurį įrašomos viršūnių koordinatės ir viršūnių indeksų sekos.

```
kubas.off... - □ ×
File Edit Format View Help
OFF
8 6 0
0 0 0
0 0 1
0 1 0
0 1 1
1 0 0
1 0 1
1 1 0
1 1 1
4 0 4 5 1
4 0 1 3 2
4 0 2 6 4
4 1 5 7 3
4 2 3 7 6
4 4 6 7 5
```

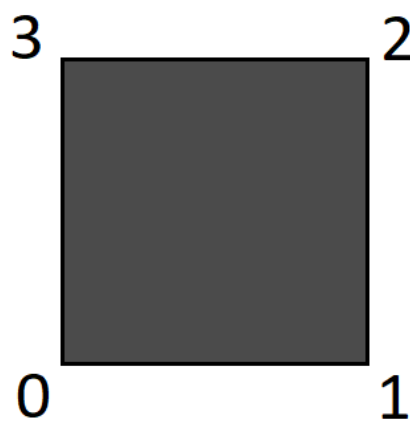
Vidinė ir išorinė siena

Išorinė keturkampio siena



4 0 1 2 3

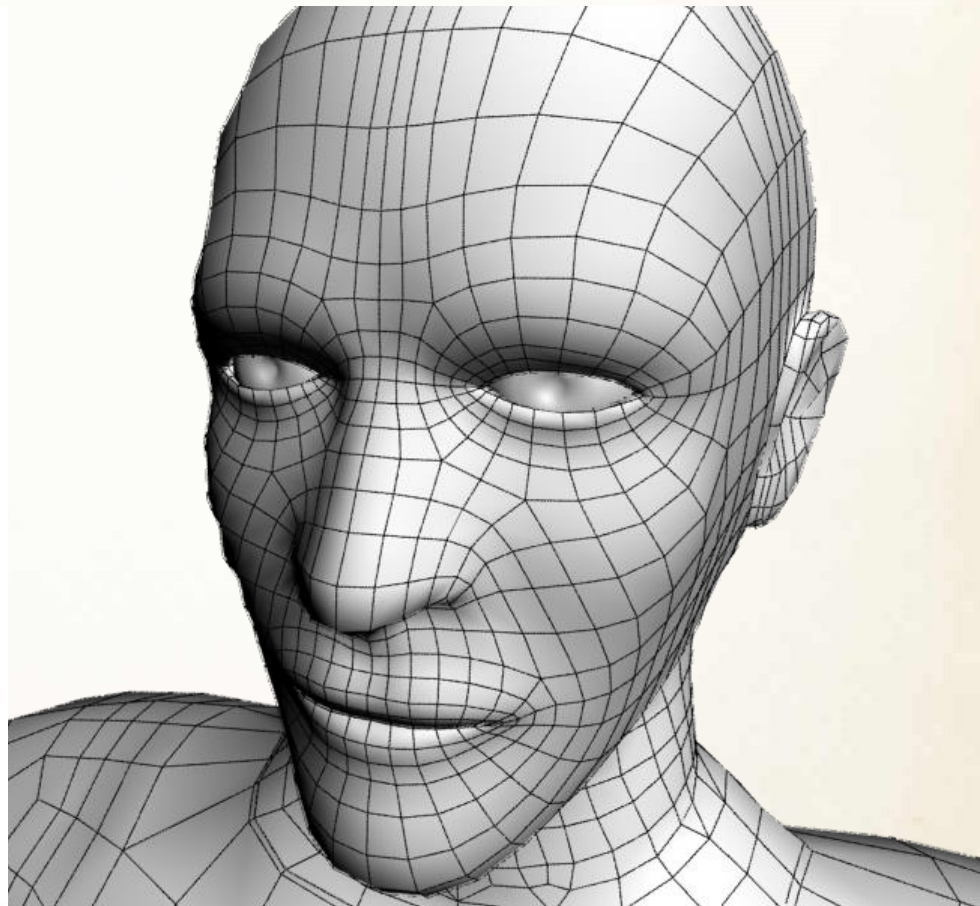
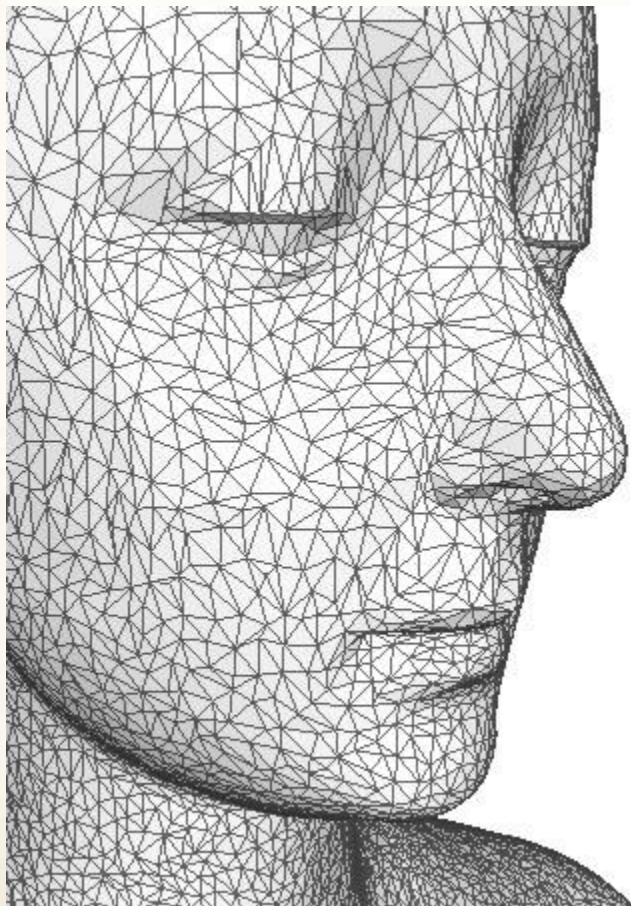
Vidinė keturkampio siena



4 3 2 1 0

Žvelgiant į 3D modelį iš išorės, viršūnių indeksus reikia išdėstyti prieš laikrodžio rodyklę.
Žvelgiant į 3D modelį iš vidaus, viršūnių indeksus reikia išdėstyti pagal laikrodžio rodyklę.

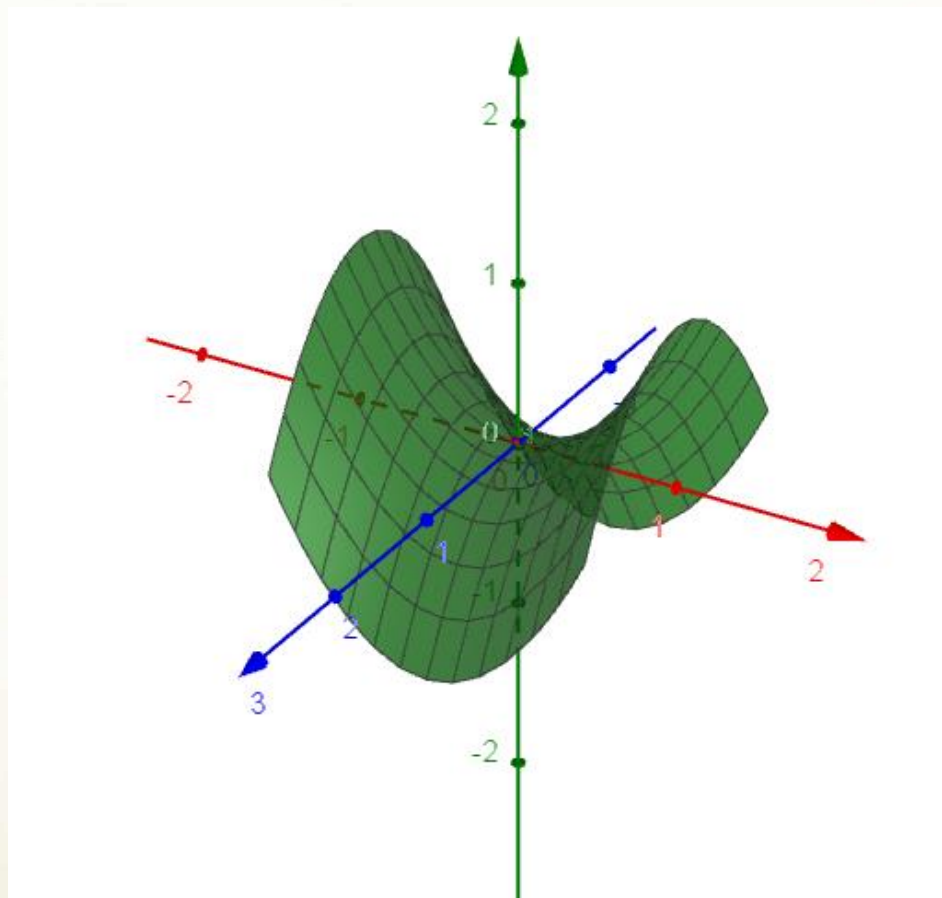
Trianguliarusis ir kvadrianguliarusis paviršius



Rekomenduojami 3D modelio kūrimo etapai

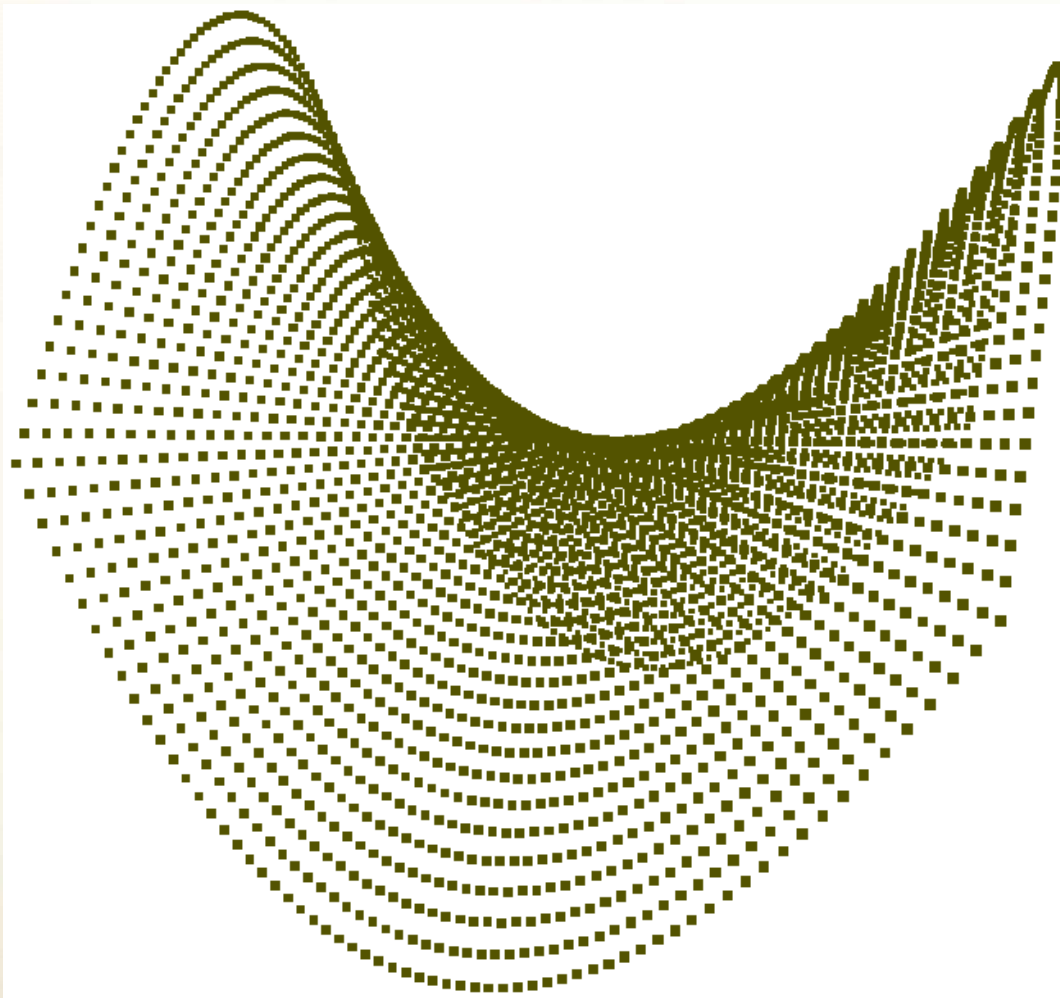
1. 3D eskizo sudarymas;
2. Taškų debesies generavimas (*.xyz formatas);
3. Taškų susiejimas sienomis (*.off formatas);
4. Sienų nuspalvinimas (*.off formatas).

3D eskizo sudarymas (Geogebra)



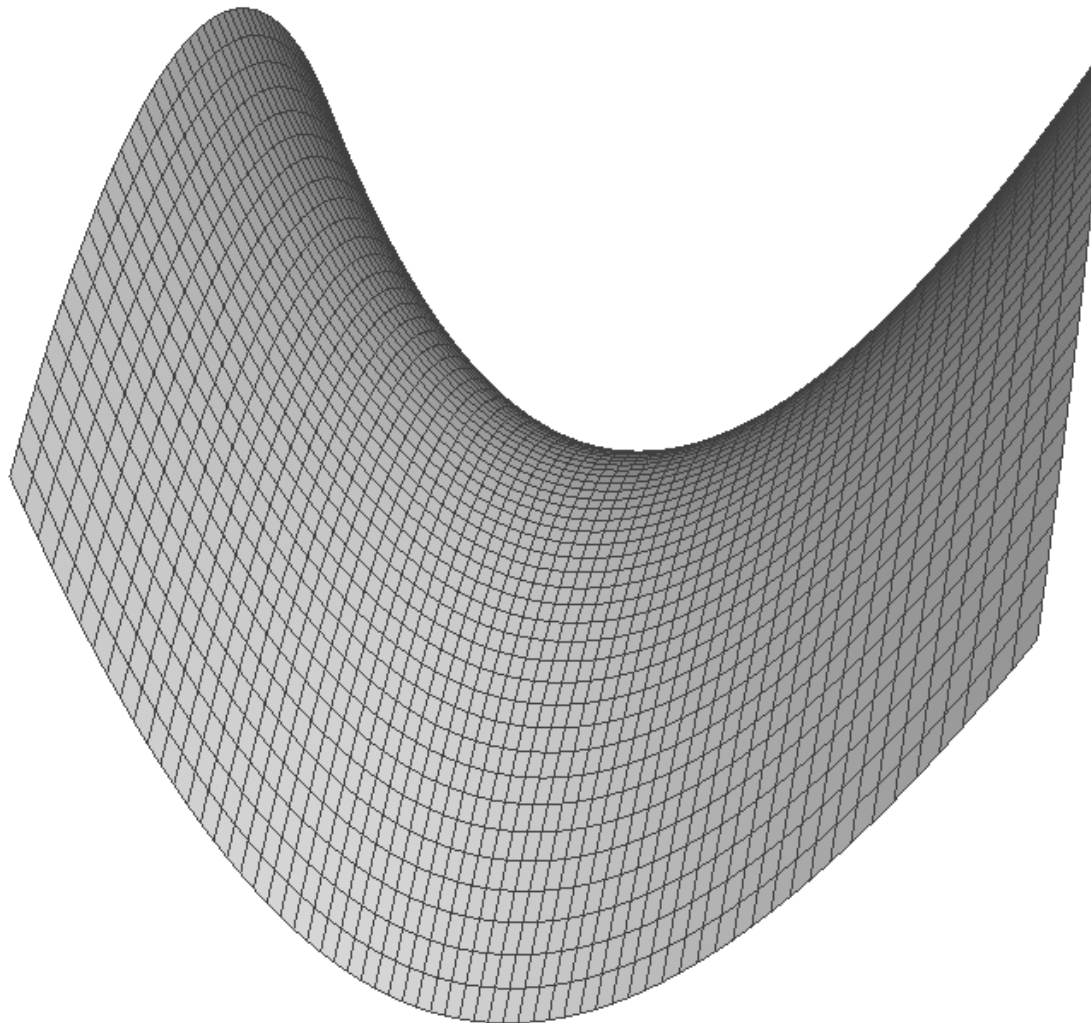
Surface($u, u^2 - v^2, v, u, -1, 1, v, -1, 1$)

Taškų debesies generavimas (* .xyz formatas)



```
balnas123.xyz - Notepad
File Edit Format View Help
-1.000000 0.000000 -1.000000
-1.000000 0.061523 -0.968750
-1.000000 0.121094 -0.937500
-1.000000 0.178711 -0.906250
-1.000000 0.234375 -0.875000
-1.000000 0.288086 -0.843750
-1.000000 0.339844 -0.812500
-1.000000 0.389648 -0.781250
-1.000000 0.437500 -0.750000
-1.000000 0.483398 -0.718750
-1.000000 0.527344 -0.687500
-1.000000 0.569336 -0.656250
-1.000000 0.609375 -0.625000
-1.000000 0.647461 -0.593750
-1.000000 0.683594 -0.562500
-1.000000 0.717773 -0.531250
-1.000000 0.750000 -0.500000
-1.000000 0.780273 -0.468750
-1.000000 0.808594 -0.437500
-1.000000 0.834961 -0.406250
-1.000000 0.859375 -0.375000
-1.000000 0.881836 -0.343750
-1.000000 0.902344 -0.312500
```

Taškų susiejimas sienomis (* .off formatas)



