

## 1.13 Klasifikace kvadrik - řešené příklady

**Příklad 4:** Vyšetřete kvadriku  $8x^2 - 8y^2 - 3z^2 - 12xy + 10xz + 10yz - 2x + 14y - 10z - 3 = 0$

```
[ > restart;
[ > with(LinearAlgebra):
[ > with(linalg):
[ > with(plots):
[ > X:=Vector[row]([x,y,z,1]);
                                X := [x, y, z, 1]
[ > K:=Matrix(a,1..4,1..4,shape=symmetric);
                                K :=  $\begin{bmatrix} a(1,1) & a(1,2) & a(1,3) & a(1,4) \\ a(1,2) & a(2,2) & a(2,3) & a(2,4) \\ a(1,3) & a(2,3) & a(3,3) & a(3,4) \\ a(1,4) & a(2,4) & a(3,4) & a(4,4) \end{bmatrix}$ 
[ > Kv:=sort(expand(X.K.Transpose(X)),[x,y,z])=0;
Kv :=  $a(1,1)x^2 + 2a(1,2)xy + 2a(1,3)xz + a(2,2)y^2 + 2a(2,3)yz + a(3,3)z^2$ 
       $+ 2a(1,4)x + 2a(2,4)y + 2a(3,4)z + a(4,4) = 0$ 
[ > Tecna:=[x=m+t*u,y=n+t*v,z=p+t*w];
                                Tecna := [x = m + t u, y = n + t v, z = p + t w]
[ > Kv0:=simplify(eval(Kv,[x=m,y=n,z=p]));
Kv0 :=  $a(1,1)m^2 + 2a(1,2)mn + 2a(1,3)mp + a(2,2)n^2 + 2a(2,3)np + a(3,3)p^2$ 
       $+ 2a(1,4)m + 2a(2,4)n + 2a(3,4)p + a(4,4) = 0$ 
[ > Kv1:=simplify(eval(Kv,Tecna));
Kv1 :=  $2a(1,2)mn + 2a(1,3)mp + 2a(2,3)np + a(1,1)m^2 + a(2,2)n^2 + a(3,3)p^2$ 
       $+ 2a(1,4)m + 2a(2,4)n + 2a(3,4)p + a(4,4) + a(1,1)t^2u^2 + a(2,2)t^2v^2 + a(3,3)t^2w^2$ 
       $+ 2a(1,4)tu + 2a(2,4)tv + 2a(3,4)tw + 2a(1,1)mtu + 2a(1,2)mtv$ 
       $+ 2a(1,2)tun + 2a(1,2)t^2uv + 2a(1,3)mtw + 2a(1,3)tup + 2a(1,3)t^2uw$ 
       $+ 2a(2,2)ntv + 2a(2,3)ntw + 2a(2,3)tvp + 2a(2,3)t^2vw + 2a(3,3)ptw = 0$ 
[ > A:=coeff(lhs(Kv1),t^2);
      A :=  $a(1,1)u^2 + a(2,2)v^2 + a(3,3)w^2 + 2a(1,2)uv + 2a(1,3)uw + 2a(2,3)vw$ 
[ > B:=1/2*coeff(lhs(Kv1),t);
      B :=  $a(1,4)u + a(2,4)v + a(3,4)w + a(1,1)mu + a(1,2)mv + a(1,2)un + a(1,3)mw$ 
