Test primeri

In[9]:=

Out[9]= primeri Test

1) Naivni vs. DLP algoritam

```
In[10]:= x1 = {1, 1, 1}; x2 = {5, 2, 1}; x3 = {6, 4, 1}; x4 = {-1, 7, 1};  (* originali *)

In[11]:= x1p = {0, 0, 1};
         x2p = {10, 0, 1}; x3p =  {10, 5, 1}; x4p = {0, 5, 1};  (* slike *)

In[12]:= Anaiv = Projective4pts[{x1, x2, x3, x4}, {x1p, x2p, x3p, x4p}];
         (* primenjujemo naivni algoritam *)

In[13]:= MatrixForm[Anaiv] // N (* i dobijamo matricu A *)
```

Out[13]//MatrixForm=

$$\begin{pmatrix} 0.0641026 & 0.0213675 & -0.0854701 \\ -0.0274725 & 0.10989 & -0.0824176 \\ -0.0103965 & 0.0105401 & 0.0586799 \end{pmatrix}$$

```
In[14]:= Adlp = ProjectiveDLP[{x1, x2, x3, x4}, {x1p, x2p, x3p, x4p}]
          (* DLP algoritam na iste tacke *)

Out[14]= {{0.341879, 0.11396, -0.455838},
          {-0.146519, 0.586078, -0.439558}, {-0.0554476, 0.0562137, 0.312958}}

In[15]:= (* rezultat je isti - do na proporcionalnost *)

In[16]:= (Anaiv[[1, 1]] / Adlp[[1, 1]]) Adlp // MatrixForm
```

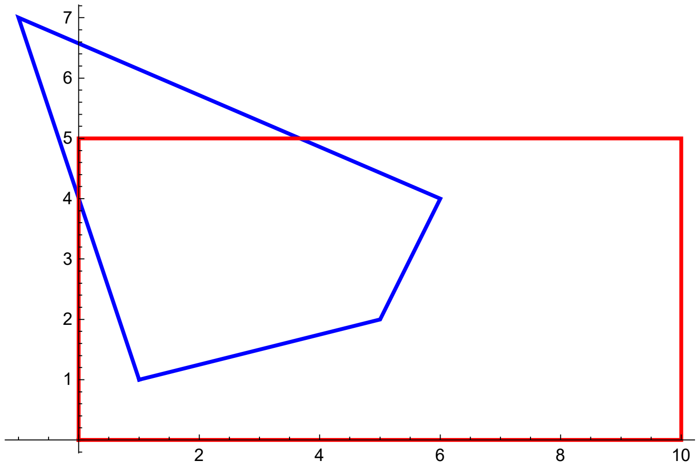
Out[16]//MatrixForm=

$$\begin{pmatrix} 0.0641026 & 0.0213675 & -0.0854701 \\ -0.0274725 & 0.10989 & -0.0824176 \\ -0.0103965 & 0.0105401 & 0.0586799 \end{pmatrix}$$

```
In[17]:= affinize[Anaiv.x5 // N]

Out[17]= affinize[{{0.0641026, 0.0213675, -0.0854701},
            {-0.0274725, 0.10989, -0.0824176}, {-0.0103965, 0.0105401, 0.0586799}}.x5]

In[18]:= orig = ShowPolygon[{x1, x2, x3, x4, x1}, Blue];

In[19]:= slika = ShowPolygon[{x1p, x2p, x3p, x4p,  x1p}, Red];
```

In[20]:=
```
Show[orig, slika, Axes → True] (* plavo=originali, crveno= slika *)
```

Out[20]=



2) Osobine DLP algoritma

In[31]:=
```
x5 = {3, 1, 1};
```

In[32]:=
```
affinize[Anaiv.x5 // N]
```

Out[32]= {3.3711, -1.44476}

Ovo bi trebala biti slika preslikavanjem (odredjenim sa 4 para tacaka) pete tacke x5, u afinim koordinatama. Ali mi cemo zaokruziti tacku x5p

In[33]:=
```
x5p = {3, -1, 1};
```

In[34]:=
```
Adlp5 = ProjectiveDLP[{x1, x2, x3, x4, x5}, {x1p, x2p, x3p, x4p, x5p}];
```

In[35]:= **Dobijeno projektivno preslikavanje nije isto kao ono sa 4 tacke, ali je priblizno**

Syntax::sntxf: "Dobijeno projektivno preslikavanje nije isto kao ono sa 4 tacke"
cannot be followed by ", ali je priblizno".

