5. přednáška (Lagrangeův interpolační polynom, Bézierovy křivky)

Lagrangeův interpolační polynom

Příklad:

Najděte lineární polynom $\mathrm{p}(x) = a\,x + b$, jehož graf prochází body $[1, 2]$ a $[7, 5]$.

```
> restart;
> p:=x->a*x+b;
```

$$p := x \to a\,x + b$$

```
> p(1)=2;
```

$$a + b = 2$$

```
> p(7)=5;
```

$$7\,a + b = 5$$

```
> solve({p(1)=2,p(7)=5},{a,b});
```

$$\{a = \frac{1}{2}, b = \frac{3}{2}\}$$

```
> assign(%);
> a;
```

$$\frac{1}{2}$$

```
> b;
```

$$\frac{3}{2}$$

```
> p(x);
```

$$\frac{x}{2} + \frac{3}{2}$$

Příklad:

Najděte polynom nejvýše druhého stupně takový, aby jeho graf procházel body $[-2, 9]$, $[-1, 0]$ a $[3, 4]$.

```
> restart;
> p:=x->a*x^2+b*x+c;
```

$$p := x \to a\,x^2 + b\,x + c$$

```
> solve({p(-2)=9,p(-1)=0,p(3)=4},{a,b,c});
```

$$\{a = 2, b = -3, c = -5\}$$

```
> assign(%);
```

```
> p(x);
```

$$2x^2 - 3x - 5$$

Příklad:

Najděte polynom nejvýše 3. stupně, aby v bodech $-1, 2$ a 4 byl roven nule a v bodě 3 byl roven jedné.

```
> restart;
> p:=x->a*x^3+b*x^2+c*x+d;
```

$$p := x \rightarrow a x^3 + b x^2 + c x + d$$

```
> solve({p(-1)=0,p(2)=0,p(4)=0,p(3)=1},{a,b,c,d});
```

$$\{a = \frac{-1}{4}, b = \frac{5}{4}, c = \frac{-1}{2}, d = -2\}$$

```
> assign(%);
> p(x);
```

$$-\frac{1}{4}x^3 + \frac{5}{4}x^2 - \frac{1}{2}x - 2$$

Rozložíme polynom p na kořenové činitele:

```
> factor(p(x));
```

$$-\frac{(x-2)(x-4)(x+1)}{4}$$

Nyní vyřešme všechny úlohy s pomocí příkazu PolynomialInterpolation z balíku CurveFitting.

```
> with(CurveFitting);
```

$[ArrayInterpolation, BSpline, BSplineCurve, Interactive, LeastSquares,$
$PolynomialInterpolation, RationalInterpolation, Spline, ThieleInterpolation]$

Příklad:

Najděte lineární polynom $\mathrm{p}(x) = a\,x + b$, jehož graf prochází body $[1, 2]$ a $[7, 5]$.

```
> body:=[[1,2],[7,5]];
```

$$body := [[1, 2], [7, 5]]$$

```
> PolynomialInterpolation(body,x);
```

$$\frac{x}{2} + \frac{3}{2}$$

Příklad:

Najděte polynom nejvýše druhého stupně takový, aby jeho graf procházel body $[-2, 9]$, $[-1, 0]$ a $[3, 4]$.

```
> PolynomialInterpolation([[-2,9],[-1,0],[3,4]],x);
```

$$2x^2 - 3x - 5$$

```
> PolynomialInterpolation([[-2,9],[-1,0],[3,4]],x,form=Lagrange
  );
```

$$\frac{9(x+1)(x-3)}{5} + \frac{(x+2)(x+1)}{5}$$

Další povolená syntaxe:

```
> PolynomialInterpolation([-2,-1,3],[9,0,4],x);
```

$$2x^2 - 3x - 5$$

Příklad:

Najděte polynom nejvýše 3. stupně, aby v bodech $-1, 2$ a 4 byl roven nule a v bodě 3 byl roven jedné.

```
> PolynomialInterpolation([-1,2,4,3],[0,0,0,1],x,form=Lagrange)
  ;
```

$$-\frac{(x+1)(x-2)(x-4)}{4}$$

Ještě jinak:

```
> Interactive();
>
```

Aproximace funkce $f(x) = |x|$ na intervalu $<-1, 1>$ Lagrangeovým interpolačním polynomem (interval $<-1, 1>$ přitom rozdělíme na 100 podintervalů):

```
> restart;
> with(CurveFitting):
> poly:=x->PolynomialInterpolation([seq([i,abs(i)],i=-1..1,1/50
  )],x):
```