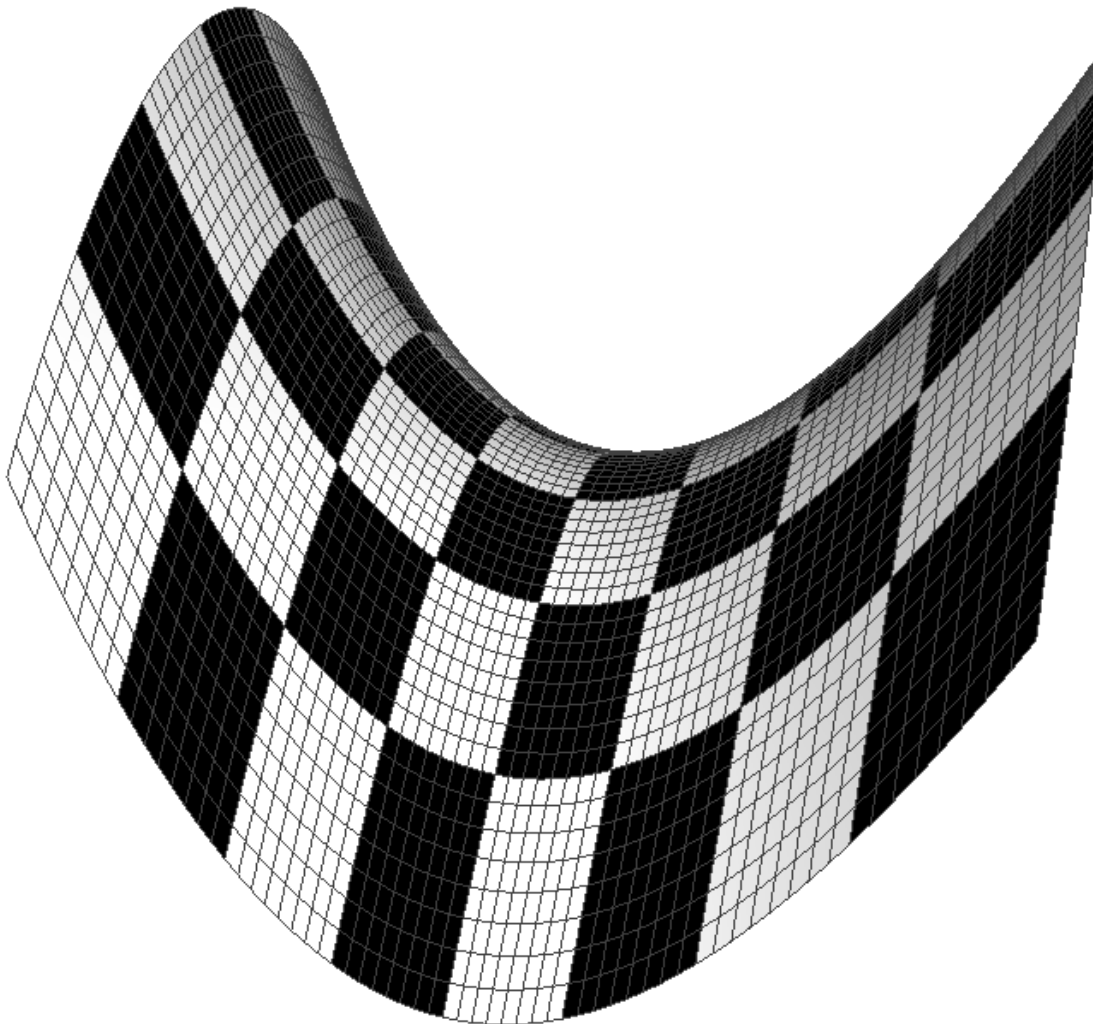
```
balnas123 - Notepad
File Edit Format View Help
OFF
4225 4096 12288
-1.000000 0.000000 -1.000000
-1.000000 0.061523 -0.968750
-1.000000 0.121094 -0.937500
-1.000000 0.178711 -0.906250
-1.000000 0.234375 -0.875000

<...>

1.000000 0.234375 0.875000
1.000000 0.178711 0.906250
1.000000 0.121094 0.937500
1.000000 0.061523 0.968750
1.000000 0.000000 1.000000
4 0 1 66 65
4 1 2 67 66
4 2 3 68 67
4 3 4 69 68
4 4 5 70 69
4 5 6 71 70
4 6 7 72 71
4 7 8 73 72
4 8 9 74 73

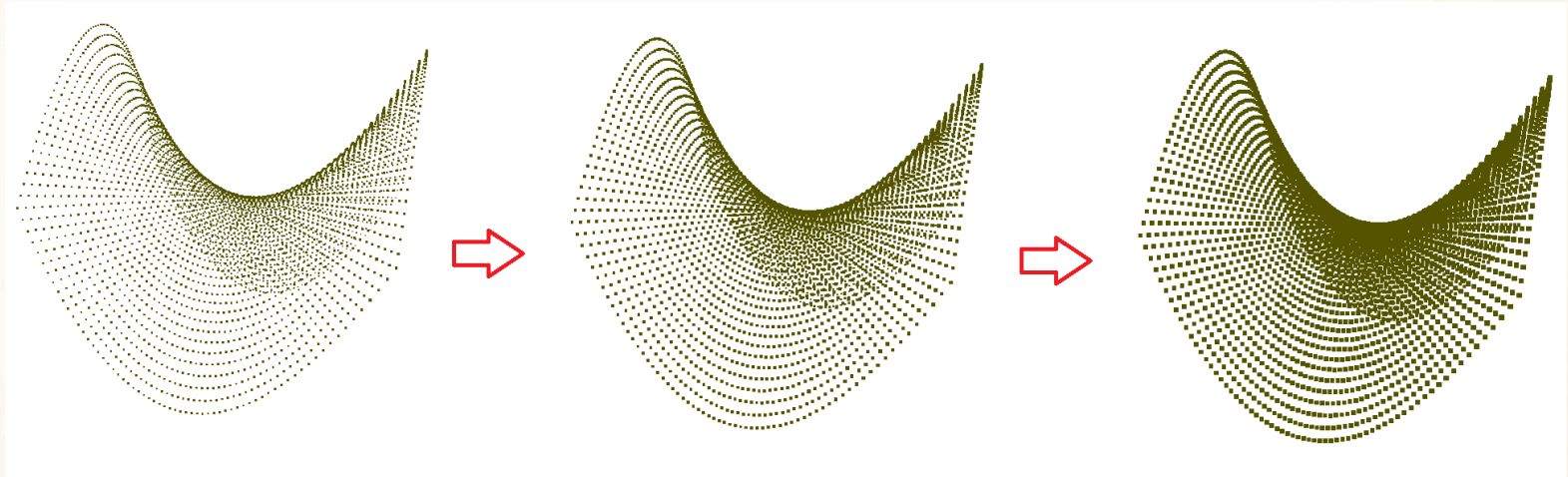
<...>
```

Sienų nuspalvinimas (* .off formatas)



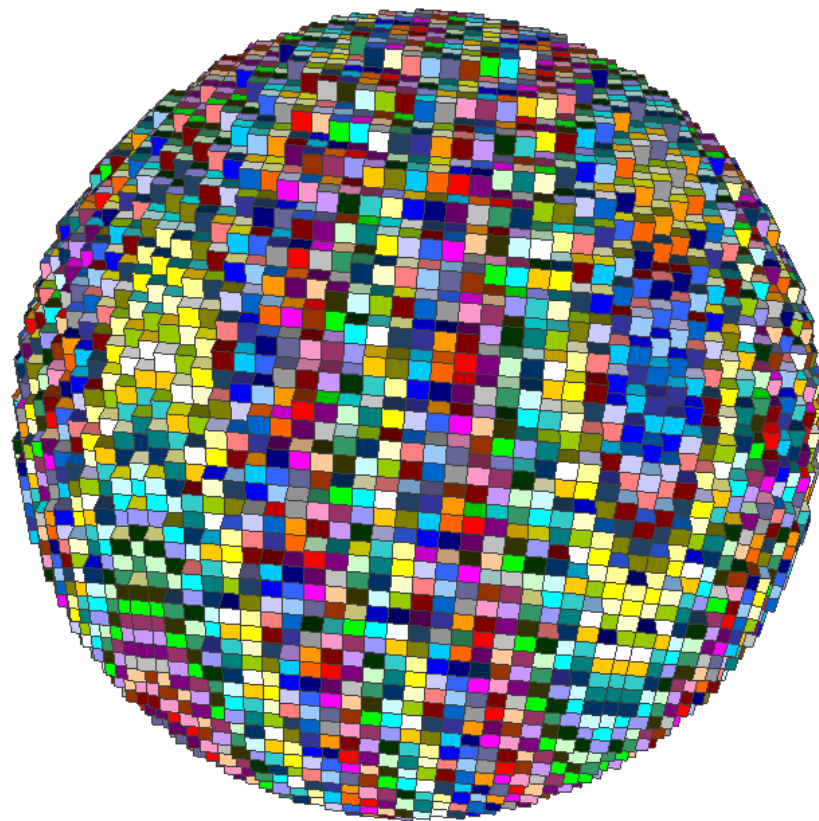
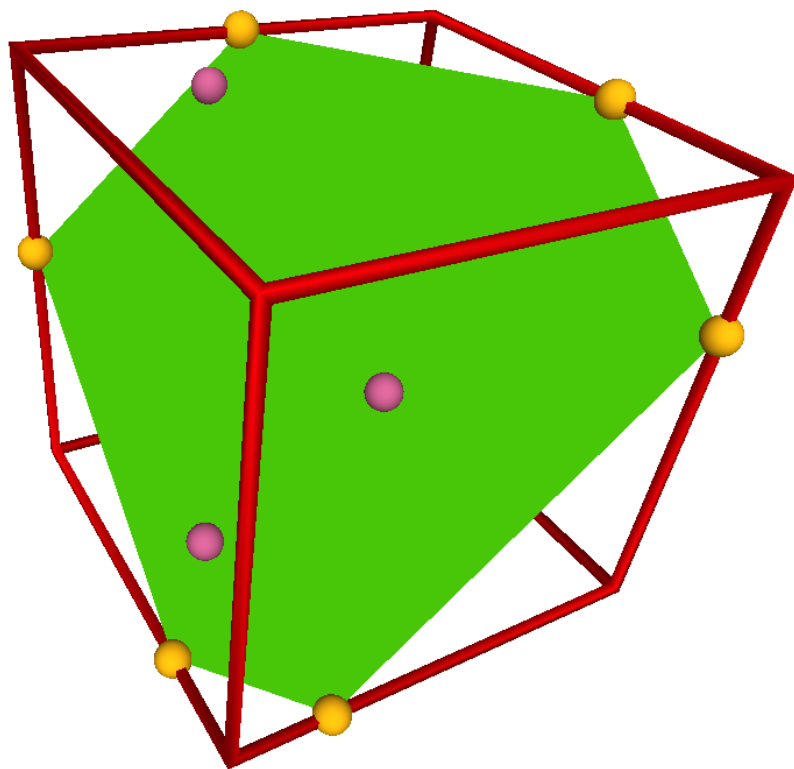
```
balnas123 - Notepad
File Edit Format View Help
OFF
4225 4096 12288
-1.000000 0.000000 -1.000000
-1.000000 0.061523 -0.968750
-1.000000 0.121094 -0.937500
-1.000000 0.178711 -0.906250
-1.000000 0.234375 -0.875000
<...>
1.000000 0.234375 0.875000
1.000000 0.178711 0.906250
1.000000 0.121094 0.937500
1.000000 0.061523 0.968750
1.000000 0.000000 1.000000
4 0 1 66 65 0 0 0
4 1 2 67 66 0 0 0
4 2 3 68 67 0 0 0
4 3 4 69 68 0 0 0
4 4 5 70 69 0 0 0
4 5 6 71 70 0 0 0
4 6 7 72 71 0 0 0
4 7 8 73 72 0 0 0
4 8 9 74 73 255 255 255
<...>
```


Taškų paryškimas MeshLab programoje

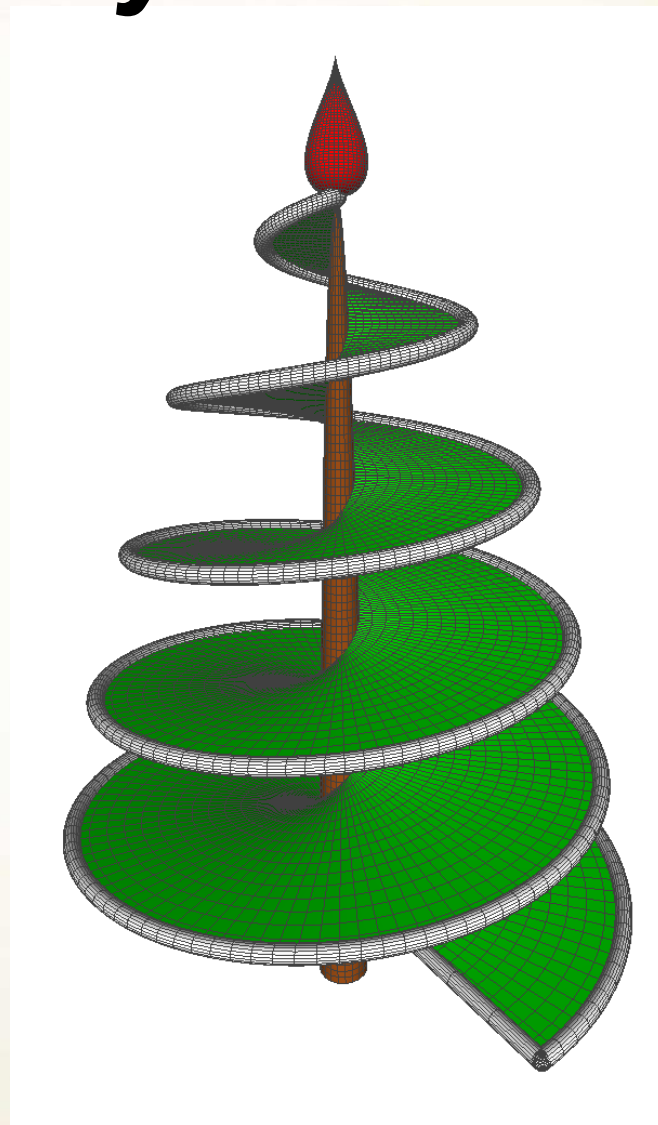
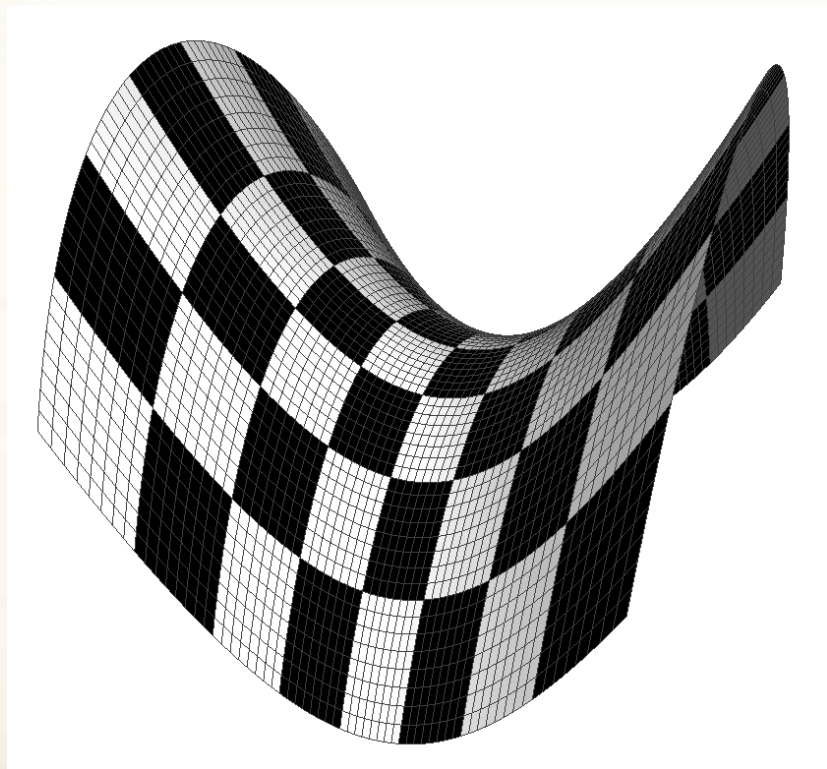


Alt + Mouse Wheel Up,
Alt + Mouse Wheel Down.

