

RAČUNARSKA GRAFIKA

Oznaka predmeta: RAG

Predavanje broj: 05

Nastavna jedinica: Geometrijske transformacije , 2D

Nastavne teme:

Matrična predstava, homogene koordinate, 2D transformacije: translacija, rotacija, skaliranje, refleksija, smicanje, složene transformacije, primena.

Predavač: prof. dr Perica S. Štrbac, dipl. ing.

Literatura:

Dragan Cvetković : "Računarska grafika", CET, Beograd, 2006.

James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes:
"Computer Graphics: Principles and Practice", 2nd ed. in C,
Addison-Wesley, 2000.

Geometrijske transformacije

- Geometrijske transformacije preslikavaju originalnu tačku u njenu sliku
- Postoje dve konvencije za geometrijske transformacije kretanja (translacije i rotacije):
 - konvencija pokretnog koordinatnog sistema (pokretne virtuelne kamere, odnosno posmatrača)
 - konvencija pokretnog objekta
- Konvencije se mogu kombinovati (kreće se i objekat i posmatrač)
- Ovde se podrazumeva
 - da se tačka nalazi u desnom pravouglom koordinatnom sistemu
 - da se primenjuje konvencija pokretne virtuelne kamere

2D transformacije

- 2D desni pravougli koordinatni sistem je definisan na sledeći način:
 - koordinatna osa prve koordinate (X) se prevodi u osu druge koordinate (Y) rotacijom oko koordinatnog početka za 90° u smeru suprotnom od kretanja kazaljke na časovniku
- U slučaju afinih transformacija, ako su koordinate originalne tačke (x,y), koordinate neke tačke nakon transformacije su (x',y',1):

$$x' = A1 \cdot x + B1 \cdot y + C1 \cdot 1$$

$$y' = A2 \cdot x + B2 \cdot y + C2 \cdot 1$$

$$1 = 0 \cdot x + 0 \cdot y + 1 \cdot 1$$

Matrična predstava transformacija

- Poslednjim identitetom se proširuje sistem jednačina, da bi se predstavio u matričnom obliku:

$$Q=P \cdot M \qquad [x' \quad y' \quad 1] = [x \quad y \quad 1] \cdot \begin{bmatrix} A1 & A2 & 0 \\ B1 & B2 & 0 \\ C1 & C2 & 1 \end{bmatrix}$$

gde je:

Q –slika tačke,

P –original tačke,

M –matrica transformacija

- Drugi oblik matrične jednačine: $Q^T = M^T \cdot P^T$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A1 & B1 & C1 \\ A2 & B2 & C2 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Homogene koordinate

- Predstava tačke u vektorskom obliku $[x \ y]$ se proširuje trećom koordinatom jednakom 1: $[x \ y \ 1]$
- Koordinatni sistem za ovakvo predstavljanje tačke se naziva sistemom sa *homogenim koordinatama*
- Homogene koordinate čine matematički aparat uniformnim za sve transformacije
- Transformacija se predstavlja adekvatnom matricom transformacije M
- Treća kolona matrice M je konstantna za sve 2D transformacije
- Jednom izračunata matrica transformacije se koristi
 - da bi se sve relevantne tačke neke 2D slike transformisale na jedinstven način
- U petlji se množe vektori originalnih tačaka matricom transformacije:
for (i=0; i<n; i++) transform(p[i],&q[i],M);
- Elementarnim transformacijama u 2D se smatraju
 - translacija,
 - rotacija,
 - skaliranje
 - smicanje