Hodnota interpolačního polynomu v polovině posledního dílku:

```
> poly(0.99);
```

$$-0.5075224627\ 10^{32}$$

```
> poly(99/100);
```

3640269951810318113713099392751536876042560580968573220689511208909718 60\

322985815921 /
627710173538668076383578942320766641610235544446403451289600

```
> evalf(%);
```

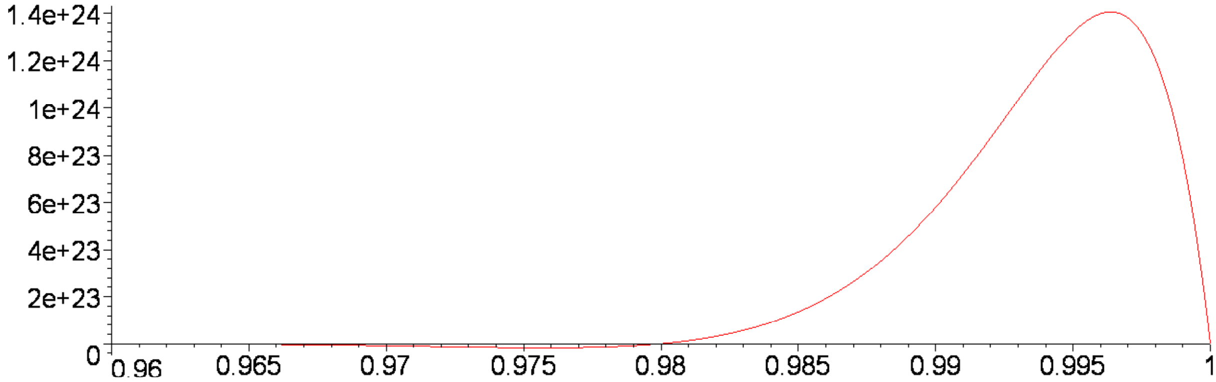
$$0.5799284615\ 10^{24}$$

(Tato hodnota je důvěryhodná.)

```
> Digits:=100;
```

$$Digits := 100$$

```
> plot(poly,0.96..1);
```




```
>
```

Bézierovy křivky

```
> restart;
> with(plots):
> with(plottools):
```

Lineární Bézierova křivka:
(je dána dvěma řídícími body)

```
> B[0]:=[1,2]; B[1]:=[7,4];
```

$$B_0 := [1, 2]$$

$$B_1 := [7, 4]$$

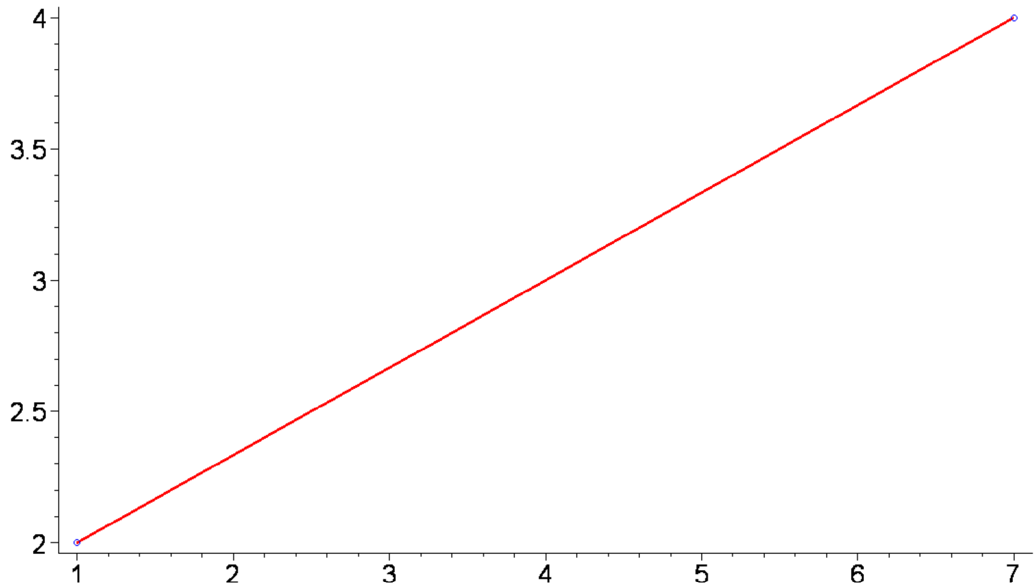
```
> bezier1:=(1-t)*B[0]+t*B[1];
```

$$bezier1 := (1 - t)\,[1, 2] + t\,[7, 4]$$

```
> bezier1:=expand(bezier1);
```

$$bezier1 := [6\,t + 1, 2\,t + 2]$$

```
> body:=pointplot([B[0],B[1]],symbolsize=15,symbol=circle,color
=blue):
> krivka:=plot([op(bezier1),t=0..1],thickness=3):
> display([body,krivka]);
```



Lineární Bézierova křivka je tedy vždy úsečka spojující řídící body.