3D modelių pavyzdžiai



3D modelių pavyzdžiai



Parametrinės kreivės sąvoka

- 2D parametrinės kreivės užrašymas:

$$x = x(t),$$

$$y = y(t).$$

- 3D parametrinės kreivės užrašymas:

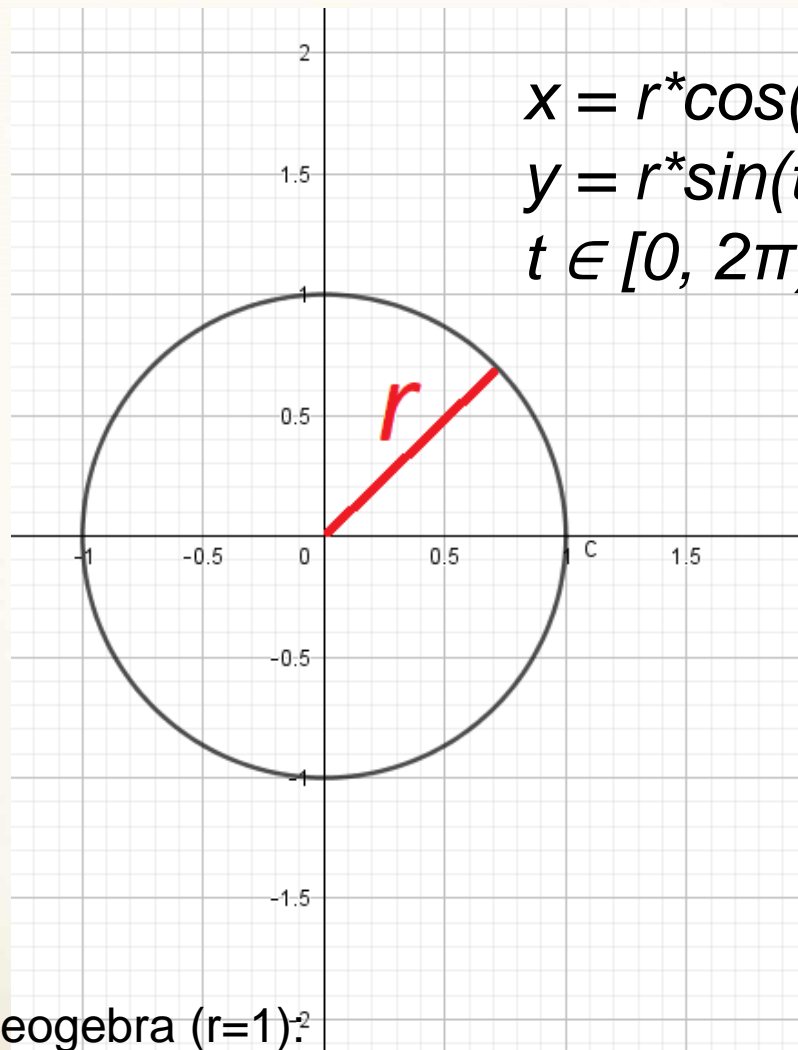
$$x = x(t),$$

$$y = y(t),$$

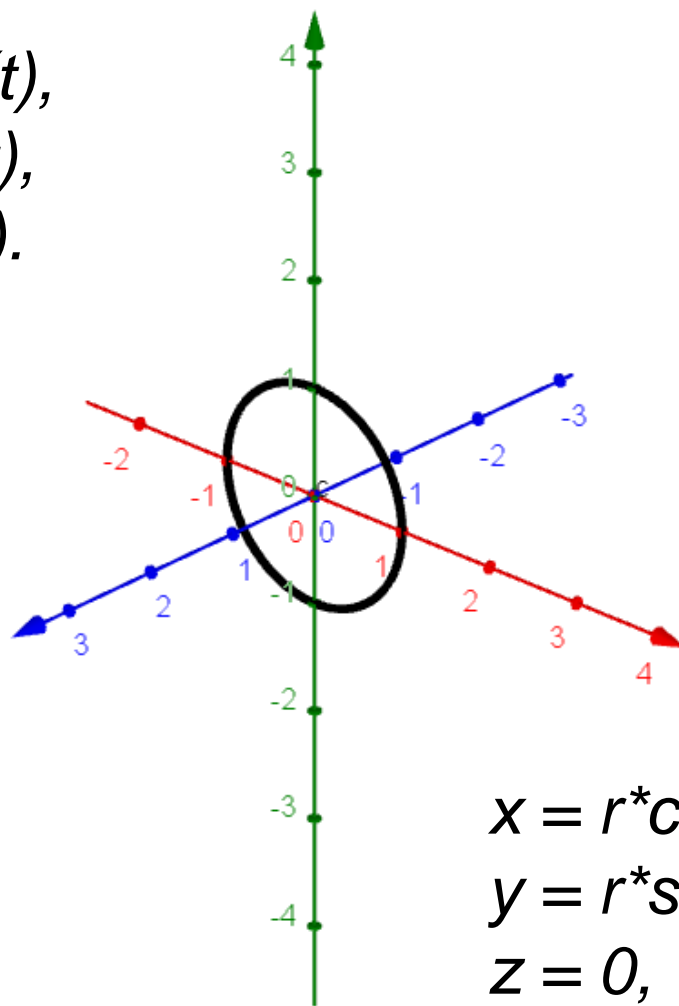
$$z = z(t).$$

t – parametras, priklausantis intervalui $[t_{min}, t_{max}]$.

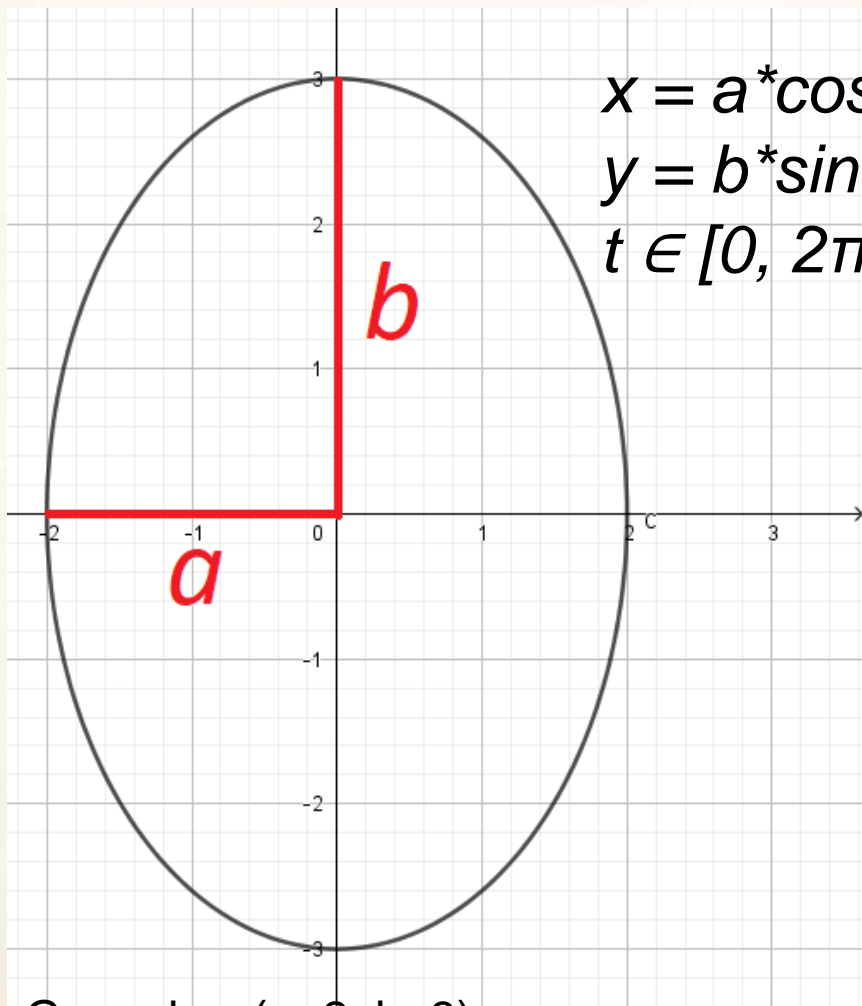
Apskritimo lygtis XY plokštumoje



Geogebra (r=1)²
Curve(cos(t), sin(t), t, 0, 2π)

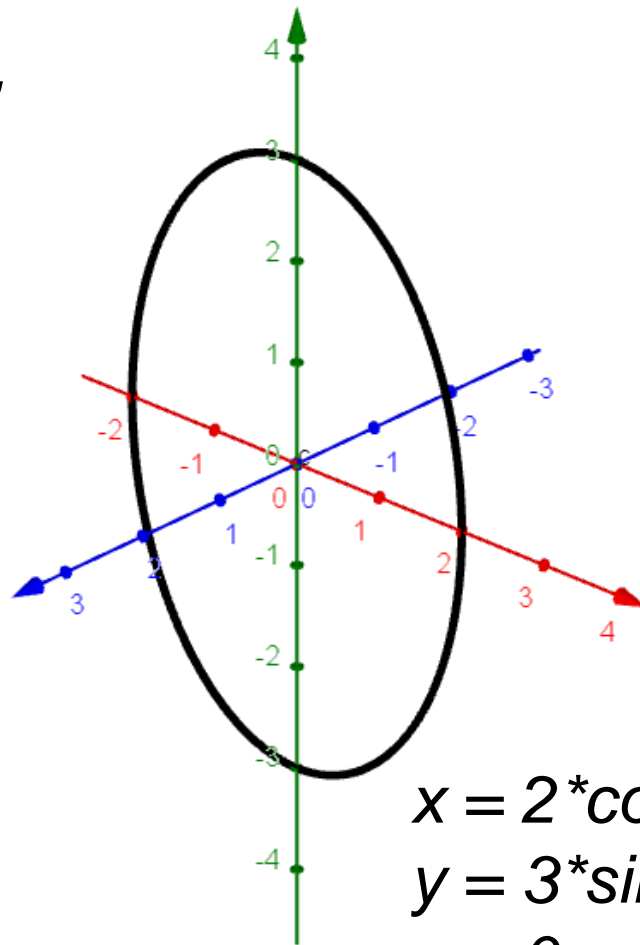


Elipses lygtis XY plokštumoje



$$\begin{aligned}x &= a \cdot \cos(t), \\y &= b \cdot \sin(t), \\t &\in [0, 2\pi).\end{aligned}$$

Geogebra (a=2, b=3):
Curve(2 cos(t), 3 sin(t), t, 0, 2π)

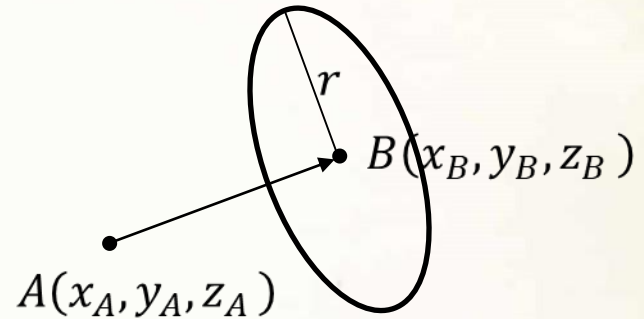


$$\begin{aligned}x &= 2 \cdot \cos(t), \\y &= 3 \cdot \sin(t), \\z &= 0, \\t &\in [0, 2\pi).\end{aligned}$$

Apskritimo lygtis erdvėje

<http://demonstrations.wolfram.com/ParametricEquationOfACircleIn3D/>

Apskritimo (kurio spindulys r ir centrinis taškas B), ortogonalus vektoriumi AB , parametrinė lygtis:



$$x(t) = x_B - \frac{r \cos(t)(y_B - y_A)\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2} + r \sin(t)(x_B - x_A)(z_B - z_A)}{\sqrt{((x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2)((x_B - x_A)^2 + (y_B - y_A)^2)}}$$

$$y(t) = y_B + \frac{r \cos(t)(x_B - x_A)\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2} - r \sin(t)(z_B - z_A)(y_B - y_A)}{\sqrt{((x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2)((x_B - x_A)^2 + (y_B - y_A)^2)}}$$

$$z(t) = z_B + \frac{r \sin(t)\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}}{\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2}}, \quad t \in [0, 2\pi)$$

$$\operatorname{sgn}(x) = \begin{cases} -1, & x < 0 \\ 0, & x = 0 \\ 1, & x > 0 \end{cases}$$

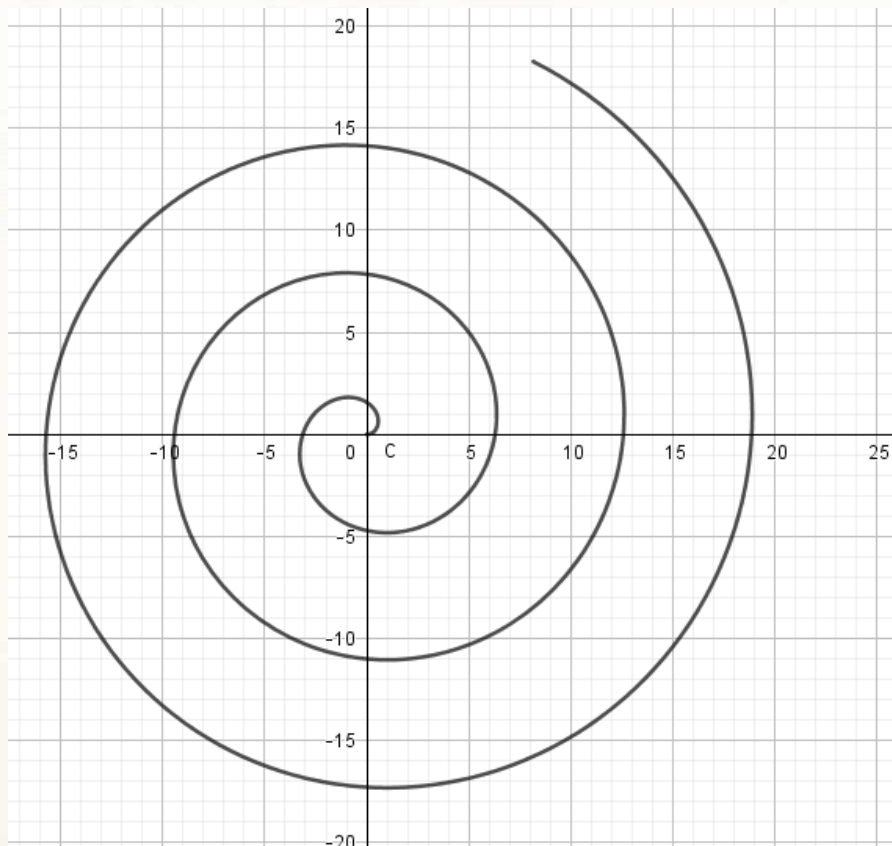
$$x(t) = x_B - \frac{r\sqrt{2}}{2} (\operatorname{sgn}(z_B - z_A)\sin(t) + \cos(t))$$

Jei $x_A = x_B$ ir $y_A = y_B$, tai:

$$y(t) = y_B - \frac{r\sqrt{2}}{2} (\operatorname{sgn}(z_B - z_A)\sin(t) - \cos(t))$$

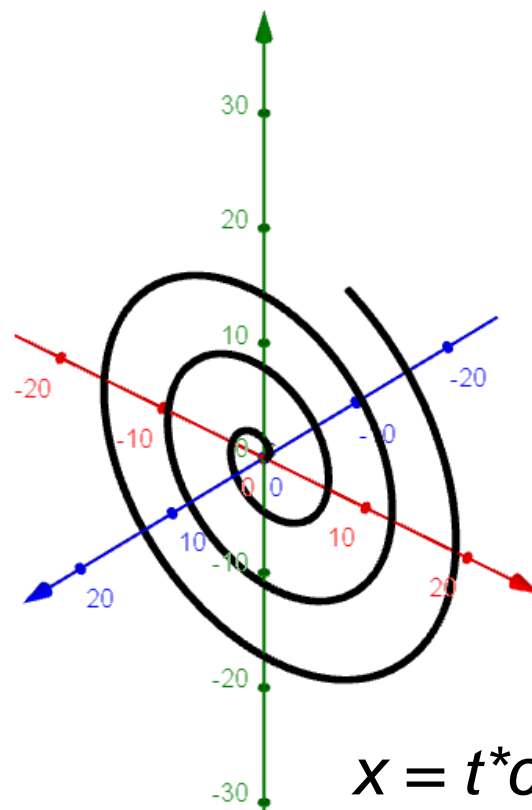
$$z(t) = z_B, \quad t \in [0, 2\pi)$$

Kitų parametrinių kreivių pavyzdžiai



Geogebra:

Curve(t cos(t), t sin(t), t, 0, 20)

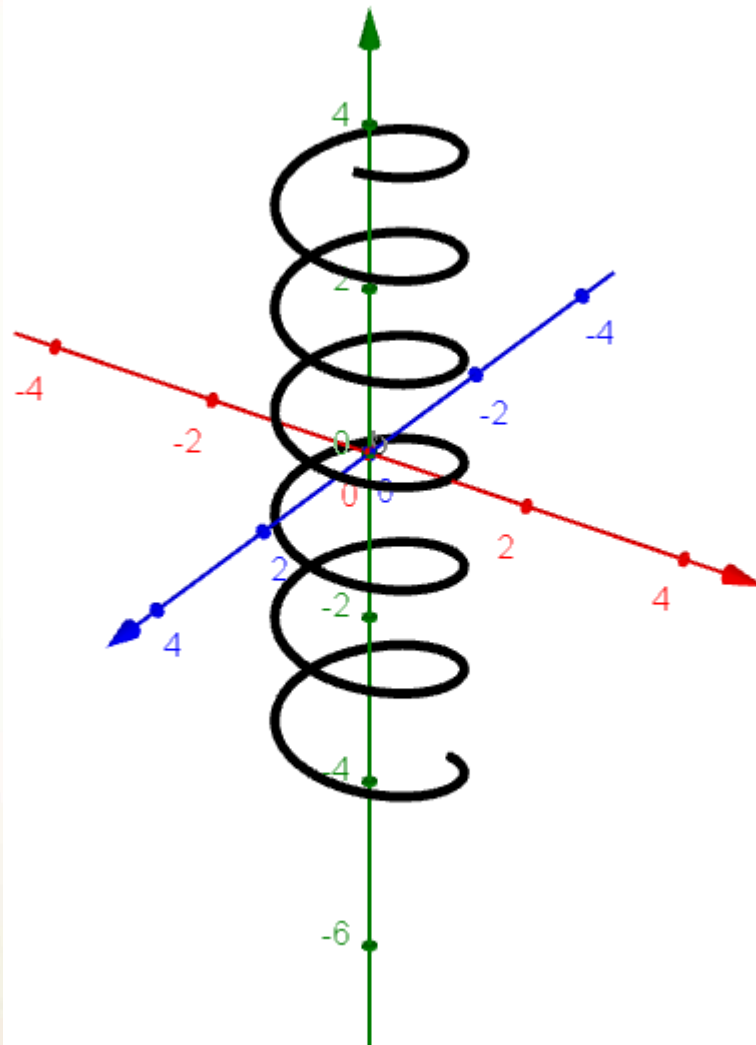


$$\begin{aligned}x &= t \cdot \cos(t), \\y &= t \cdot \sin(t), \\z &= 0, \\t &\in [0, 20].\end{aligned}$$