Translacija

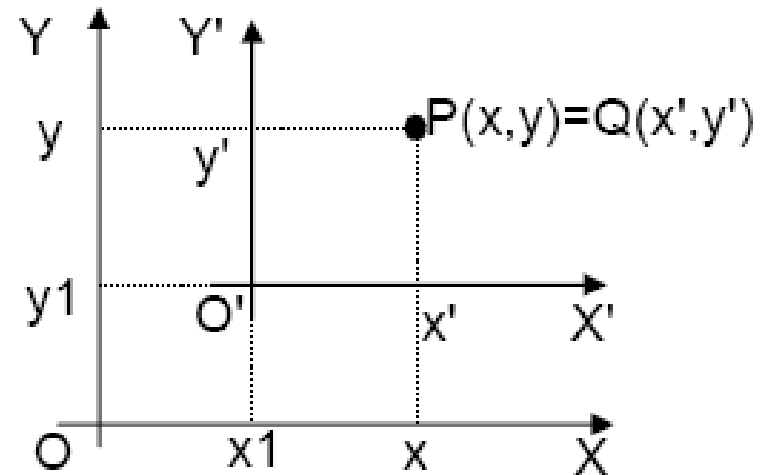
- Koordinatni početak $O(0,0)$ se translatorno pomera u tačku $O'(x_1, y_1)$:

$$x' = 1 \cdot x + 0 \cdot y - x_1 \cdot 1$$

$$y' = 0 \cdot x + 1 \cdot y - y_1 \cdot 1$$

$$1 = 0 \cdot x + 0 \cdot y + 1 \cdot 1$$

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -x_1 & -y_1 & 1 \end{bmatrix}$$

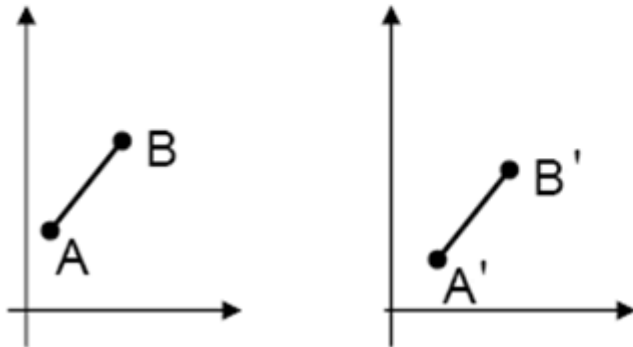


- Elementi C_1 i C_2 u opštoj matrici M
 - nazivaju se translatornim elementima

Translacija -primer

- Zadat je linijski segment krajnjim tačkama A(1,3) i B(4,6)
 - izvršiti translaciju koordinatnog sistema tako da se koordinatni početak premesti u tačku O'(-1,1)

$$A' = [1 \ 3 \ 1] \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & -1 & 1 \end{bmatrix} = [2 \ 2 \ 1] \quad B' = [4 \ 6 \ 1] \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & -1 & 1 \end{bmatrix} = [5 \ 5 \ 1]$$



Rotacija

- Koordinatni sistem rotira oko centra u koordinatnom početku za ugao α u smeru nasuprot kretanja kazaljke na časovniku

$$x' = (x + y \cdot \operatorname{tg} \alpha) \cdot \cos \alpha$$

$$y = x \cdot \operatorname{tg} \alpha + y' / \cos \alpha$$

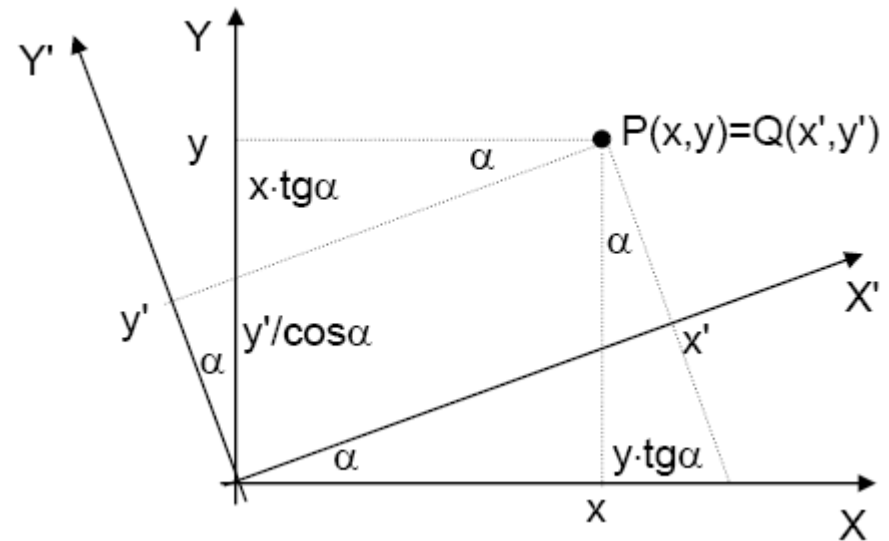
$$x' = x \cdot \cos \alpha + y \cdot \sin \alpha$$