Kvadratická Bézierova křivka:
(je dána třemi řídícími body)

```
> B[0]:=[-1,2]; B[1]:=[0,0]; B[2]:=[2,4];
```

$$B_0 := [-1, 2]$$

$$B_1 := [0, 0]$$

$$B_2 := [2, 4]$$

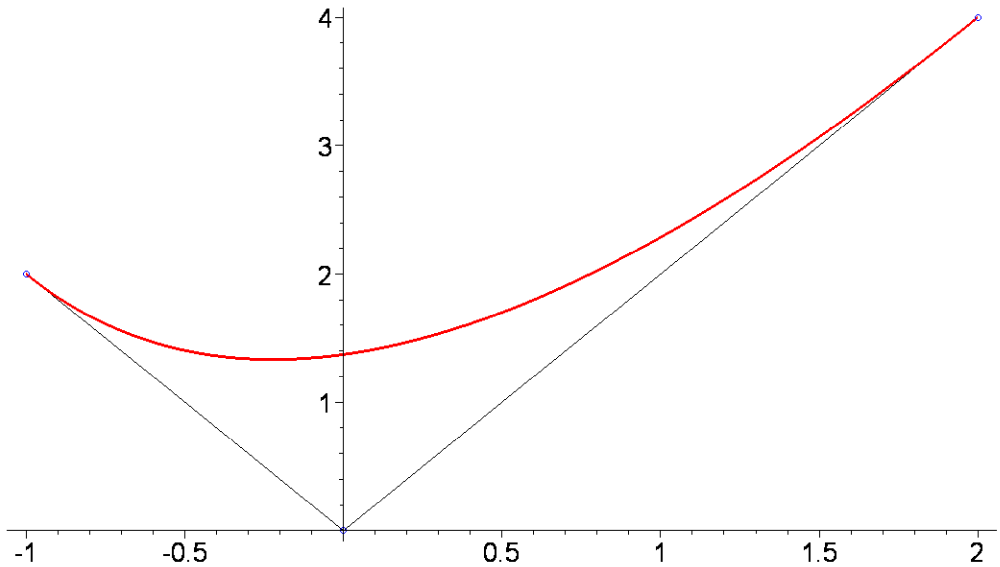
```
> bezier2:=(1-t)^2*B[0]+2*(1-t)*t*B[1]+t^2*B[2];
```

$$bezier2 := (1-t)^2\,[-1, 2] + 2\,(1-t)\,t\,[0, 0] + t^2\,[2, 4]$$

```
> bezier2:=expand(bezier2);
```

$$bezier2 := [t^2 - 1 + 2\,t, 6\,t^2 + 2 - 4\,t]$$

```
> body:=pointplot([B[0],B[1],B[2]],symbolsize=15,symbol=circle,
  color=blue):
> krivka:=plot([op(bezier2),t=0..1],thickness=3):
> polyg:=seq(line(B[i],B[i+1],color=black),i=0..1):
> display([body,krivka,polyg]);
```



Pozor! Záleží na pořadí řídících bodů.

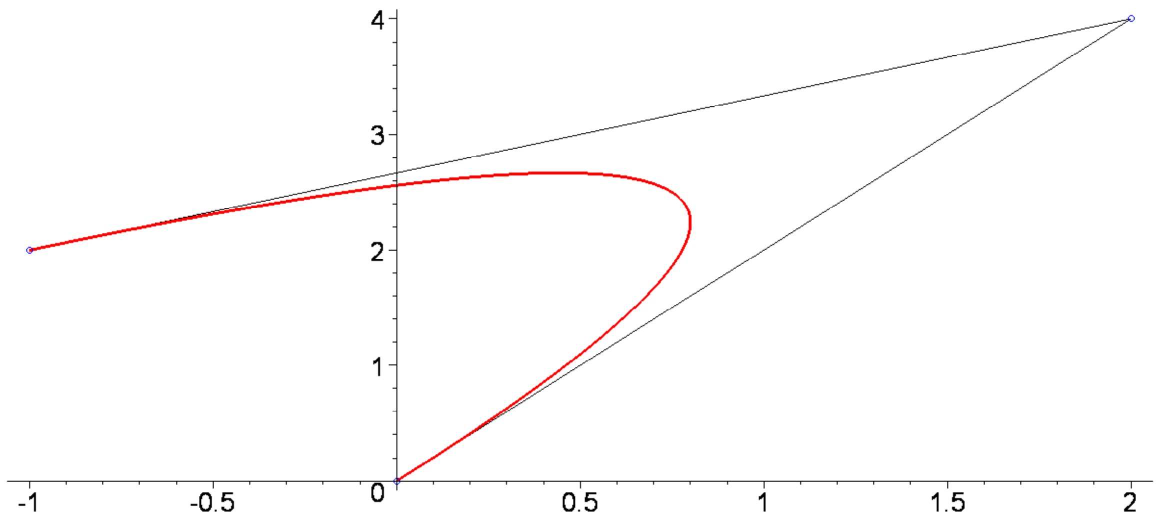
```
> bezier2:=(1-t)^2*B[0]+2*(1-t)*t*B[2]+t^2*B[1];
```