Out[35]= ali je priblizno

In[36]:=
```
MatrixForm /@ {Adlp, Adlp5}
```

Out[36]= $\left\{\begin{pmatrix} 0.341879 & 0.11396 & -0.455838 \\ -0.146519 & 0.586078 & -0.439558 \\ -0.0554476 & 0.0562137 & 0.312958 \end{pmatrix}, \begin{pmatrix} 0.343498 & 0.118237 & -0.48379 \\ -0.152467 & 0.585069 & -0.395034 \\ -0.0573214 & 0.0555633 & 0.325575 \end{pmatrix}\right\}$

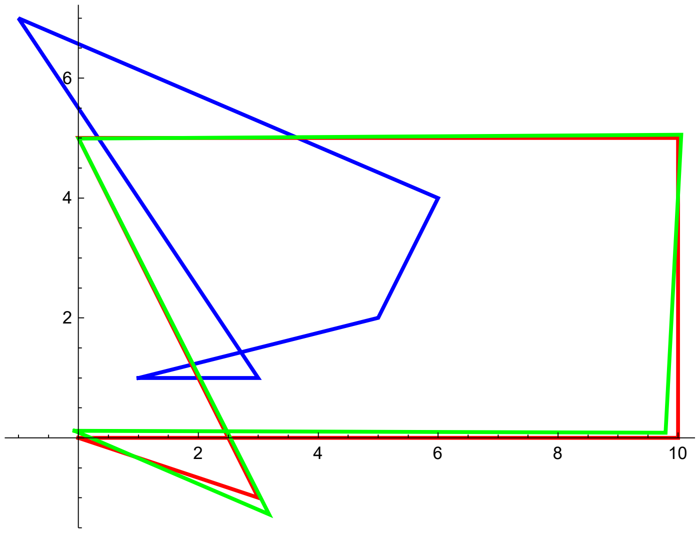
In[37]:=
```
orig = ShowPolygon[{x1, x2, x3, x4, x5, x1}, Blue];
```

In[38]:=
```
zadataSlika = ShowPolygon[{x1p, x2p, x3p, x4p, x5p, x1p}, Red];
```

In[39]:=
```
dobijenaSlika = ShowPolygon[(Adlp5.#) & /@ {x1, x2, x3, x4, x5, x1}, Green];
```

In[40]:=
```
Show[orig, zadataSlika, dobijenaSlika, Axes → True]
(* plavo=originali, crveno= slika, zeleno = dobijenaSlika *)
```

Out[40]=



DLP algoritam nije oseteljiv na permutaciju odgovarajucih tacaka - ispod menjamo x2 i x3

In[41]:=
```
MatrixForm /@ {ProjectiveDLP[{x1, x2, x3, x4, x5}, {x1p, x2p, x3p, x4p, x5p}],
  ProjectiveDLP[{x1, x3, x2, x4, x5}, {x1p, x3p, x2p, x4p, x5p}]}
```

Out[41]=

$$\left\{\begin{pmatrix} 0.343498 & 0.118237 & -0.48379 \\ -0.152467 & 0.585069 & -0.395034 \\ -0.0573214 & 0.0555633 & 0.325575 \end{pmatrix}, \begin{pmatrix} 0.343498 & 0.118237 & -0.48379 \\ -0.152467 & 0.585069 & -0.395034 \\ -0.0573214 & 0.0555633 & 0.325575 \end{pmatrix}\right\}$$

3) DLP algoritam nije invarijantan na promenu koordinata

Pretpostavimo da smo izabrali drugacije koordinate istih tacaka, u originalu i u slici. Recimo postavili koordinatni pocetak u centar slike, mesto u donji gornji levi ugao, ili da smo zarotirali koordinatni sistem. Ocekujemo da cemo dobiti isto preslikavanje, ali na zalost, necemo !!!