$$y' = -x \cdot \sin \alpha + y \cdot \cos \alpha$$

$$x' = \cos \alpha \cdot x + \sin \alpha \cdot y + 0 \cdot 1$$

$$y' = -\sin \alpha \cdot x + \cos \alpha \cdot y + 0 \cdot 1$$

$$1 = 0 \cdot x + 0 \cdot y + 1 \cdot 1$$



- Elementi A1, A2, B1 i B2 u opštoj matrici M
– nazivaju se rotacionim elementima

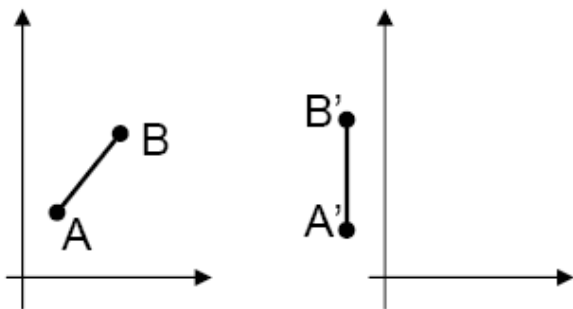
$$R = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotacija -primer

- Zadat je linijski segment krajnjim tačkama A(1,3) i B(4,6)
 - rotirati koordinatni sistem za ugao $\alpha=45^\circ$ u smeru kazaljke na časovniku

$$A' = [1 \ 3 \ 1] \cdot \begin{bmatrix} \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [-\sqrt{2} \ 2\sqrt{2} \ 1]$$

$$B' = [4 \ 6 \ 1] \cdot \begin{bmatrix} \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [-\sqrt{2} \ 5\sqrt{2} \ 1]$$



Skaliranje

- Faktori skaliranja po X i Y osi: S_x i S_y
- Efekat:
 - $S < 1 \Rightarrow$ smanjivanje objekta (udaljavanje posmatrača),
 - $S > 1 \Rightarrow$ povećanje objekta (približavanje posmatrača)
- Preslikavanje:
 - $x' = S_x \cdot x + 0 \cdot y + 0 \cdot 1$
 - $y' = 0 \cdot x + S_y \cdot y + 0 \cdot 1$
 - $1 = 0 \cdot x + 0 \cdot y + 1 \cdot 1$

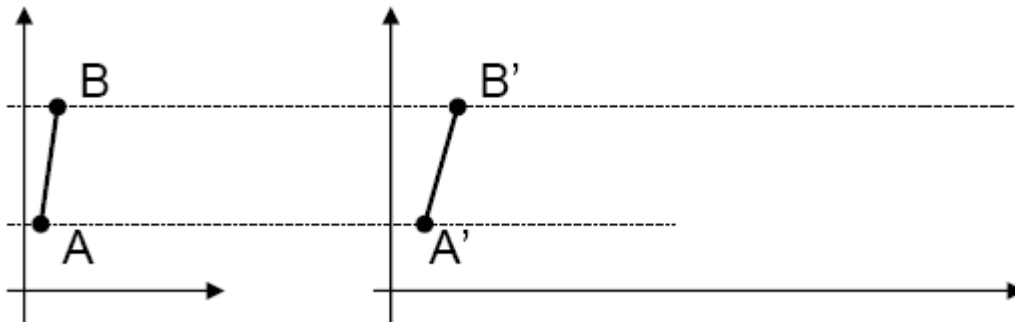
$$S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Skaliranje -primer