Kitų parametrinių kreivių pavyzdžiai

$$\begin{aligned}x &= \cos(t), \\y &= t / 5, \\z &= \sin(t), \\t &\in [-20, 20].\end{aligned}$$

Geogebra:
Curve(cos(t), t / 5, sin(t), t, -20, 20)

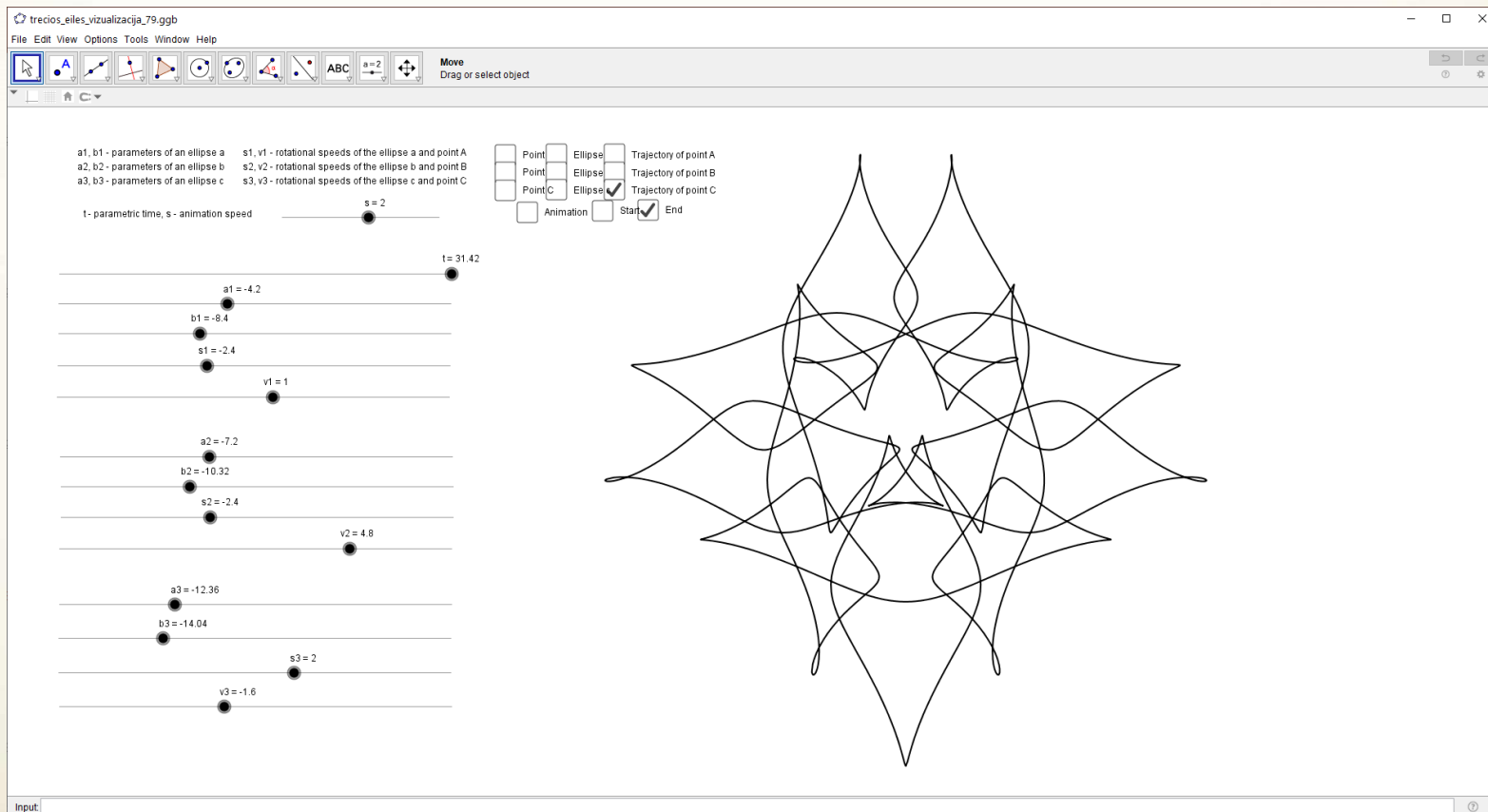


Elipsinés parametrinés 2D kreivés

$$x(t) = \sin(s_1 t) b_1 \cos(v_1 t) + \cos(s_1 t) a_1 \sin(v_1 t) + \sin(s_2 t) b_2 \cos(v_2 t) + \cos(s_2 t) a_2 \sin(v_2 t) + \sin(s_3 t) b_3 \cos(v_3 t) + \cos(s_3 t) a_3 \sin(v_3 t)$$

$$y(t) = -a_1 \sin(v_1 t) \sin(s_1 t) + b_1 \cos(v_1 t) \cos(s_1 t) - a_2 \sin(v_2 t) \sin(s_2 t) + b_2 \cos(v_2 t) \cos(s_2 t) - a_3 \sin(v_3 t) \sin(s_3 t) + b_3 \cos(v_3 t) \cos(s_3 t)$$

$$t \in [0, 10\pi]$$



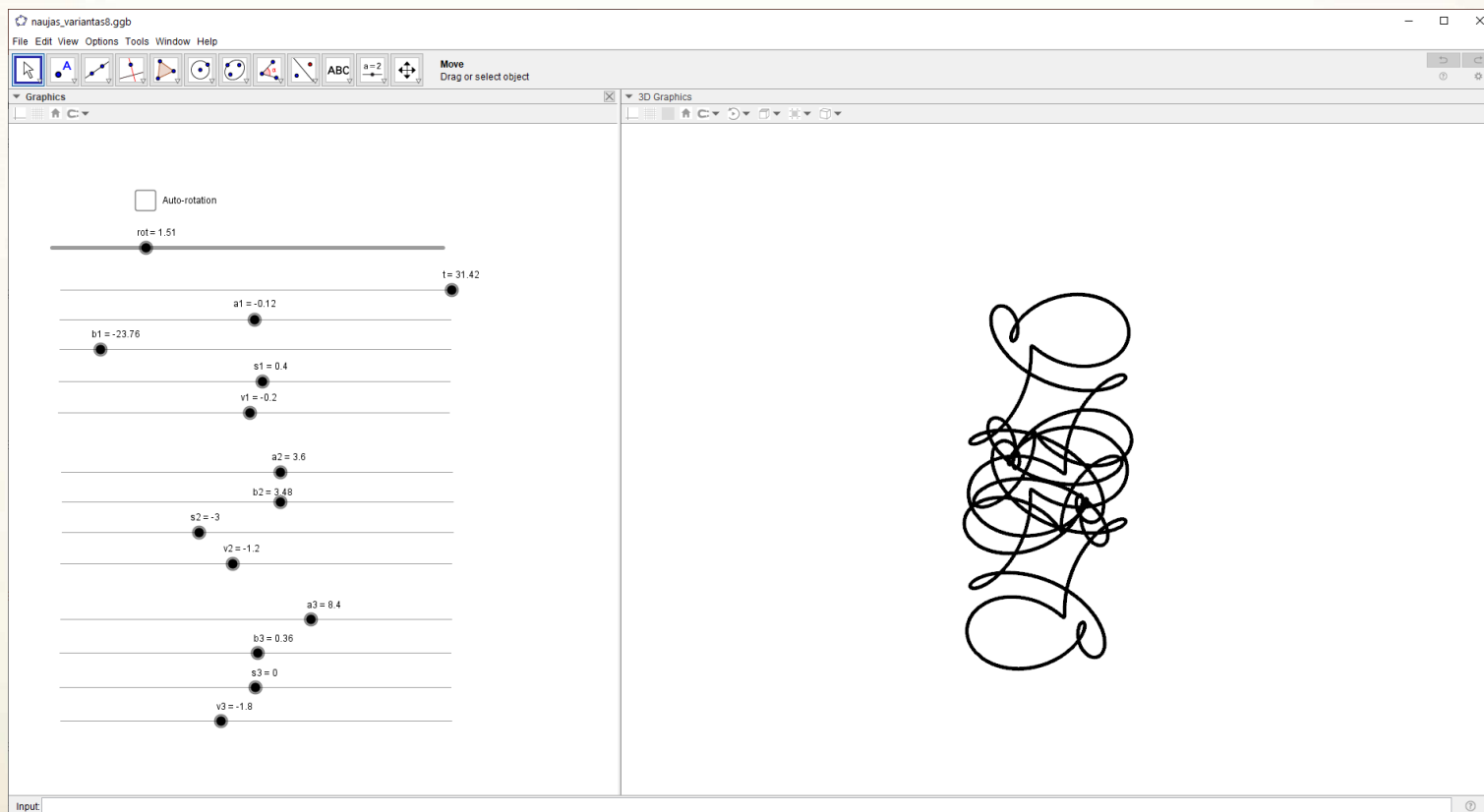
Elipsinés parametrinés 3D kreivés

$$x(t) = \sin(s_3 t) b_3 \cos(v_3 t) + \cos(s_3 t) a_3 \sin(v_3 t) - a_2 \sin(v_2 t) \sin(s_2 t) + b_2 \cos(v_2 t) \cos(s_2 t)$$

$$y(t) = \sin(s_1 t) b_1 \cos(v_1 t) + \cos(s_1 t) a_1 \sin(v_1 t) - a_3 \sin(v_3 t) \sin(s_3 t) + b_3 \cos(v_3 t) \cos(s_3 t)$$

$$z(t) = \sin(s_2 t) b_2 \cos(v_2 t) + \cos(s_2 t) a_2 \sin(v_2 t) - a_1 \sin(v_1 t) \sin(s_1 t) + b_1 \cos(v_1 t) \cos(s_1 t)$$

$$t \in [0, 10\pi]$$

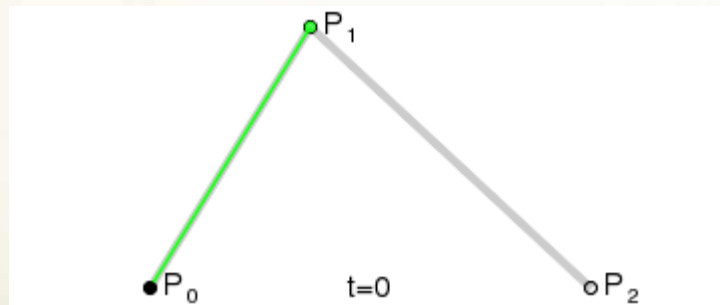


Bezjė kreivės

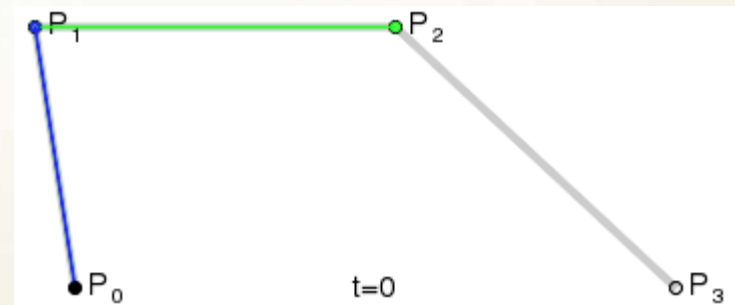
$$\mathbf{B}(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i \mathbf{P}_i$$

\mathbf{P}_i – kontroliniai taškai, n – kreivės eilė.

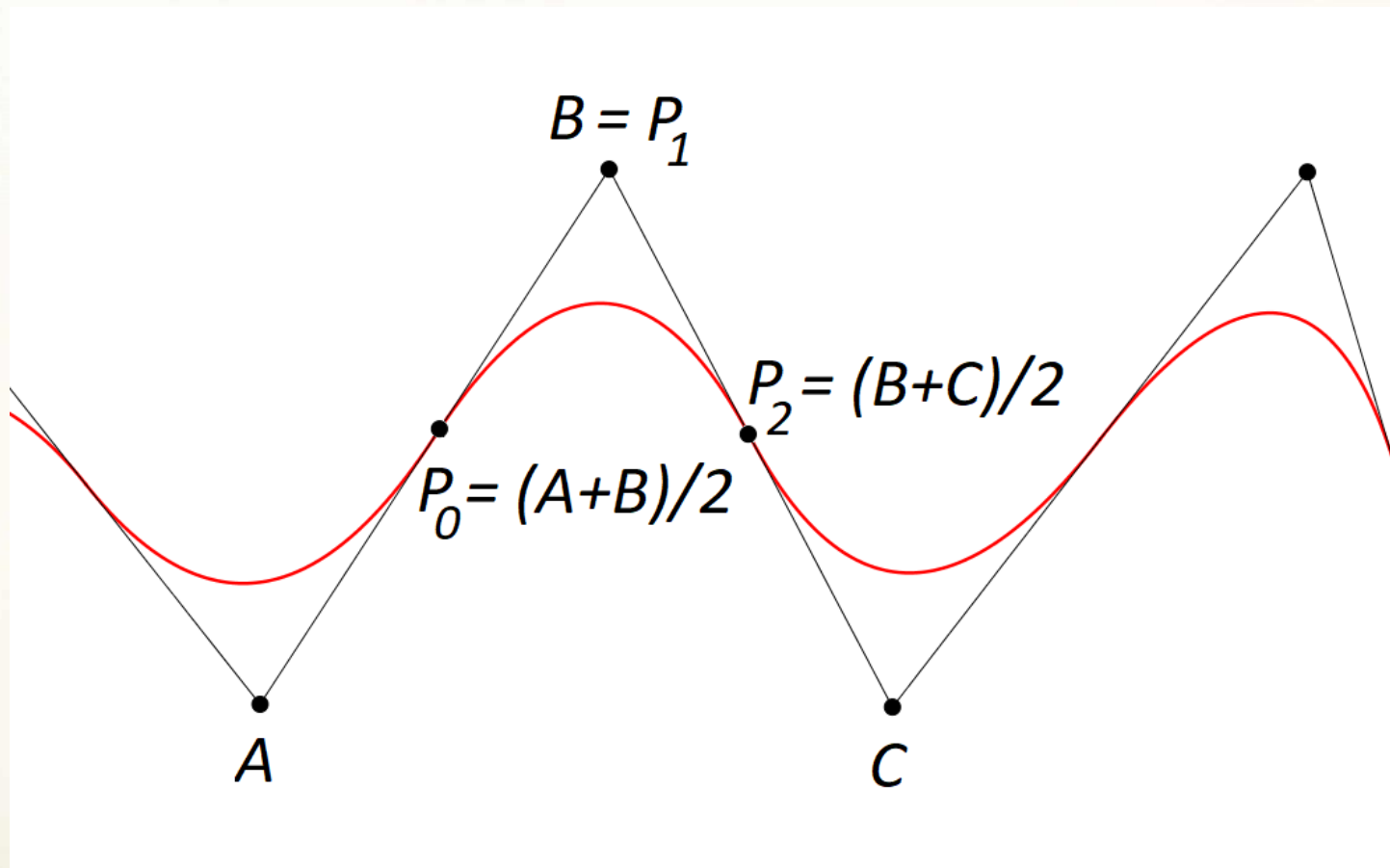
$n = 2$:



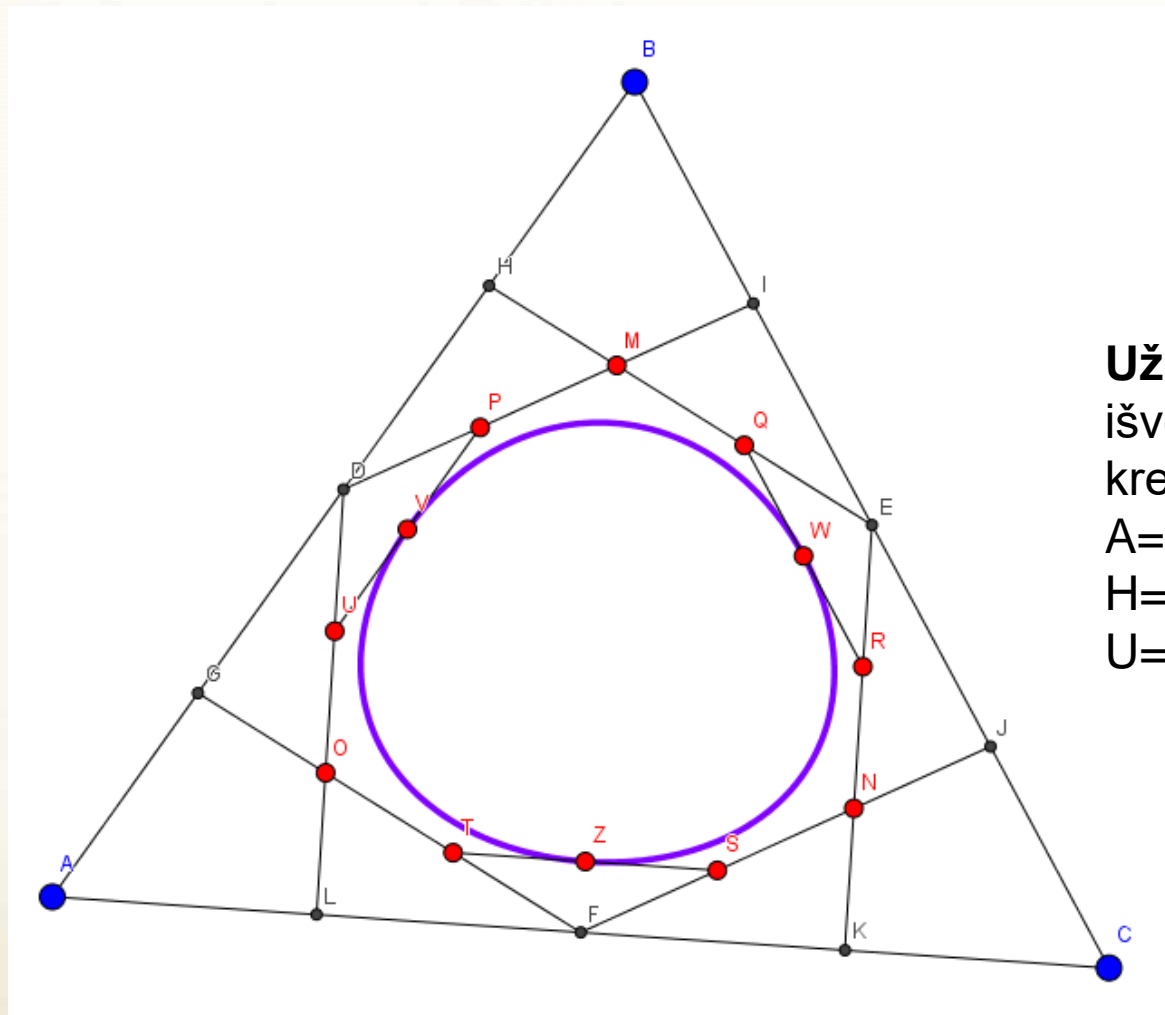
$n = 3$:



2 eilēs Bezzē kreiviu kontūras



4 eilēs Bežjē kreivju kontūras



Užduotis: pagal brēžinj
išvesti 4 eilēs Bežjē
kreivju lygtis, kai
 $A=G$, $G=D$, $D=H$,
 $H=B$, $D=P$, $P=M$,
 $U=V$, $V=P$ ir t. t.