In[61]:=
```
C1 = {{0, 1, 2}, {-1, 0, 3}, {0, 0, 1}};
(* matrica transformacije koordinata originala *)
```

In[62]:=
```
C2 = {{1, -1, 5}, {1, 1, -2}, {0, 0, 1}};
(* matrica transformacije koordinata slika *)
```

In[63]:=
```
Adlp = ProjectiveDLP[{x1, x2, x3, x4, x5}, {x1p, x2p, x3p, x4p, x5p}];
```

In[64]:=
```
{nx1, nx2, nx3, nx4, nx5} = Map[(C1.#) &, {x1, x2, x3, x4, x5}];
(* nove koordinate originala i slika *)
{nx1p, nx2p, nx3p, nx4p, nx5p} = Map[(C2.#) &, {x1p, x2p, x3p, x4p, x5p}];
```

In[66]:=
```
nAdlp = ProjectiveDLP[{nx1, nx2, nx3, nx4, nx5}, {nx1p, nx2p, nx3p, nx4p, nx5p}];
(* DLP algoritam primenjen na nove koordinate *)
```

Proveravamo da li je rezultat DLP algoritma primenjenog na nove koordinate, isti kao rezultat starog u novim koordinatama

In[67]:=
```
AdlpStari = Inverse[C2].nAdlp.C1
```

Out[67]=
```
{{-0.106031, -0.0360324, 0.14643},
 {0.0478924, -0.182046, 0.121891}, {0.0179828, -0.0171818, -0.102178}}
```

In[68]:=
```
MatrixForm /@ {nAdlp, AdlpStari}
```

Out[68]=

$$\left\{\begin{pmatrix} 0.0601042 & 0.0640095 & -0.798587 \\ -0.183715 & 0.0941045 & 0.557792 \\ -0.0171818 & -0.0179828 & -0.0138657 \end{pmatrix}, \begin{pmatrix} -0.106031 & -0.0360324 & 0.14643 \\ 0.0478924 & -0.182046 & 0.121891 \\ 0.0179828 & -0.0171818 & -0.102178 \end{pmatrix}\right\}$$

Probamo da reskaliramo matrice

In[69]:=
```
MatrixForm /@ {Adlp, (Adlp[[1]] / AdlpStari[[1]]) AdlpStari}
```

Out[69]=

$$\left\{\begin{pmatrix} 0.343498 & 0.118237 & -0.48379 \\ -0.152467 & 0.585069 & -0.395034 \\ -0.0573214 & 0.0555633 & 0.325575 \end{pmatrix}, \begin{pmatrix} 0.343498 & 0.11673 & -0.474372 \\ -0.157154 & 0.597365 & -0.399972 \\ -0.0594137 & 0.0567672 & 0.337586 \end{pmatrix}\right\}$$

VIdimo da se na drugoj decimali pojavljuje razlika. Nije strasno, ali DLP algoritam nije invarijantan u odnosu na promenu koordinata, tj. nije geometrijski!!!

4) Modifikovani DLP algoritam - poredjenje sa DLP algoritmom i invarijantnost u odnosu na tranformaciju koordinata

In[70]:= **Kako radi normalizacija? Graficki prikaz.**

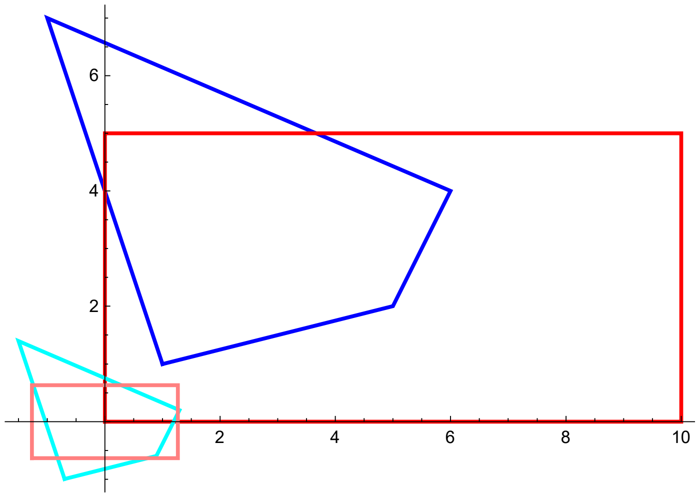
In[70]:=
```
orig = ShowPolygon[{x1, x2, x3, x4, x1}, Blue];
```