- Primer:
 - segment linije određen je krajnjim tačkama A(1,3) i B(2,7).
 - izvršiti skaliranje, ako su dati skala faktori $S_x=2$ i $S_y=1$

$$A' = [1 \quad 3 \quad 1] \cdot \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [2 \quad 3 \quad 1]$$

$$B' = [2 \quad 7 \quad 1] \cdot \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [4 \quad 7 \quad 1]$$



Refleksija

- Refleksija prema Y-osi:

$$S_x = -1,$$

$$S_y = 1$$

- Refleksija prema X-osi:

$$S_x = 1,$$

$$S_y = -1$$

- Refleksija prema proizvoljnoj osi:

- translacijom se dovede koordinatni početak na datu osu
- rotacijom se X-osa poklopi sa datom osom
- primeni se refleksija prema X-osi
- inverzna rotacija
- inverzna translacija

Smicanje (Iskošenje/*Shear*)

- Faktori smicanja po X i Y-osi: H_x i H_y , respektivno
- Preslikavanje:

$$x' = 1 \cdot x + H_x \cdot y + 0 \cdot 1$$

$$y' = H_y \cdot x + 1 \cdot y + 0 \cdot 1$$

$$1 = 0 \cdot x + 0 \cdot y + 1 \cdot 1$$

$$H = \begin{bmatrix} 1 & H_y & 0 \\ H_x & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

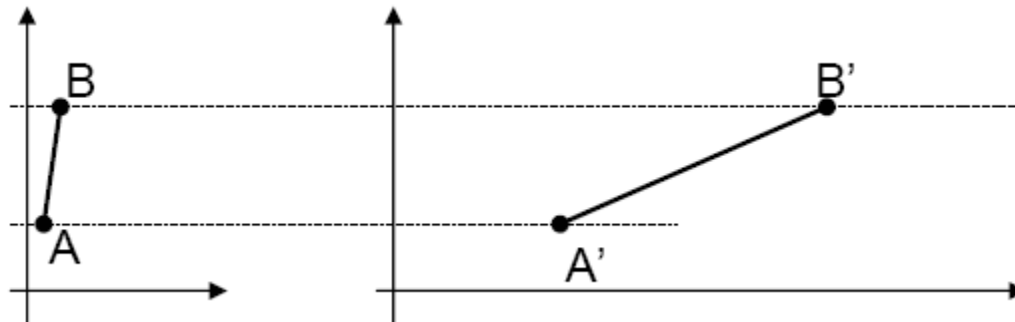
- Smicanje samo u pravcu X-ose: $H_y=0$;
- Smicanje samo u pravcu Y-ose: $H_x=0$

Smicanje -primer

- Primer:
 - segment linije određen je krajnjim tačkama A(1,3) i B(2,7)
 - izvršiti smicanje, ako su dati faktori smicanja $H_x=2$ i $H_y=0$

$$A' = [1 \quad 3 \quad 1] \cdot \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [7 \quad 3 \quad 1]$$

$$B' = [2 \quad 7 \quad 1] \cdot \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [16 \quad 7 \quad 1]$$