       $+ a(1,3)up + a(2,2)nv + a(2,3)nw + a(2,3)vp + a(3,3)pw$ 
[ > C:=sort(coeff(lhs(Kv1),t,0),[r,s,u]);
      C :=  $a(1,1)m^2 + 2a(1,2)mn + 2a(1,3)mp + a(2,2)n^2 + 2a(2,3)np + a(3,3)p^2$ 
       $+ 2a(1,4)m + 2a(2,4)n + 2a(3,4)p + a(4,4)$ 
[ > B1:=collect(B,[u,v,w]);
      B1 :=  $(a(1,1)m + a(1,4) + a(1,2)n + a(1,3)p)u$ 
       $+ (a(1,2)m + a(2,2)n + a(2,4) + a(2,3)p)v$ 
       $+ (a(2,3)n + a(3,4) + a(3,3)p + a(1,3)m)w$ 
```

```

> a1:=sort(coeff(B1,u),[m,n,p]); b1:=sort(coeff(B1,v),[m,n,p]);
c1:=sort(coeff(B1,w),[m,n,p]);
d1:=sort(coeff(coeff(coeff(B1,u,0),v,0),w,0),[m,n,p]);
      a1 := a(1,1)m + a(1,2)n + a(1,3)p + a(1,4)
      b1 := a(1,2)m + a(2,2)n + a(2,3)p + a(2,4)
      c1 := a(1,3)m + a(2,3)n + a(3,3)p + a(3,4)
      d1 := 0
> B2:=collect(B,[m,n,p]);
B2 := (a(1,2)v + a(1,3)w + a(1,1)u)m + (a(2,2)v + a(1,2)u + a(2,3)w)n
      + (a(1,3)u + a(2,3)v + a(3,3)w)p + a(1,4)u + a(2,4)v + a(3,4)w
> ra:=sort(coeff(B2,m),[u,v,w]); rb:=sort(coeff(B2,n),[u,v,w]);
rc:=sort(coeff(B2,p),[u,v,w]);
rd:=sort(coeff(coeff(coeff(B2,m,0),n,0),p,0),[u,v,w]);
      ra := a(1,1)u + a(1,2)v + a(1,3)w
      rb := a(1,2)u + a(2,2)v + a(2,3)w
      rc := a(1,3)u + a(2,3)v + a(3,3)w
      rd := a(1,4)u + a(2,4)v + a(3,4)w
> U:=Vector[row]([u,v,w,0]);
      U := [u, v, w, 0]
> PrumerR:=collect(expand(evalm(U*K&K*Transpose(X))),[x,y,z])=0;
PrumerR := (a(1,1)u + a(1,2)v + a(1,3)w)x + (a(1,2)u + a(2,2)v + a(2,3)w)y
      + (a(1,3)u + a(2,3)v + a(3,3)w)z + a(1,4)u + a(2,4)v + a(3,4)w = 0
> n:=[coeff(lhs(PrumerR),x),coeff(lhs(PrumerR),y),coeff(lhs(PrumerR),z)];
n := [a(1,1)u + a(1,2)v + a(1,3)w, a(1,2)u + a(2,2)v + a(2,3)w,
      a(1,3)u + a(2,3)v + a(3,3)w]