$$bezier2 := (1-t)^2\,[-1, 2] + 2\,(1-t)\,t\,[2, 4] + t^2\,[0, 0]$$

```
> bezier2:=expand(bezier2);
```

$$bezier2 := [6\,t - 5\,t^2 - 1, 4\,t - 6\,t^2 + 2]$$

```
> krivka:=plot([op(bezier2),t=0..1],thickness=3):
> polyg:=line(B[0],B[2],color=black),line(B[2],B[1],color=black
  ):
> display([body,krivka,polyg]);
```



Kvadratická Bézierova křivka nemusí být parabola.

Kubická Bézierova křivka:
(je dána čtyřmi řídícími body)

```
> B[0]:=[0,0]; B[1]:=[2,1]; B[2]:=[-1,4]; B[3]:=[2,5];
```

$$B_0 := [0, 0]$$

$$B_1 := [2, 1]$$

$$B_2 := [-1, 4]$$

$$B_3 := [2, 5]$$

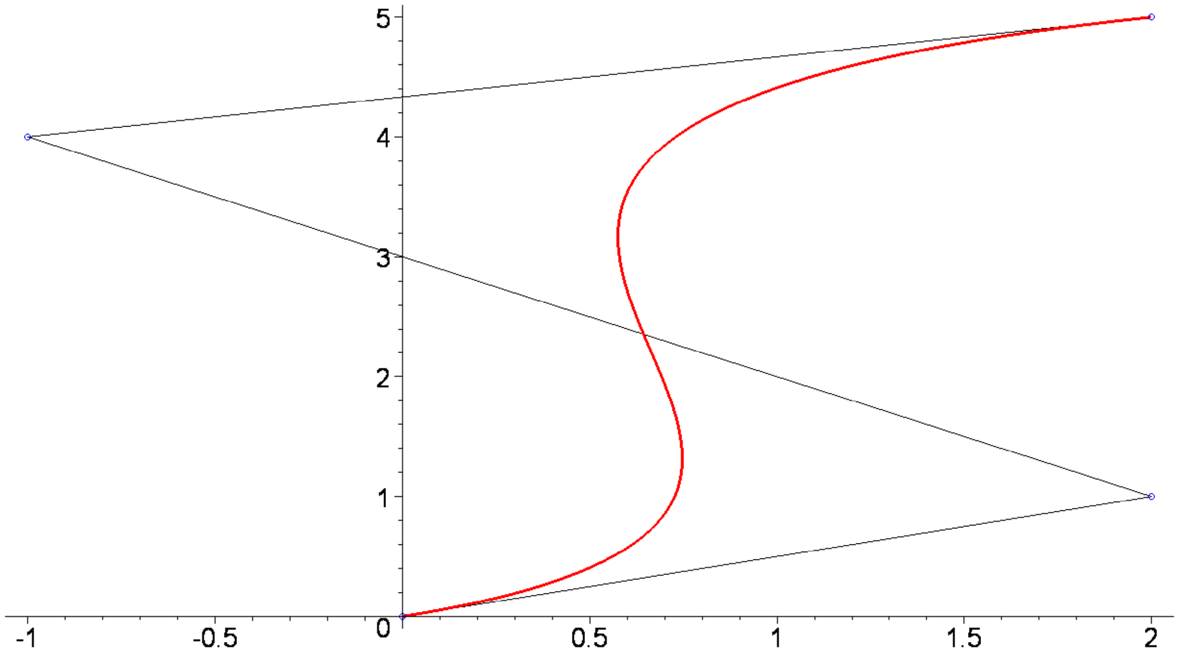
```
> bezier3:=sum(binomial(3,i)*(1-t)^(3-i)*t^i*B[i],i=0..3);
```

$$bezier3 := (1-t)^3 [0, 0] + 3 (1-t)^2 t [2, 1] + 3 (1-t) t^2 [-1, 4] + t^3 [2, 5]$$

```
> bezier3:=expand(bezier3);
```