Parametriniai paviršiai

- Parametrinio paviršiaus užrašymas:

$$x = x(u, v),$$

$$y = y(u, v),$$

$$z = z(u, v).$$

u – parametras, priklausantis intervalui $[u_{min}, u_{max}]$,

v – parametras, priklausantis intervalui $[v_{min}, v_{max}]$.

Sfera

Parametrinis sferos
paviršius:

$$x=r*\cos(u)*\sin(v),$$

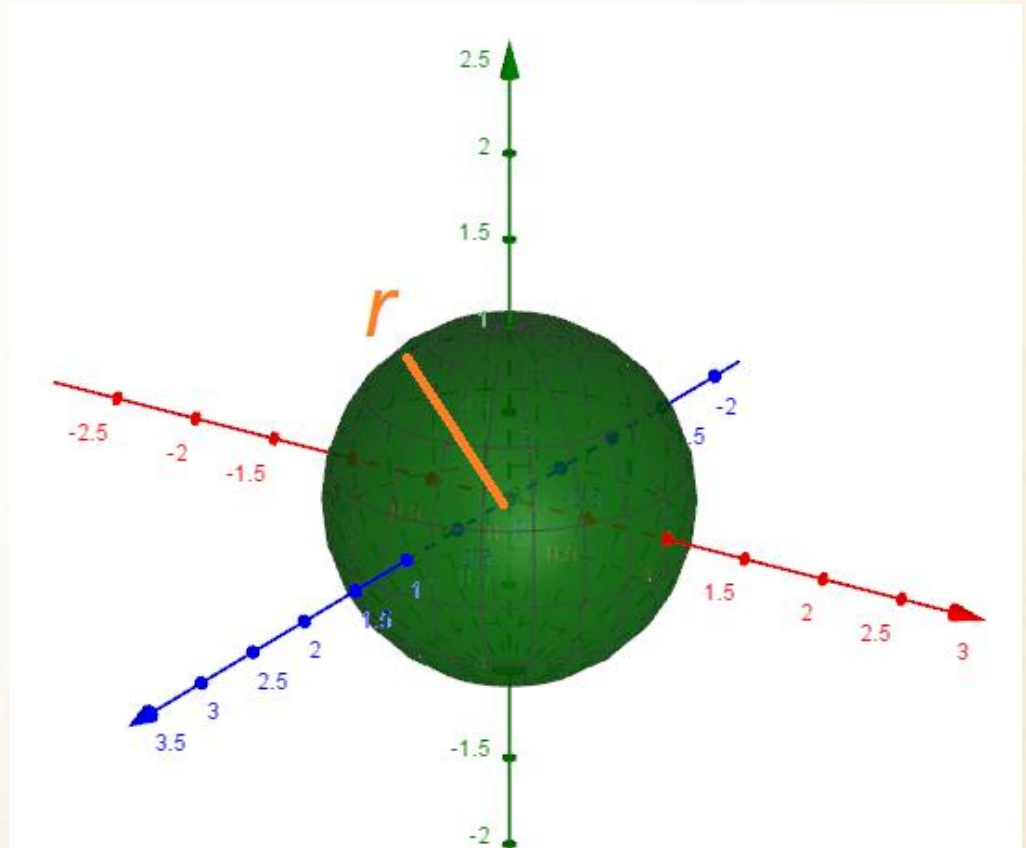
$$y=r*\cos(v),$$

$$z=r*\sin(u)*\sin(v),$$

$$u\in[0, 2\pi), v\in[0, \pi).$$

Geogebra (r=1):

Surface(cos(u) sin(v), cos(v),
sin(u) sin(v), u, 0, 2π, v, 0, π)



Cilindras

*Parametrinis
cilindro paviršius:*

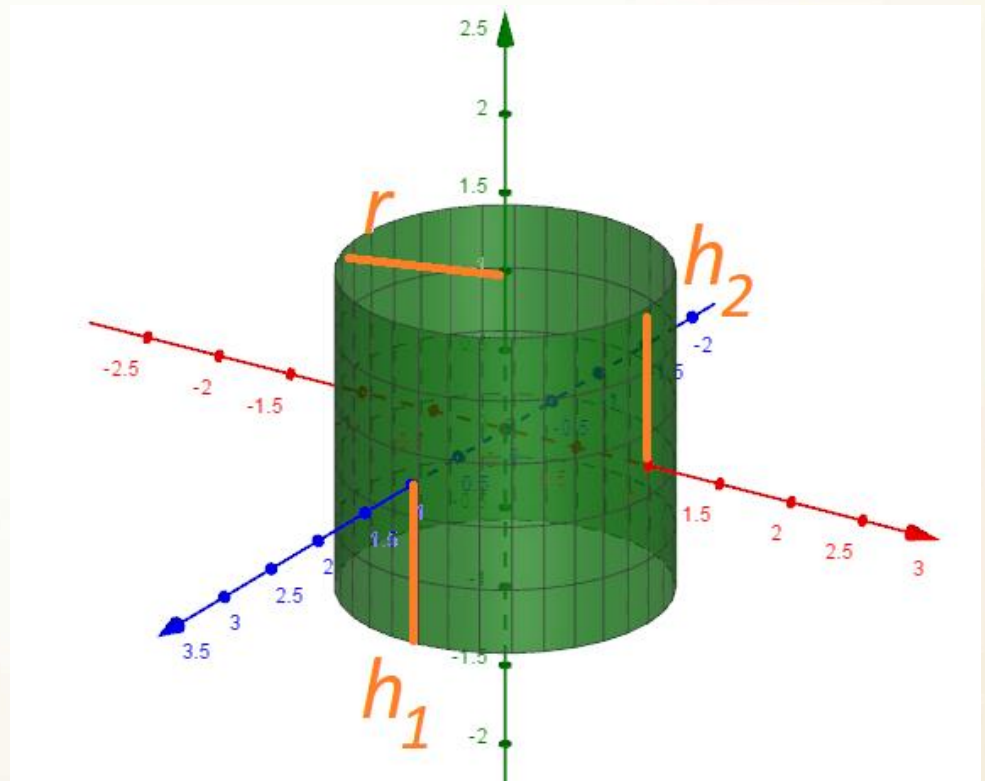
$$x=r*\cos(u),$$

$$y=v,$$

$$z=r*\sin(u),$$

$$u\in[0,2\pi), v\in[h_1, h_2].$$

Geogebra ($r=1$, $h_1=-1$, $h_2=1$):
Surface(cos(u), v, sin(u), u, 0,
 2π , v, -1, 1)



Toras

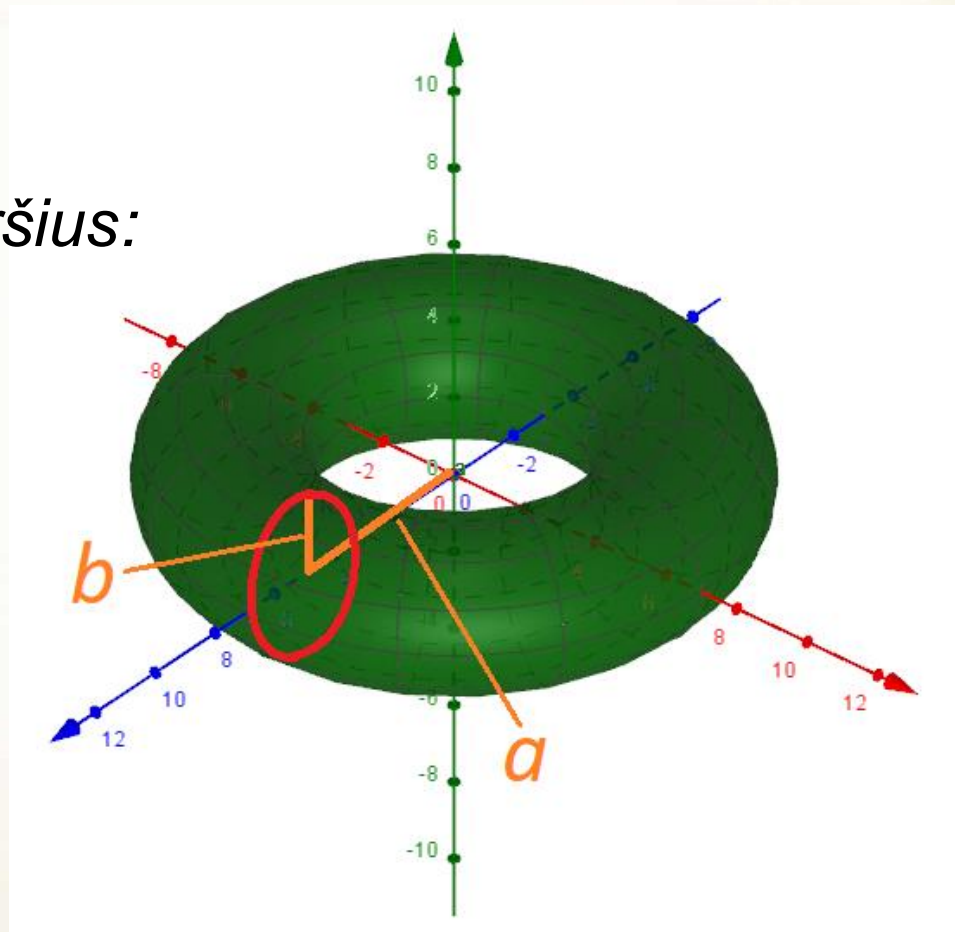
Parametrinis toro paviršius:

$$x=(a+b*\cos(u))*\cos(v),$$

$$y=b*\sin(u),$$

$$z=(a+b*\cos(u))*\sin(v),$$

$$u\in[0, 2\pi), v\in[0, 2\pi).$$



Geogebra (a=5, b=2):

Surface((5 + 2cos(u)) cos(v),

2sin(u), (5 + 2cos(u)) sin(v),

u, 0, 2π, v, 0, 2π)

Sukinys

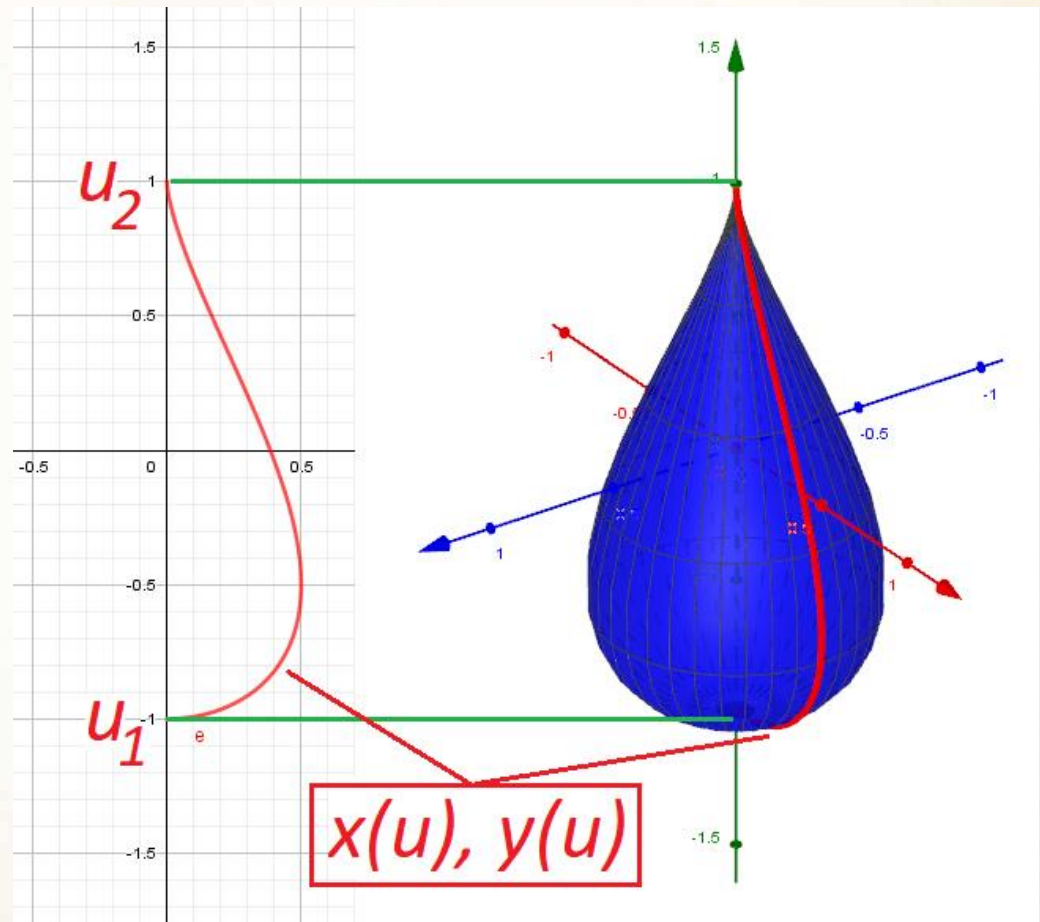
Parametrinis sukinio paviršius:

$$x = x(u) \cdot \cos(v),$$

$$y = y(u),$$

$$z = x(u) \cdot \sin(v),$$

$$u \in [u_1, u_2), v \in [0, 2\pi).$$



Geogebra $(x(u) = \sqrt{1 - u^2} (1 - u) / (3 / 2 \sqrt{3}), y(u) = u, u_1 = -1, u_2 = 1):$
 $\text{Surface}(\sqrt{1 - u^2} (1 - u) / (3 / 2 \sqrt{3}) \cos(v), u, \sqrt{1 - u^2} (1 - u) / (3 / 2 \sqrt{3}) \sin(v), u, -1, 1, v, 0, 2\pi), \text{Curve}(\sqrt{1 - u^2} (1 - u) / (3 / 2 \sqrt{3}), u, u, -1, 1)$

Bezjė paviršiai

$$p(u, v) = \sum_{i=0}^n \sum_{j=0}^m \binom{n}{i} u^i (1-u)^{n-i} \binom{m}{j} v^j (1-v)^{m-j} K_{i,j}$$

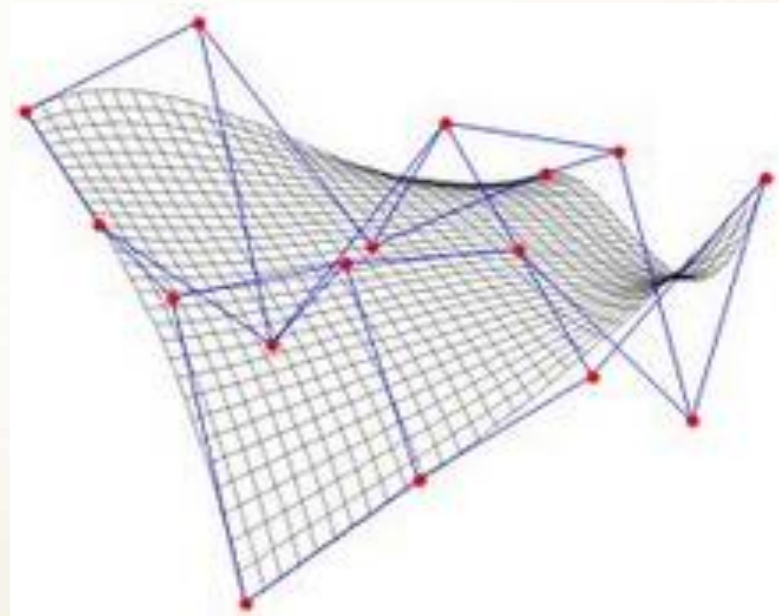
$$u, v \in [0, 1]$$

Pavyzdys, kai $n=m=3$.