```

Dosadíme hodnoty parametrů podle zadání:

```

> kv:=8*x^2-8*y^2-3*z^2-12*x*y+10*x*z+10*y*z-2*x+14*y-10*z-3=0;
      kv := 8x2 - 8y2 - 3z2 - 12xy + 10xz + 10yz - 2x + 14y - 10z - 3 = 0
> a(1,1):=coeff(lhs(kv),x^2); a(2,2):=coeff(lhs(kv),y^2);
a(3,3):=coeff(lhs(kv),z^2);
a(1,2):=1/2*coeff(coeff(lhs(kv),x),y);
a(1,3):=1/2*coeff(coeff(lhs(kv),x),z);
a(1,4):=1/2*coeff(coeff(coeff(lhs(kv),x,1),y,0),z,0);
a(2,3):=1/2*coeff(coeff(lhs(kv),y,1),z);
a(2,4):=1/2*coeff(coeff(coeff(lhs(kv),y,1),x,0),z,0);
a(3,4):=1/2*coeff(coeff(coeff(lhs(kv),z,1),y,0),x,0);
a(4,4):=coeff(coeff(coeff(lhs(kv),x,0),y,0),z,0);
      a(1,1) := 8
      a(2,2) := -8

```

```
a(3,3) := -3
a(1,2) := -6
a(1,3) := 5
a(1,4) := -1
a(2,3) := 5
a(2,4) := 7
a(3,4) := -5
a(4,4) := -3
```

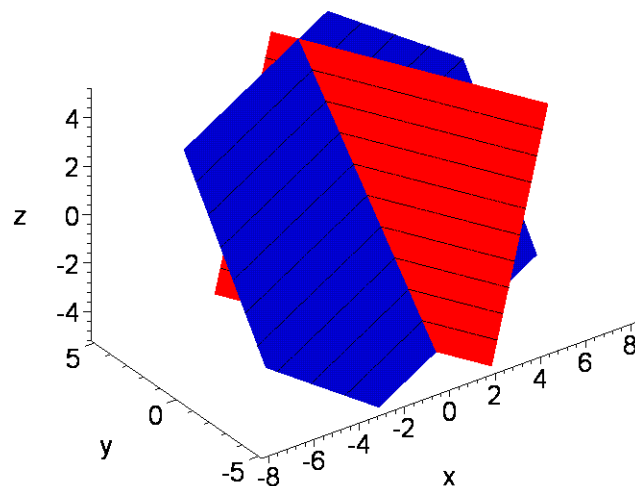
```
> K;
```

$$\begin{bmatrix} 8 & -6 & 5 & -1 \\ -6 & -8 & 5 & 7 \\ 5 & 5 & -3 & -5 \\ -1 & 7 & -5 & -3 \end{bmatrix}$$

```
> factor(kv);
```

$$(2x + 1 - 4y + 3z)(4x - 3 + 2y - z) = 0$$

```
> kvg:=plots[implicitplot3d]([op(factor(lhs(kv)))],x=-8..8,y=-5..5
,z=-5..5,orientation=[125,60],gridstyle=rectangular,axes=frame,g
rid=[40,40,40],style=patchcontour,color=[blue,red],light=[90,-5,
1,1,1]):
> display(kvg,axes=frame,scaling=constrained,orientation=[-126,60]
);
```



```
> n;
```

```

[      [8 u - 6 v + 5 w, -6 u - 8 v + 5 w, 5 u + 5 v - 3 w]
> uv := [u, v, w];
[      uv := [u, v, w]
> r := evalm(n-lambda*uv);
[      r := [8 u - 6 v + 5 w - lambda u, -6 u - 8 v + 5 w - lambda v, 5 u + 5 v - 3 w - lambda w]
> ChM := linalg[genmatrix]([r[1], r[2], r[3]], [u, v, w]);
[      ChM := [ [8 - lambda   -6     5
                -6   -8 - lambda  5
                5     5   -3 - lambda] ]
> ChR := linalg[det](ChM)=0;
[      ChR := 150 lambda - 3 lambda^2 - lambda^3 = 0
> collect(ChR, [lambda^3, lambda^2, lambda]);
[      150 lambda - 3 lambda^2 - lambda^3 = 0
> I1 := coeff(lhs(ChR), lambda, 2);
[      I1 := -3
> I2 := coeff(lhs(ChR), lambda, 1);
[      I2 := 150
> A44 := coeff(lhs(ChR), lambda, 0);
[      A44 := 0
> Delta := linalg[det](K);
[      Delta := 0
> lambda_sol := solve(ChR, {lambda});
[      lambda_sol := {lambda = 0}, {lambda = -3/2 + sqrt(609)/2}, {lambda = -3/2 - sqrt(609)/2}
> r1 := eval(r, lambda_sol[1]); r2 := eval(r, lambda_sol[2]);
[      r1 := [8 u - 6 v + 5 w, -6 u - 8 v + 5 w, 5 u + 5 v - 3 w]
[      r2 := [ 8 u - 6 v + 5 w - (-3/2 + sqrt(609)/2) u, -6 u - 8 v + 5 w - (-3/2 + sqrt(609)/2) v,
[      5 u + 5 v - 3 w - (-3/2 + sqrt(609)/2) w ]
[      r3 := [ 8 u - 6 v + 5 w - (-3/2 - sqrt(609)/2) u, -6 u - 8 v + 5 w - (-3/2 - sqrt(609)/2) v,
[      5 u + 5 v - 3 w - (-3/2 - sqrt(609)/2) w ]
> hs1 := eval(uv, solve({r1[1], r1[2], r1[3]}, {u, v, w}));
[      hs2 := eval(uv, solve({r2[1], r2[2], r2[3]}, {u, v, w}));
[      hs3 := eval(uv, solve({r3[1], r3[2], r3[3]}, {u, v, w}));
[      hs1 := [u, -7 u, -10 u]
[      hs2 := [ 121/80 w + 7/80 w sqrt(609), -97/80 w + 1/80 w sqrt(609), w ]