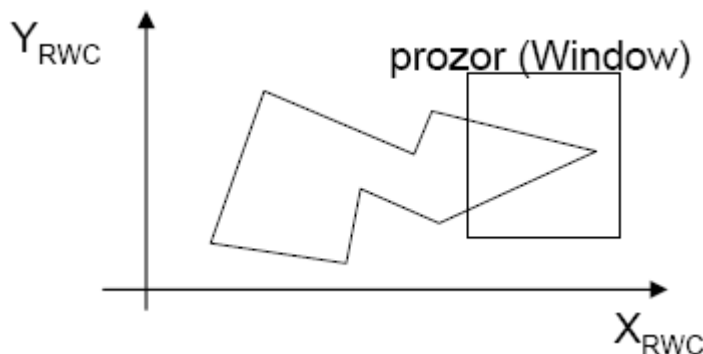


Složene transformacije

- Neka je q slika tačke p definisane sa: $q=t_3(t_2(t_1(p)))$
 - t_x su elementarne transformacije koje se primenjuju redom: $x=1, 2, 3$
- Sledi (pošto je množenje matrica asocijativno):
$$Q=((P*T_1)*T_2)*T_3 = P*(T_1*T_2*T_3) = P*T$$
 - T je kompozitna matrica složene transformacije,
 - T_1, T_2 i T_3 su matrice elementarnih transformacija koje učestvuju u složenoj transformaciji
- Redosled transformacija je veoma važan
 - **množenje matrica nije komutativno**

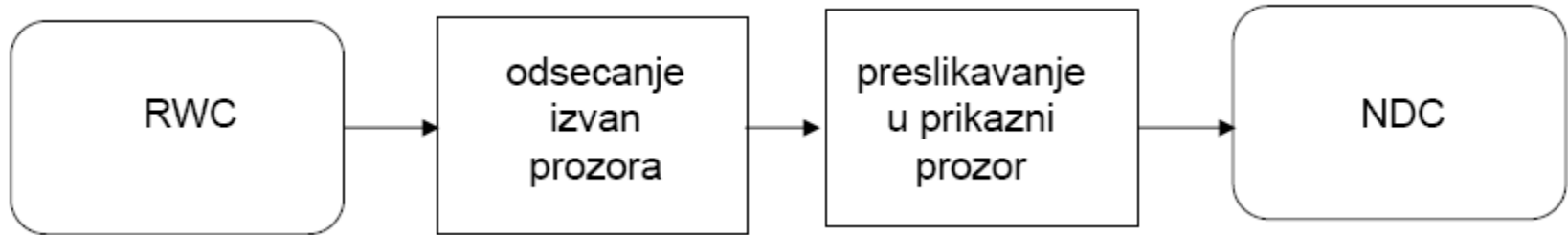
Primena 2D transformacija

- Jedna primena 2D transformacija je transformacija slike
 - iz koordinatnog sistema realnog sveta u normalizovane koordinate uređaja
- Koordinate sveta (*Real World Coordinates*, **RWC**)
 - obično u jedinicama dužine, npr. u [cm,m,Km]
- Normalizovane koordinate uređaja (*Normalized Device Coordinates*, **NDC**)
 - [0.0, ... , 1.0]
- Koordinate uređaja
 - obično u jedinicama dužine ili u pikselima
- Prozor i prikazni prozor:



Generisanje prikaza

- Proces generisanja prikaza:



- Preslikavanje (mapiranje) u prikazni prozor:
 - translacija (u RWC) tako da se koordinatni početak premesti u donji levi ugao prozora
 - skaliranje tako da se prozor preslika u prikazni prozor; pošto se koordinate prikaznog prozora daju u NDC obliku, ovim se obezbeđuje prelaz iz RWC u NDC
 - translacija (u NDC) tako da se prikazni prozor locira na željeno mesto unutar prikazne površine normalizovanog prikazivača

Primer

- Prozor je definisan pomoću sledećih relacija:
 $10 < x < 20, 8 < y < 13,$
- Prikazni prozor je u gornjem levom uglu virtuelnog ekrana, u oblasti:
 $0.0 < x_d < 0.25, 0.75 < y_d < 1.0$
- Rešenje:
 - translacija T1 koordinatnog sistema RW u donji levi ugao prozora: $O'(10,8)$
 - skaliranje S da bi se prozor preslikao u prikazni prozor:
 $S_x = (0.25 - 0.0) / (20 - 10) = 0.025$ $S_y = (1.0 - 0.75) / (13 - 8) = 0.05$
 - translacija T2 koordinatnog početka iz donjeg levog ugla prikaznog prozora u donji levi ugao virtuelnog ekrana: $O''(0.0, -0.75)$
- Kombinovanjem elementarnih transformacija dobija se kompozitna matrica totalne transformacije:

$$T1 \cdot S \cdot T2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -10 & -8 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0.025 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0.75 & 1 \end{bmatrix} = \begin{bmatrix} 0.025 & 0 & 0 \\ 0 & 0.05 & 0 \\ -0.25 & 0.35 & 1 \end{bmatrix}$$