In[71]:=
```
slika = ShowPolygon[{x1p, x2p, x3p, x4p,  x1p}, Red];
```

In[72]:=
```
normOrig =
  ShowPolygon[(normMatrix[{x1, x2, x3, x4}].#) & /@ {x1, x2, x3, x4, x1}, Cyan];
normSlika = ShowPolygon[(normMatrix[{x1p, x2p, x3p, x4p}].#) & /@
    {x1p, x2p, x3p, x4p, x1p}, Pink];
```

In[74]:=
```
Show[orig, slika, normOrig, normSlika, Axes → True]
```

Out[74]=



In[75]:=
```
x1 = {1, 1, 1}; x2 = {5, 2, 1}; x3 = {6, 4, 1}; x4 = {-1, 7, 1};  (* originali *)
```

In[76]:=
```
x1p = {0, 0, 1};
x2p = {10, 0, 1}; x3p =  {10, 5, 1}; x4p = {0, 5, 1};  (* slike *)
```

In[85]:=
```
Adlp = ProjectiveDLP[{x1, x2, x3, x4, x5}, {x1p, x2p, x3p, x4p, x5p}];
```

In[84]:=
```
AdlpNorm = ProjectiveDLPNorm[{x1, x2, x3, x4, x5}, {x1p, x2p, x3p, x4p, x5p}];
```

```
MatrixForm /@ {Adlp, (Adlp[[1]] / AdlpNorm[[1]]) AdlpNorm}
 (* skaliramo da uporedimo rezultate *)
```

Out[81]=

$$\left\{\begin{pmatrix} 0.343498 & 0.118237 & -0.48379 \\ -0.152467 & 0.585069 & -0.395034 \\ -0.0573214 & 0.0555633 & 0.325575 \end{pmatrix}, \begin{pmatrix} 0.343498 & 0.117186 & -0.477321 \\ -0.15736 & 0.595448 & -0.39459 \\ -0.059134 & 0.0567652 & 0.33472 \end{pmatrix}\right\}$$

Vidimo da se rezultati dobijeni DLP i modifikovanim DLP alogritmom u ovom primeru razlikuju na drugoj decimali.

Ipak, modifikovani DLP je numericki stabilniji i NE ZAVISI OD IZBORA KOORDINATA.

Hajde da promenimo koordinatni sistem kao u prethodnom primeru da vidimo sta se desava

```
In[•]:= C1 = {{0, 1, 2}, {-1, 0, 3}, {0, 0, 1}};
(* matrica transformacije koordinata originala *)
```

```
In[•]:= C2 = {{1, -1, 5}, {1, 1, -2}, {0, 0, 1}};
(* matrica transformacije koordinata slika *)
```

```
In[•]:= {nx1, nx2, nx3, nx4, nx5} = Map[(C1.#) &, {x1, x2, x3, x4, x5}];
(* nove koordinate originala i slika *)
{nx1p, nx2p, nx3p, nx4p, nx5p} = Map[(C2.#) &, {x1p, x2p, x3p, x4p, x5p}];
```

```
In[90]:= nAdlpNorm =
  ProjectiveDLPNorm[{nx1, nx2, nx3, nx4, nx5}, {nx1p, nx2p, nx3p, nx4p, nx5p}] ;
 (* Normalizovani DLP algoritam primenjen na nove koordinate *)
```

Proveravamo da li je rezultat modifikovanog DLP algoritma primenjenog na nove koordinate, isti kao rezultat starog u novim koordinatama

```
AdlpStariNorm = Inverse[C2].nAdlpNorm.C1; (* Vratimo u stari koordinatni sistem*)
```

```
In[91]:= MatrixForm /@ {AdlpNorm, AdlpStariNorm}
```

Out[91]=

$$\left\{\begin{pmatrix} -0.519471 & -0.17722 & 0.721852 \\ 0.23586 & -0.89249 & 0.591432 \\ 0.0882325 & -0.0846979 & -0.499427 \end{pmatrix}, \begin{pmatrix} 0.519471 & 0.17722 & -0.721852 \\ -0.23586 & 0.89249 & -0.591432 \\ -0.0882325 & 0.0846979 & 0.499427 \end{pmatrix}\right\}$$

Ovaj put su matrice iste, tj. modifikovani DLP ne zavisi od izbora koordinata