```

$$hs3 := \left[ \frac{121}{80} w - \frac{7}{80} w \sqrt{609}, -\frac{97}{80} w - \frac{1}{80} w \sqrt{609}, w \right]$$

> PrumerR;

$$(8u - 6v + 5w)x + (-6u - 8v + 5w)y + (5u + 5v - 3w)z - u + 7v - 5w = 0$$

> Smer1:=eval(hs1, {u=1, v=1, w=1});

$$Smer1 := [1, -7, -10]$$

> PrumerR1:=eval(PrumerR, [u=Smer1[1], v=Smer1[2], w=Smer1[3]]);

$$PrumerR1 := 0 = 0$$

> Smer2:=eval(hs2, {u=1, v=1, w=1});

$$Smer2 := \left[ \frac{121}{80} + \frac{7\sqrt{609}}{80}, -\frac{97}{80} + \frac{\sqrt{609}}{80}, 1 \right]$$

> PrumerR2:=eval(PrumerR, [u=Smer2[1], v=Smer2[2], w=Smer2[3]]);

$$PrumerR2 := \left( \frac{195}{8} + \frac{5\sqrt{609}}{8} \right) x + \left( \frac{45}{8} - \frac{5\sqrt{609}}{8} \right) y + \left( -\frac{3}{2} + \frac{\sqrt{609}}{2} \right) z - 15 = 0$$

> Smer3:=eval(hs3, {u=1, v=1, w=1});

$$Smer3 := \left[ \frac{121}{80} - \frac{7\sqrt{609}}{80}, -\frac{97}{80} - \frac{\sqrt{609}}{80}, 1 \right]$$

> PrumerR3:=eval(PrumerR, [u=Smer3[1], v=Smer3[2], w=Smer3[3]]);

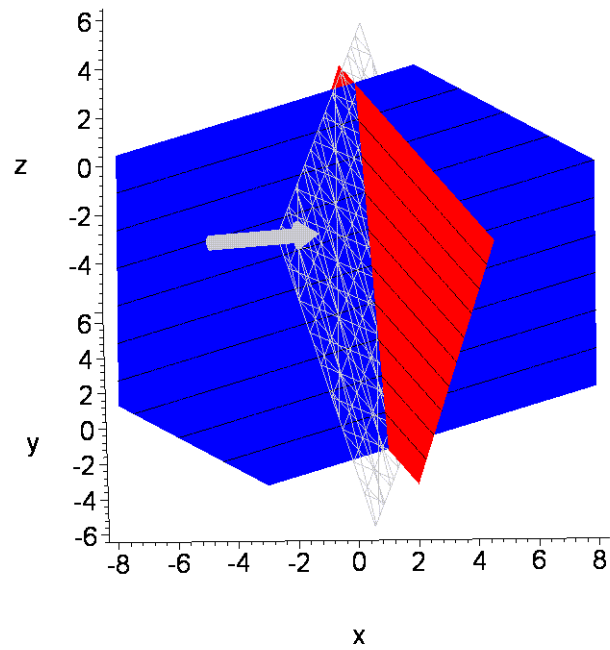
$$PrumerR3 := \left( \frac{195}{8} - \frac{5\sqrt{609}}{8} \right) x + \left( \frac{45}{8} + \frac{5\sqrt{609}}{8} \right) y + \left( -\frac{3}{2} - \frac{\sqrt{609}}{2} \right) z - 15 = 0$$