Raudona spalva pažymėti kontroliniai taškai.

Mėlyna spalva pažymėtas kontrolinis tinklas.

Juoda spalva pažymėta Bezjė paviršiaus aproksimacija.



Trečios eilės Bežjė paviršius

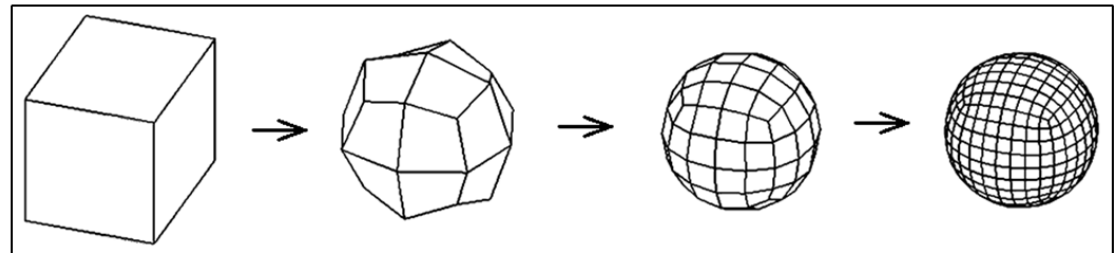
Tegu $B(u, v)$ – trečios eilės Bežjė paviršius, kurio parametrinė forma

$$B(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 \binom{3}{i} u^i (1-u)^{3-i} \binom{3}{j} v^j (1-v)^{3-j} K_{i,j}.$$

Apskaičiavus binominius koeficientus, paviršiaus parametrinę lygtį galima suvesti į matricinę forma:

$$B(u, v) = (1 \ u \ u^2 \ u^3) \begin{pmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{pmatrix} \begin{pmatrix} K_{0,0} & K_{0,1} & K_{0,2} & K_{0,3} \\ K_{1,0} & K_{1,1} & K_{1,2} & K_{1,3} \\ K_{2,0} & K_{2,1} & K_{2,2} & K_{2,3} \\ K_{3,0} & K_{3,1} & K_{3,2} & K_{3,3} \end{pmatrix} \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ v \\ v^2 \\ v^3 \end{pmatrix}.$$

$K_{i,j}$ – kontroliniai taškai, pagal kurių išsidėstymą suformuojamas Bežjė paviršius (arba C2 paviršiaus fragmentas).



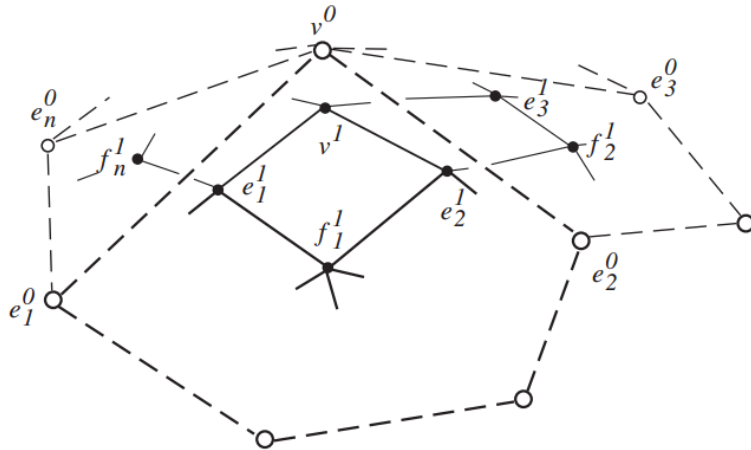
Catmull-Clark algoritmo 3 iteracijų vizualizacija.

Catmull ir Clarko algoritmas (1978 m.)

Subdivision Rules

The Catmull-Clark smooth subdivision rules for face, edge, and vertex points, are defined as:

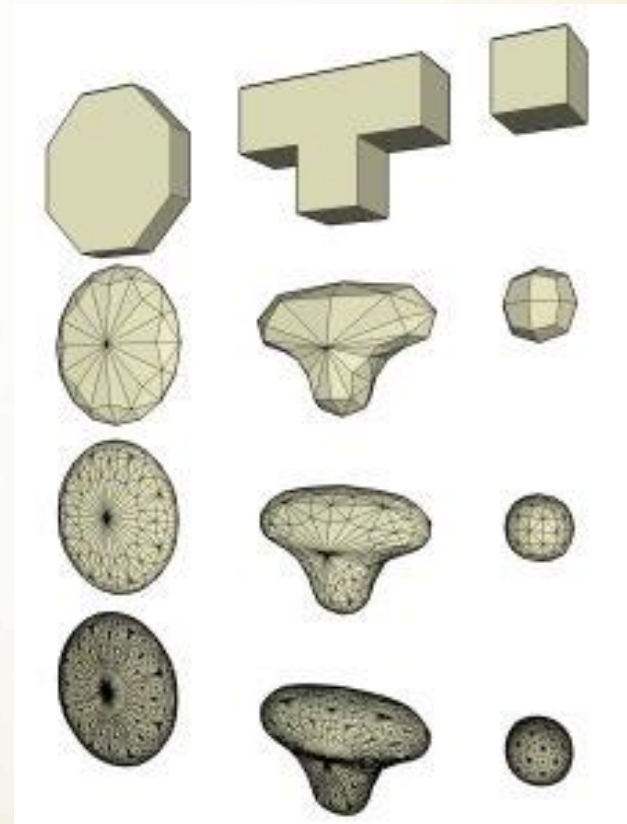
- Faces rule: f^{i+1} is the centroid of the vertices surrounding the face.
- Edge rule: $e_j^{i+1} = \frac{1}{4}(v^i + e_j^i + f_{j-1}^{i+1} + f_j^{i+1})$,
- Vertex rule: $v^{i+1} = \frac{n-2}{n}v^i + \frac{1}{n^2} \sum_j e_j^i + \frac{1}{n^2} \sum_j f_j^{i+1}$.



The following *sharp rules* are used for both boundaries and sharp edges (crease rules):

- $e_j^{i+1} = \frac{1}{2}(v^i + e_j^i)$
- $v_j^{i+1} = \frac{1}{8}(e_j^i + 6v^i + e_k^i)$

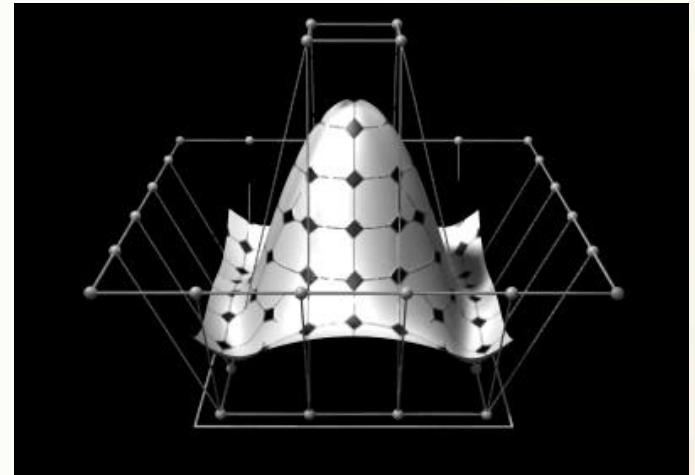
Pavyzdžiai:



C2 paviršius

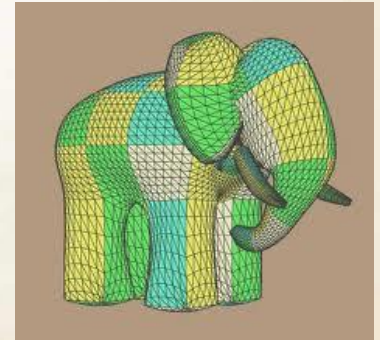
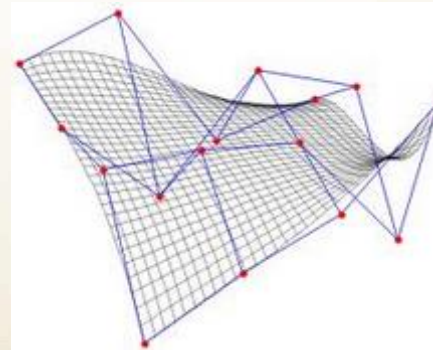
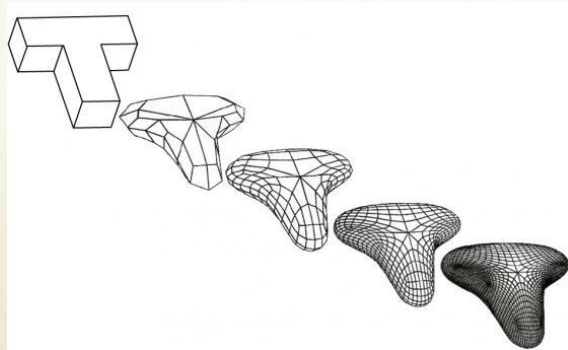
Glodus paviršius, sudarytas iš atskirų mažesnių paviršių, kurių sandūroje sutampa antros eilės išvestinės (kreivumas κ) kiekvieno tokio paviršiaus atžvilgiu:

$$\kappa = \frac{|x' y'' - y' x''|}{(x'^2 + y'^2)^{\frac{3}{2}}}$$



C2 paviršiaus konstravimo metodai:

Catmull'o ir Clark'o algoritmas **Aproksimavimo Bežė paviršiais metodas**



C2 paviršiaus konstravimas iš Bežjė paviršių (1)

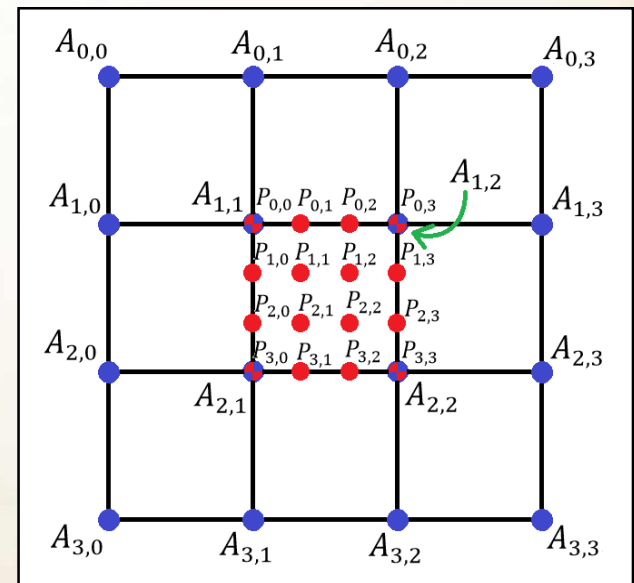
Trečios eilės Bežjė kreivių tenzorinės sandaugos koeficientai (kaukės):

$$\left[\frac{1}{6}, \frac{2}{3}, \frac{1}{6} \right] \times \left[\frac{1}{6}, \frac{2}{3}, \frac{1}{6} \right] = \begin{bmatrix} \frac{1}{36} & \frac{1}{9} & \frac{1}{36} \\ \frac{1}{9} & \frac{4}{9} & \frac{1}{9} \\ \frac{1}{36} & \frac{1}{9} & \frac{1}{36} \end{bmatrix}$$

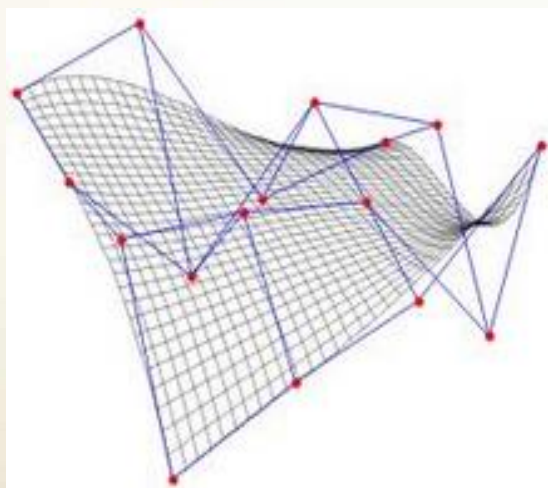
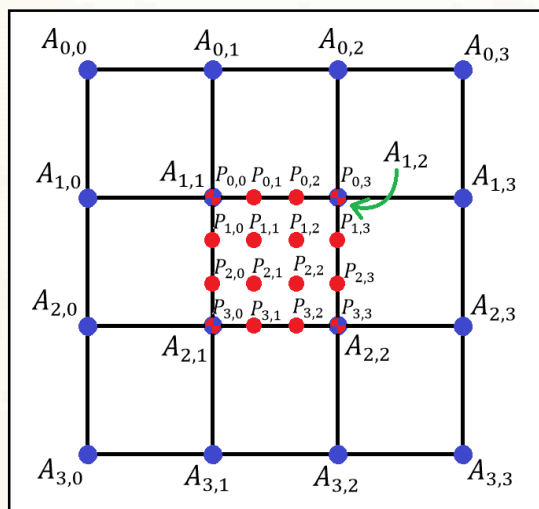
$$\left[\frac{1}{3}, \frac{2}{3} \right] \times \left[\frac{1}{3}, \frac{2}{3} \right] = \begin{bmatrix} \frac{1}{18} & \frac{2}{9} & \frac{1}{18} \\ \frac{1}{9} & \frac{4}{9} & \frac{1}{9} \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{9} & \frac{2}{9} \\ \frac{2}{9} & \frac{4}{9} \end{bmatrix} = \left[\frac{1}{6}, \frac{2}{3}, \frac{1}{6} \right] \times \left[\frac{1}{3}, \frac{2}{3} \right]$$

Pagal šias kaukes apskaičiuojami Bežjė paviršiaus kontroliniai taškai.



Kontrolinių Bežjė paviršiaus taškų apskaičiavimas



$$P_{0,0} = \frac{1}{36}A_{0,0} + \frac{1}{36}A_{0,2} + \frac{1}{36}A_{2,0} + \frac{1}{36}A_{2,2} + \frac{1}{9}A_{0,1} + \frac{1}{9}A_{1,0} + \frac{1}{9}A_{1,2} + \frac{1}{9}A_{2,1} + \frac{4}{9}A_{1,1},$$

$$P_{0,1} = \frac{1}{18}A_{0,2} + \frac{1}{18}A_{2,2} + \frac{1}{9}A_{0,1} + \frac{1}{9}A_{2,1} + \frac{2}{9}A_{1,2} + \frac{4}{9}A_{1,1},$$

$$P_{0,2} = \frac{1}{18}A_{0,1} + \frac{1}{18}A_{2,1} + \frac{1}{9}A_{0,2} + \frac{1}{9}A_{2,2} + \frac{2}{9}A_{1,1} + \frac{4}{9}A_{1,2},$$

$$P_{0,3} = \frac{1}{36}A_{0,1} + \frac{1}{36}A_{0,3} + \frac{1}{36}A_{2,1} + \frac{1}{36}A_{2,3} + \frac{1}{9}A_{0,2} + \frac{1}{9}A_{1,1} + \frac{1}{9}A_{1,3} + \frac{1}{9}A_{2,2} + \frac{4}{9}A_{1,2},$$

$$P_{1,0} = \frac{1}{18}A_{2,0} + \frac{1}{18}A_{2,2} + \frac{1}{9}A_{1,0} + \frac{1}{9}A_{1,2} + \frac{2}{9}A_{2,1} + \frac{4}{9}A_{1,1},$$

$$P_{1,1} = \frac{1}{9}A_{2,2} + \frac{2}{9}A_{1,2} + \frac{2}{9}A_{2,1} + \frac{4}{9}A_{1,1},$$

$$P_{1,2} = \frac{1}{9}A_{2,1} + \frac{2}{9}A_{1,1} + \frac{2}{9}A_{2,2} + \frac{4}{9}A_{1,2},$$

$$P_{1,3} = \frac{1}{18}A_{2,1} + \frac{1}{18}A_{2,3} + \frac{1}{9}A_{1,1} + \frac{1}{9}A_{1,3} + \frac{2}{9}A_{2,2} + \frac{4}{9}A_{1,2},$$

$$P_{2,0} = \frac{1}{18}A_{1,0} + \frac{1}{18}A_{1,2} + \frac{1}{9}A_{2,0} + \frac{1}{9}A_{2,2} + \frac{2}{9}A_{1,1} + \frac{4}{9}A_{2,1},$$

$$P_{2,1} = \frac{1}{9}A_{1,2} + \frac{2}{9}A_{1,1} + \frac{2}{9}A_{2,2} + \frac{4}{9}A_{2,1},$$

$$P_{2,2} = \frac{1}{9}A_{1,1} + \frac{2}{9}A_{1,2} + \frac{2}{9}A_{2,1} + \frac{4}{9}A_{2,2},$$

$$P_{2,3} = \frac{1}{18}A_{1,1} + \frac{1}{18}A_{1,3} + \frac{1}{9}A_{2,1} + \frac{1}{9}A_{2,3} + \frac{2}{9}A_{1,2} + \frac{4}{9}A_{2,2},$$