Primer 1/4

```
#pragma comment( lib, "opengl32.lib" )
#pragma comment( lib, "glu32.lib" )
#pragma comment( lib, "glut32.lib" )
#pragma comment( linker, "/subsystem:windows /entry:mainCRTStartup")
#include <GL/glut.h> //POSLEDNJI INCLUDE
```

```
int font=(int)GLUT_STROKE_ROMAN;
int deltax=0;
int deltax=0;
const int col=3;
//neka je ovo matrica transformacije T,  $Q=P*T$ 
double T[3][3]={
    { 0.866, 0.500, 0.0},
    { -0.500, 0.866, 0.0},
    { 0.000, 0.000, 1.0}
};
```

```
//neka nam je ovo opis tacaka trougla u Afinim koordinatama
double trougao[3][3]={
    { 100.0, 100.0, 1.0}, //A
    { 100.0, 200.0, 1.0}, //B
    { 300.0, 100.0, 1.0} //C
};
```

Primer 2/4

```
//hajde da postavimo: Transformisi(m, m1, m2) gde je m=m1*m2 (moze i m*=m2)
void Transformisi(double* m, double* m1, double* m2){
    double mtemp[col][col];          int i,j,k;
    for(i=0;i<col;i++)
        for(j=0;j<col;j++){
            mtemp[i][j]=0;
            for(k=0;k<col;k++) mtemp[i][j] += m1[i*col+k] * m2[k*col+j];
        }
    for(i=0;i<col;i++) for(j=0;j<col;j++) m[i*col+j]=mtemp[i][j];
}

void SkenirajTastaturu(unsigned char c, int x, int y){
    //kolege, podesite T da se demonstrira samo ortogonalna translacija
    deltax=deltay=0.0;
    switch(c){
        case 'd':deltax= 1.0;break;
        case 'a':deltax=-1.0;break;
        case 'w':deltay= 1.0;break;
        case 's':deltay=-1.0;break;
        default: exit(0);
    }
    //mala igra i za translaciju
    T[2][0] = deltax; T[2][1] = deltay;
    Transformisi((double*)trougao, (double*)trougao, (double*)T);
    glutPostRedisplay(); }
```

Primer 3/4

```
void renderStrokeCharacter(float x, float y, float z, void *font, char *string)
{
    char *c;
    glPushMatrix();
    glTranslatef(x, y, z);
    for (c=string; *c != '\0'; c++) glutStrokeCharacter(font, *c);
    glPopMatrix();
}

void Priказuj(){
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_TRIANGLES);
        //original P
        glColor3d(255,0,0);
        glVertex2i(100,100);
        glVertex2i(100,200);
        glVertex2i(300,100);
        //slika Q
        glColor3d(0,0,255);
        glVertex2f(trougao[0][0],trougao[0][1]);
        glVertex2f(trougao[1][0],trougao[1][1]);
        glVertex2f(trougao[2][0],trougao[2][1]);
    glEnd();
    renderStrokeCharacter(0,250,0,(void *)font,"Primer");
    glFlush(); }
```

```
void main(int argc, char* argv[])
{
    glutInit(&argc,argv);
    glutInitDisplayMode(GLUT_RGB);
    glutInitWindowSize(500,500);
    int win=glutCreateWindow("Kolege, probajte 2D transformacije");

    glClearColor(0,0,0,0);
    gluOrtho2D(0,500,0,500);
    glutDisplayFunc(Prikazuji);
    glutKeyboardFunc(SkenirajTastaturu);

    //glutFullScreen();
    glutMainLoop();
}
```

Izlaz