> col2:=grey;

> PrumerR2g:=implicitplot3d(PrumerR2, x=-6..6, y=-6..6, z=-6..6, grid=[10,10,10], style=wireframe, color=col2):

> Smer2g:=plottools[arrow]([-5, -5, 5], vector(Smer2), 0.5, 1, 0.2, cylindrical\_arrow, color=col2):

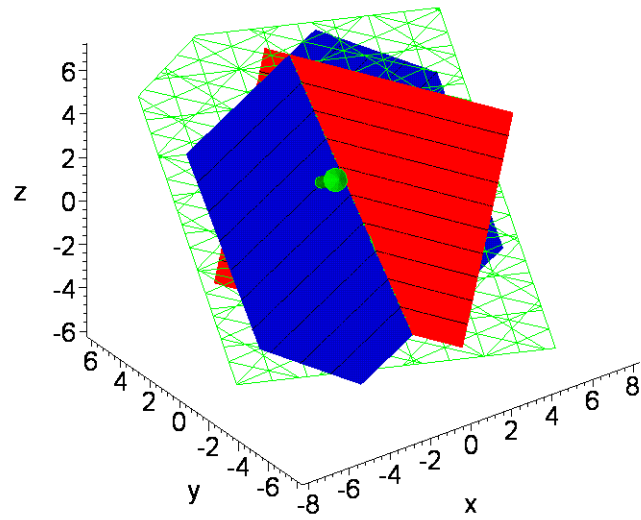
> display(kvg, PrumerR2g, Smer2g, axes=frame, scaling=constrained, orientation=[-126, 60]);



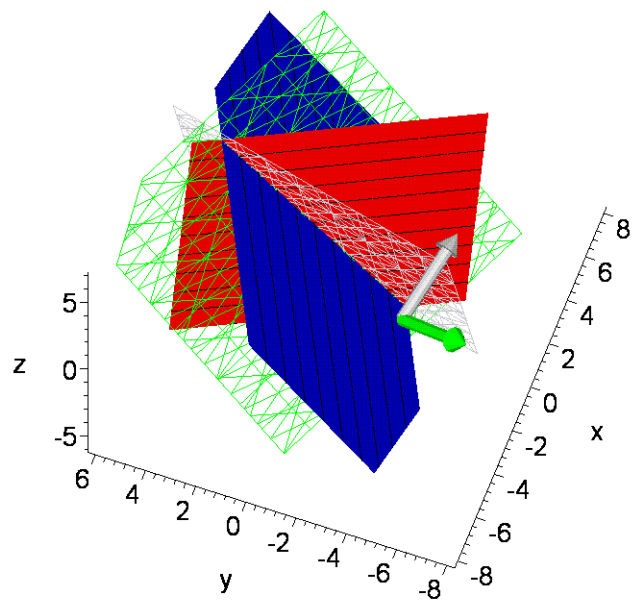
```

> col3:=green:
> PrumerR3g:=implicitplot3d(PrumerR3,x=-6..6,y=-6..6,z=-6..6,grid=
[10,10,10],style=wireframe,color=col3):
> Smer3g:=plottools[arrow]([-5, -5, 5], vector(2*Smer3), 0.5, 1,
0.2, cylindrical_arrow,color=col3):
> display(kvg,PrumerR3g,Smer3g,axes=frame,scaling=constrained,orie
ntation=[-126,60]);

```



```
> display(kvg,PrumerR2g,Smer2g,PrumerR3g,Smer3g,axes=frame,scaling
=constrained,orientation=[-160,30]);
```



```
[ >
```