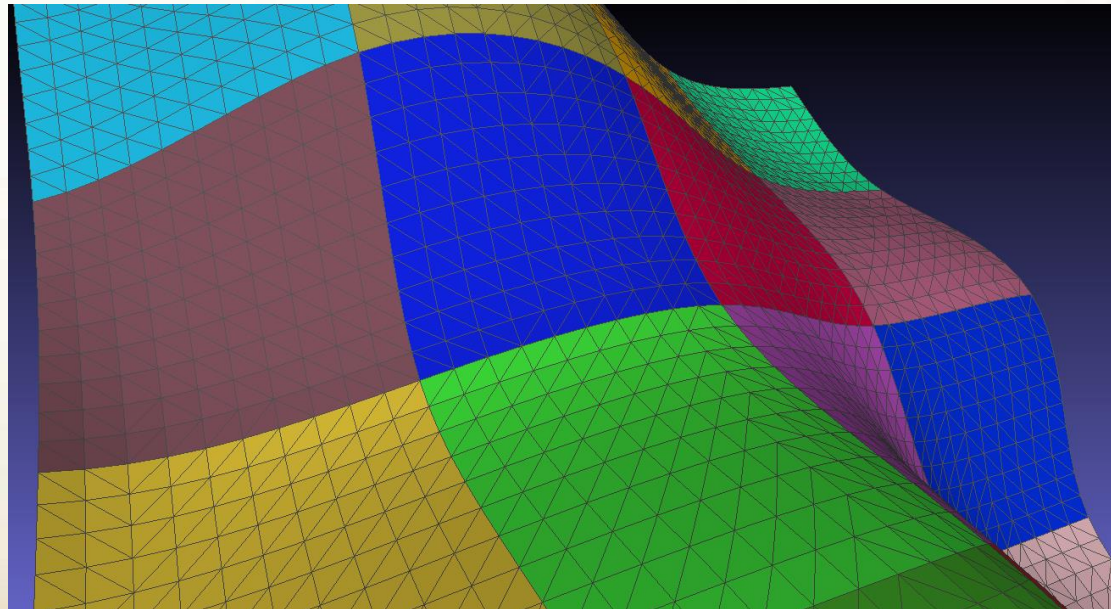
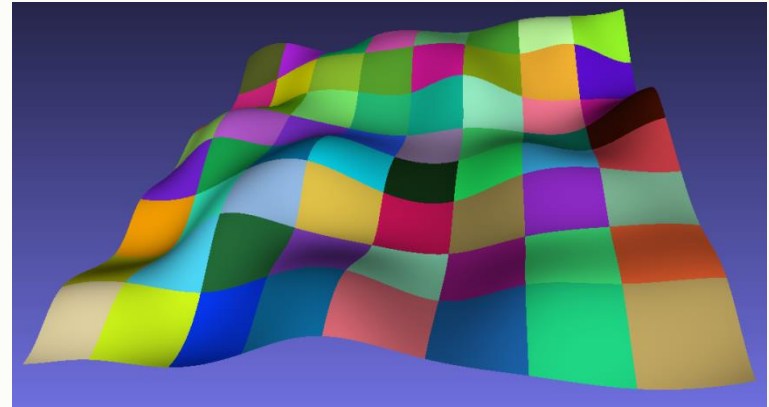
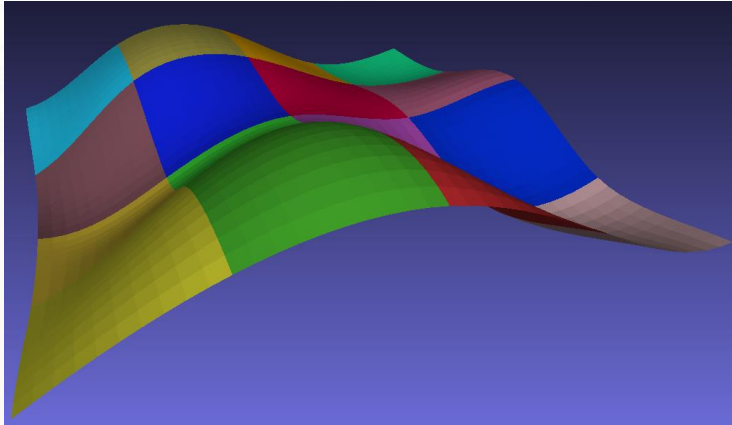
$$P_{3,0} = \frac{1}{36}A_{1,0} + \frac{1}{36}A_{1,2} + \frac{1}{36}A_{3,0} + \frac{1}{36}A_{3,2} + \frac{1}{9}A_{1,1} + \frac{1}{9}A_{2,0} + \frac{1}{9}A_{2,2} + \frac{1}{9}A_{3,1} + \frac{4}{9}A_{2,1},$$

$$P_{3,1} = \frac{1}{18}A_{1,2} + \frac{1}{18}A_{3,2} + \frac{1}{9}A_{1,1} + \frac{1}{9}A_{3,1} + \frac{2}{9}A_{2,2} + \frac{4}{9}A_{2,1},$$

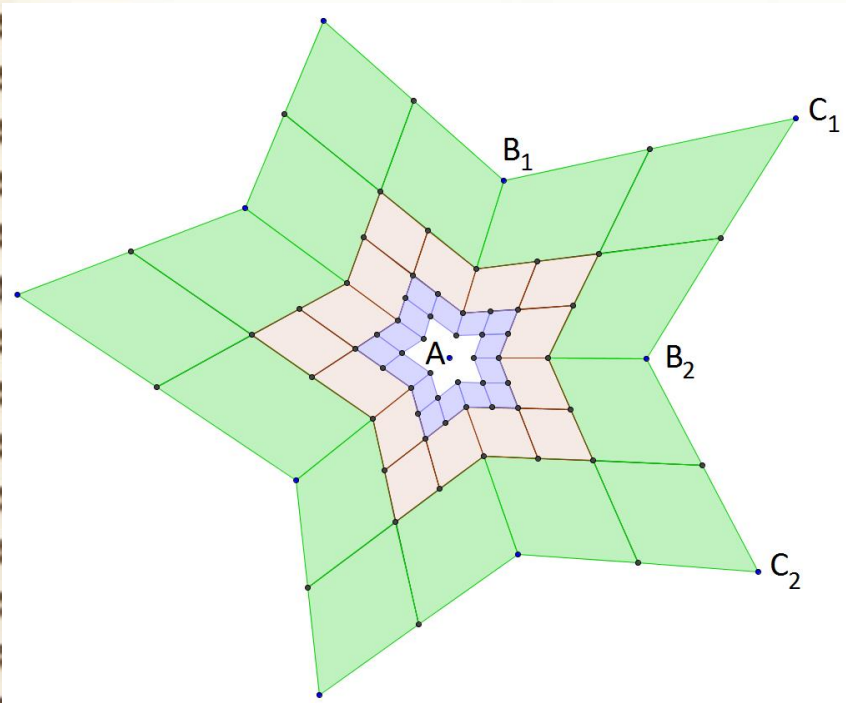
$$P_{3,2} = \frac{1}{18}A_{1,1} + \frac{1}{18}A_{3,1} + \frac{1}{9}A_{1,2} + \frac{1}{9}A_{3,2} + \frac{2}{9}A_{2,1} + \frac{4}{9}A_{2,2},$$

$$P_{3,3} = \frac{1}{36}A_{1,1} + \frac{1}{36}A_{1,3} + \frac{1}{36}A_{3,1} + \frac{1}{36}A_{3,3} + \frac{1}{9}A_{1,2} + \frac{1}{9}A_{2,1} + \frac{1}{9}A_{2,3} + \frac{1}{9}A_{3,2} + \frac{4}{9}A_{2,2}.$$

C2 paviršiaus konstravimas iš Bezjė paviršių. Pavyzdžiai



Bezjė paviršių konstravimas ypatingųjų taškų aplinkoje



Naudojamos šios kaukės paviršių dalinimui.

Naujo taško koordinatės viršūnės vietoje:

$$\begin{bmatrix} \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

Naujo taško koordinatės briaunos vietoje:

$$\begin{bmatrix} \frac{1}{16} & \frac{3}{8} & \frac{1}{16} \\ \frac{1}{16} & \frac{3}{8} & \frac{1}{16} \end{bmatrix}$$

Naujo taško koordinatės sienos vietoje:

$$\begin{bmatrix} \frac{1}{64} & \frac{3}{32} & \frac{1}{64} \\ \frac{3}{32} & \frac{9}{16} & \frac{3}{32} \\ \frac{1}{64} & \frac{3}{32} & \frac{1}{64} \end{bmatrix}$$

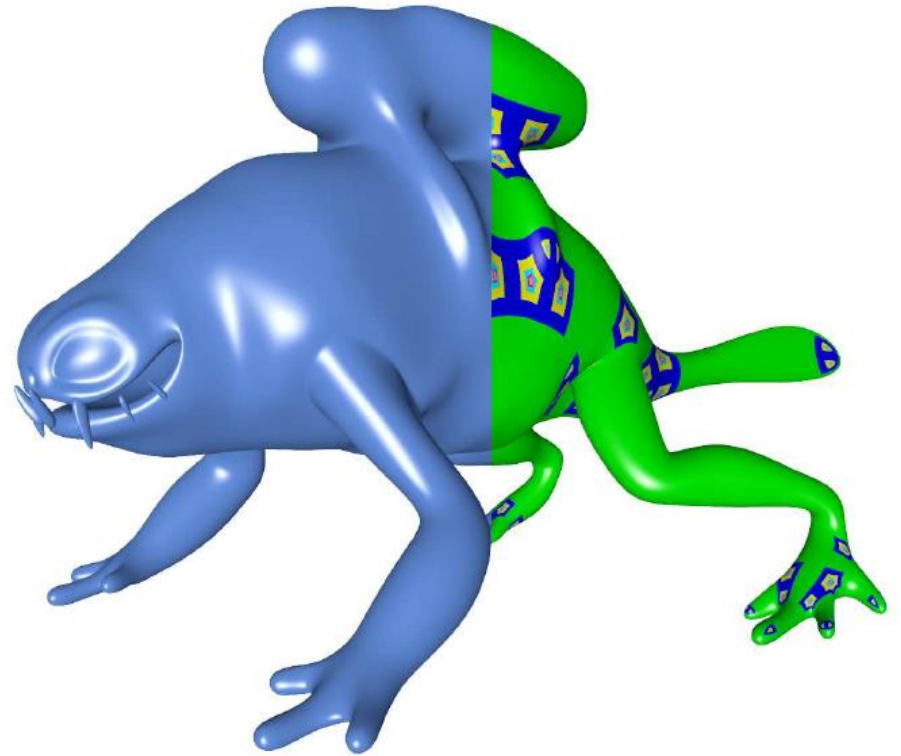
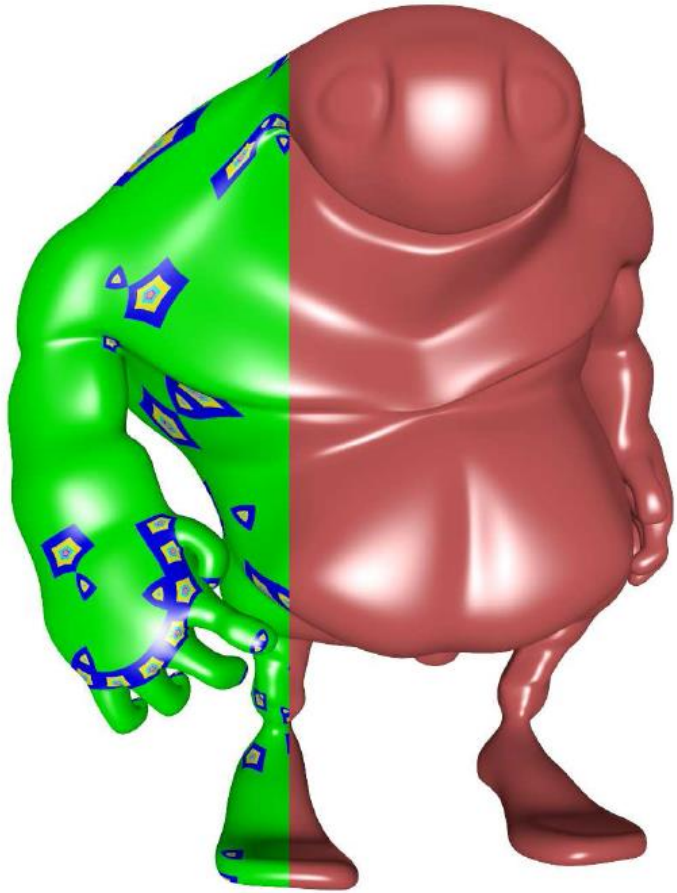
$$\text{centras} = \frac{n}{n+2}A + \frac{4n}{n(n+5)} \sum_{i=1}^n B_i + \frac{1}{n(n+5)} \sum_{i=1}^n C_i$$

K. Karčiauskas, J. Peters (2007)



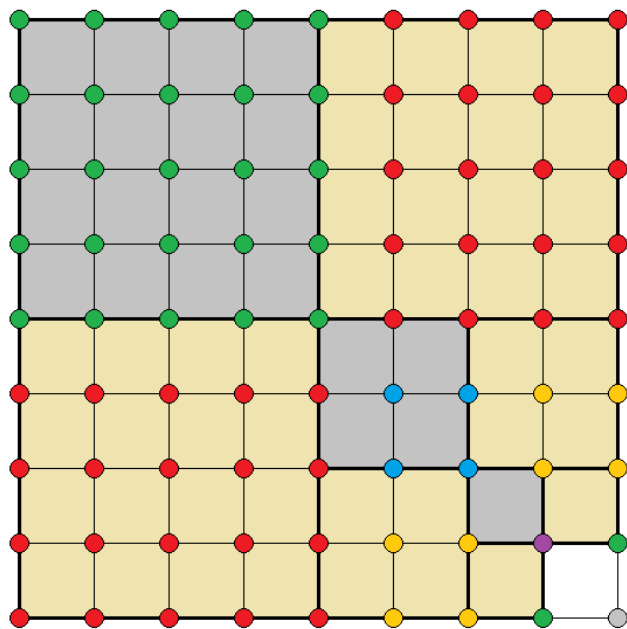
Programa vizualizavimui: **BezierView**,
failo formatas: **.bw**.

Matthias Nießner, Charles Loop, Microsoft Research, Mark Meyer and Tony DeRose (2012)



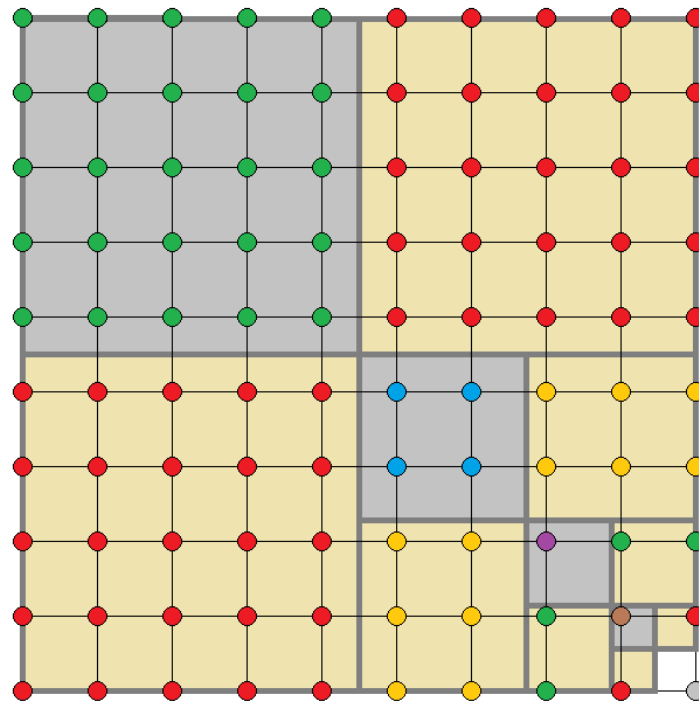
Begalinio proceso keitimas baigtiniu (1)

$n = 8$



0.000000, 0.250000, 0.500000, 0.750000, 1.000000,
 0.500000, 1.000000,
 1.000000,
 0.250000, 0.500000, 0.750000, 1.000000,
 0.500000, 1.000000,
 1.000000,

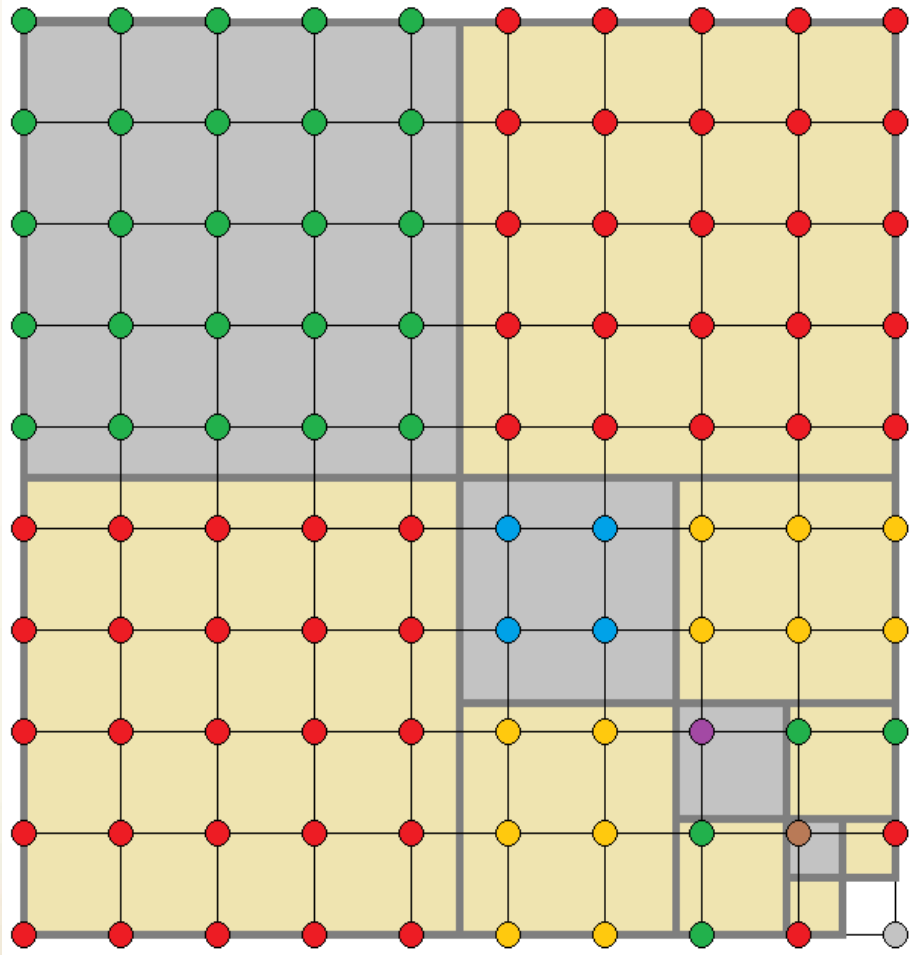
$n = 9$



0.000000, 0.222222, 0.444444, 0.666667, 0.888889,
 0.222222, 0.666667,
 0.222222,
 0.222222,
 0.111111, 0.333333, 0.555556, 0.777778, 1.000000,
 0.111111, 0.555556, 1.000000,
 0.111111, 1.000000,
 1.000000,

Iteracijų skaičius: $\lceil \log_2(n - 1) \rceil + 1$

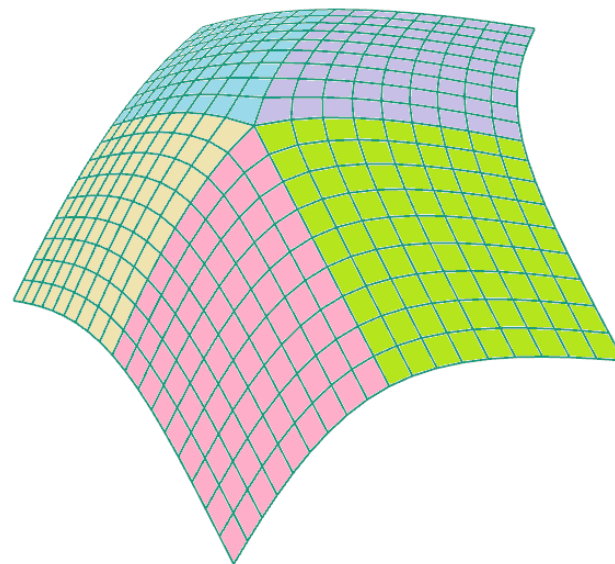
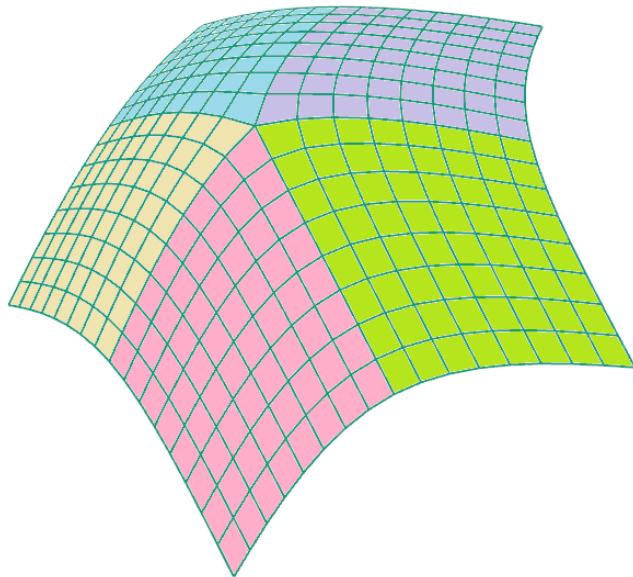
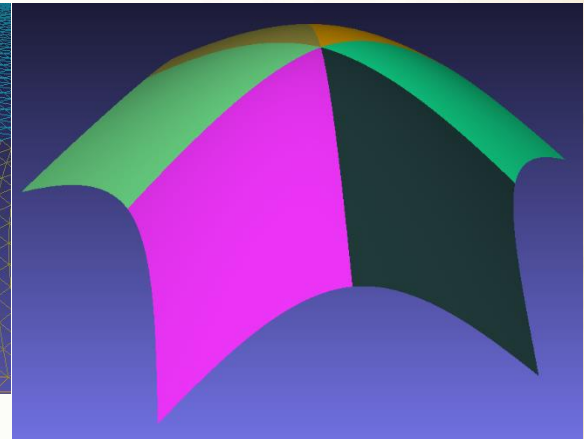
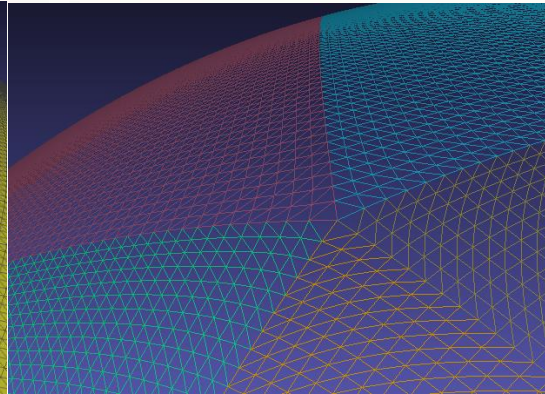
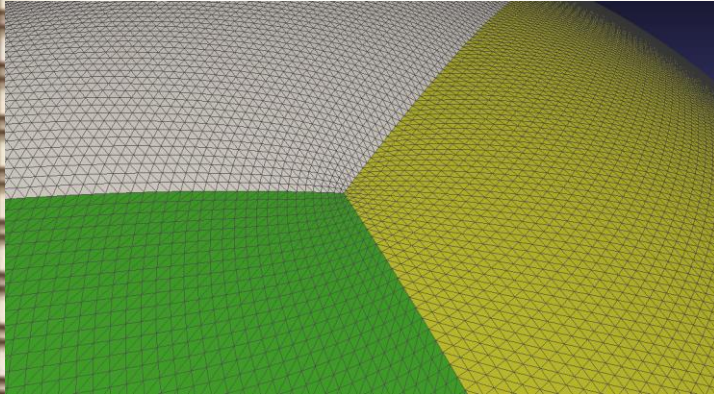
Begalinio proceso keitimas baigtiniu (2)



Tinklas(n)

1. $d \rightarrow \lceil \log_2(n - 1) \rceil + 1$
2. $m_1 \rightarrow \lfloor \frac{n+2}{2} \rfloor$
3. $m_2 \rightarrow n + 1 - m_1$
4. **for** $i \rightarrow 1$ **to** d **do**
5. $p \rightarrow \frac{2^i m_2}{n} - 1$
6. **for** $j \rightarrow 1$ **to** m_1 **do**
7. $T_1[i][m_1 - j + 1] \rightarrow 1 - p - \frac{(j-1)2^i}{n}$
8. **for** $k \rightarrow 1$ **to** m_2 **do**
9. $T_2[i][m_2 - k + 1] \rightarrow 1 - \frac{(k-1)2^i}{n}$
10. $m_1 \rightarrow \lfloor \frac{m_2}{2} \rfloor$
11. $m_2 \rightarrow m_2 - m_1$
12. **return** T_1, T_2

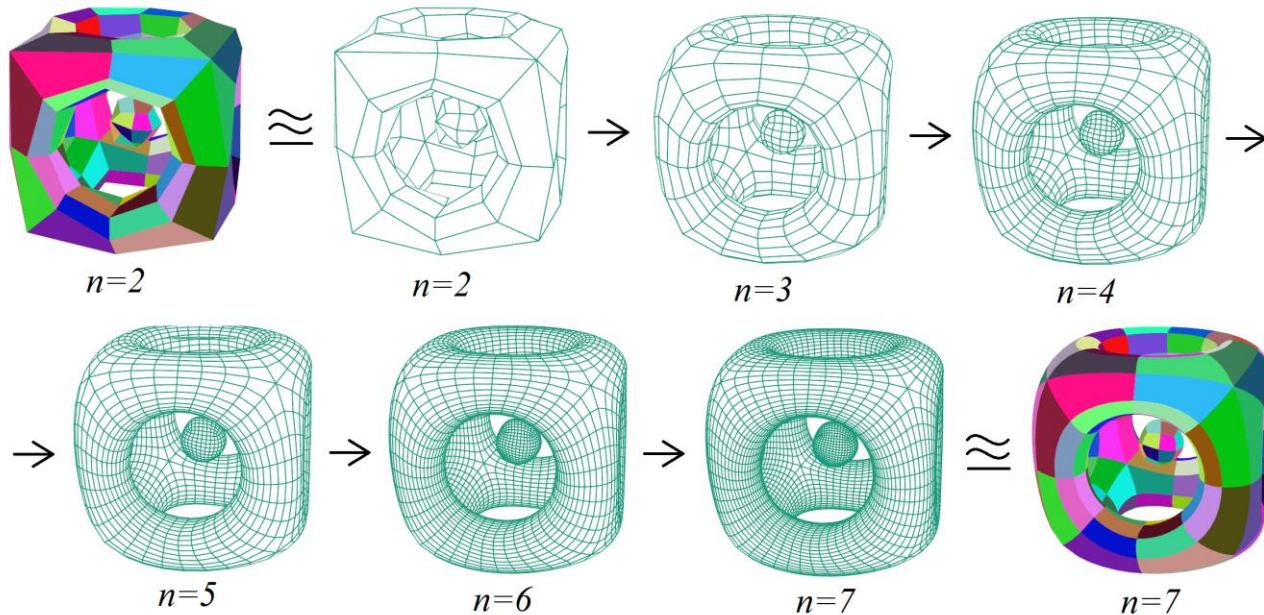
C2 paviršiaus formavimas ypatingųjų taškų srityse



Bendroji algoritmo idėja

$C2_paviršius(n)$

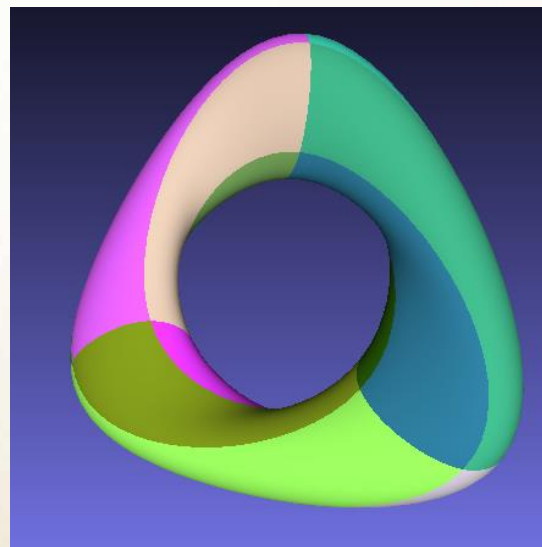
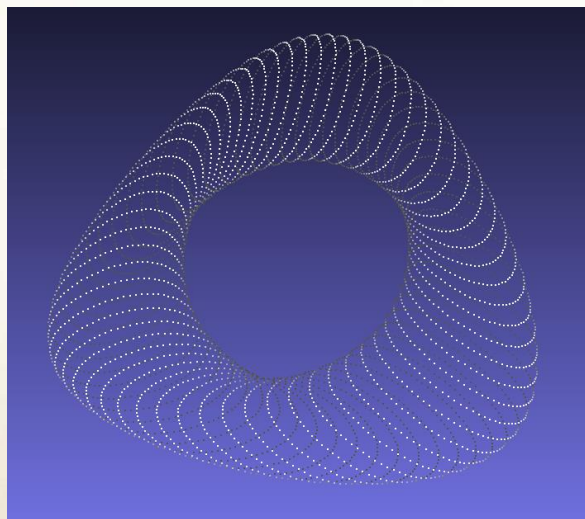
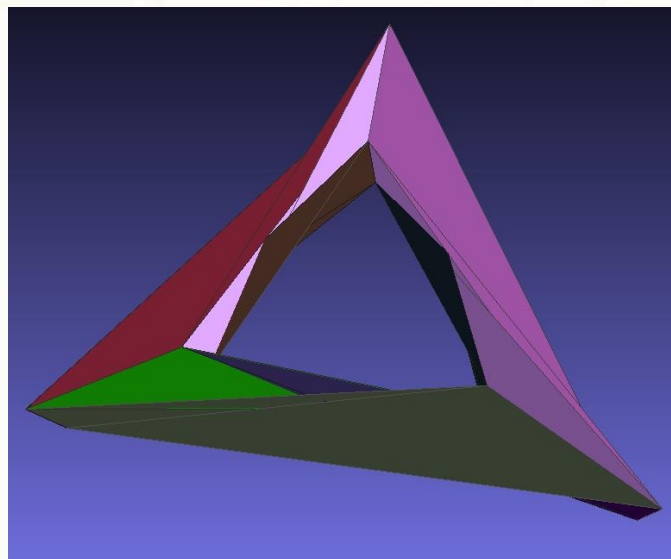
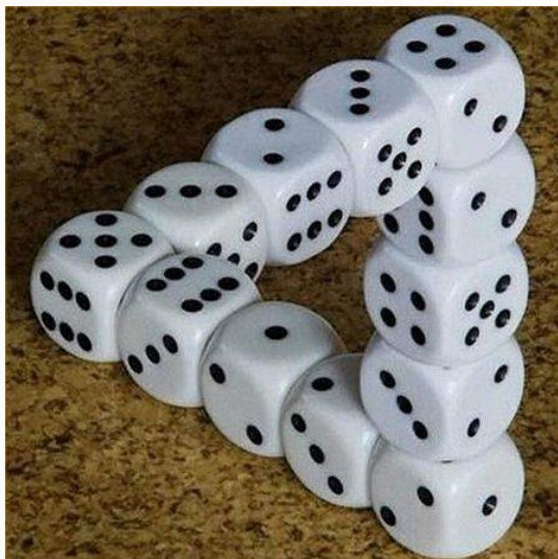
1. Inicializacija,
2. Tinklas(n),
3. Realizuok $\lceil \log_2(n - 1) \rceil + 1$ kartų Catmull-Clark algoritmą keturkampiams, kurių viršūnės – ypatingieji taškai,
4. Apskaičiuok visų Bežjė paviršių lygtis,
5. Į šias lygtis įsistatyk atitinkamas $u, v \in [0, 1]$ parametrų reikšmes ir apskaičiuok $C2$ paviršiaus taškus,
6. Susiedamas šiuos taškus po 4 išsaugok rezultata, kuris atitinka $C2$ paviršiaus aproksimaciją.



Algoritmo taikymas

- Ši Bežjė paviršių aproksimavimo metodika gali būti taikoma optimizavimo uždaviniuose, kuriuose pagrindinis tikslas maksimizuoti C2 paviršiaus detalumą, kai:
 - kompiuterio pastovioji atmintis, skirta $C2_paviršius(n)$ algoritmo rezultato išsaugojimui, turi tam tikrus apribojimus;
 - yra žinoma perteklinė informacijos riba, pavyzdžiui, naudojant 3D spausdintuvą trimačiams objektams suformuoti, papildomas C2 paviršiaus taškų skaičius neturi įtakos rezultato kokybei.

„Nejmanomas“ trikampis



Loop-Schaefer algoritmas (2008)



US 20080043023A1

(19) **United States**
 (12) **Patent Application Publication** (10) **Pub. No.: US 2008/0043023 A1**
 Loop et al. (43) **Pub. Date: Feb. 21, 2008**

(54) **APPROXIMATING SUBDIVISION SURFACES WITH BEZIER PATCHES**

Publication Classification

(75) Inventors: **Charles T. Loop**, Bellevue, WA (US); **Scott David Schaefer**, Bryan, TX (US)

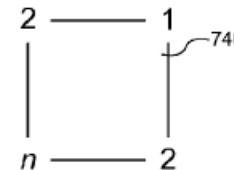
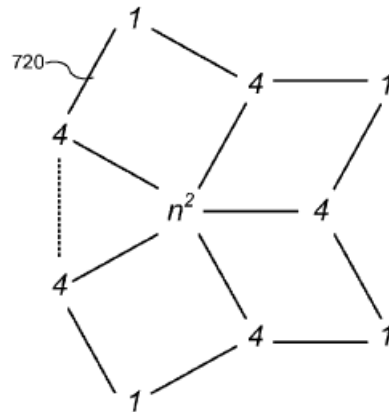
(51) **Int. Cl.** *G06T 11/20* (2006.01)
 (52) **U.S. Cl.** 345/441

Correspondence Address:
KLARQUIST SPARKMAN LLP
121 S.W. SALMON STREET, SUITE 1600
PORTLAND, OR 97204

(57) **ABSTRACT**

Surface modeling systems and techniques are described which approximate Catmull-Clark subdivision surfaces. A quadrilateral mesh is analyzed by applying shape and tangent masks to faces on the quad-mesh. Through application of the shape masks, a shape patch is created which approximates the subdivision limit surface. This shape patch can be used for rendering surface shape. Through application of tangent masks, tangent patches are created which comprise tangent vectors which give rise to continuous normal vector fields, which can be used for shading of the surface.

(73) Assignee: **Microsoft Corporation**, Redmond, WA (US)
 (21) Appl. No.: **11/464,800**
 (22) Filed: **Aug. 15, 2006**



Masks for approximating shape

Ačiū už dėmesį.