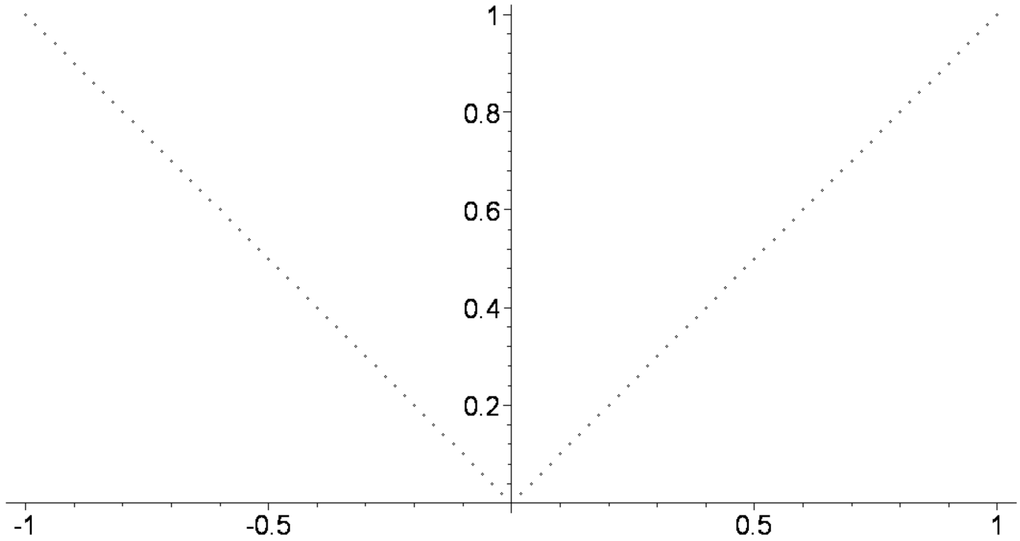
$$bezier3 := [11 t^3 - 15 t^2 + 6 t, -4 t^3 + 6 t^2 + 3 t]$$

```
> krivka:=plot([op(bezier3),t=0..1],thickness=3):
> body:=pointplot([seq(B[i],i=0..3)],symbolsize=15,symbol=circl
  e,color=blue):
> polyg:=seq(line(B[i],B[i+1],color=black),i=0..2):
> display([body,krivka,polyg]);
```



Aproximace funkce f$(x) = |x|$ (definované na intervalu <–1, 1>) Bézierovou křivkou 100. řádu:

```
> B:=seq([i,abs(i)],i=-1..1,1/50):
> body:=pointplot([B],symbolsize=5):
> display(body,scaling=constrained);
```

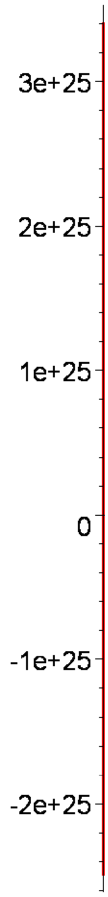



```
> bezier:=sum(binomial(100,i)*(1-t)^(100-i)*t^i*B[i+1],i=0..100
  ):
> bezier:=expand(bezier);
```

$$
\begin{aligned}
\mathit{bezier} := [&-1 + 2\,t,\ 1 - 2\,t + 395652331551232130724754891 2\,t^{51} \\
&- 18641311775009975389916336220 0\,t^{52} + 430508785143626601457690481760 0\,t^{53} \\
&- 6494836240068045765201207021120 0\,t^{54} \\
&+ 7197459433311770716163883053404 80\,t^{55} \\
&- 6246366579624143871527941364204 880\,t^{56} \\
&+ 4419943603125856189794625175840 8800\,t^{57} \\
&- 2621483792198783671188536311188 38400\,t^{58} \\
&+ 1329625465788959345090117781564 616800\,t^{59} \\
&- 5855276588233602449304259415853 071760\,t^{60} \\
&+ 2265320122660869144320992167444 7949760\,t^{61} \\
&- 7772505992707380935060295706482 1411200\,t^{62} \\
&+ 2383157260726945635908699133283 80993600\,t^{63} \\
&- 6570868697244727510546341119895 50479950\,t^{64} \\
&+ 1637662659928685933397703479112 418119260\,t^{65} \\
&- 3705418543677026758394803831527 087461760\,t^{66}
\end{aligned}
$$

$$+ 7638969199184915238854959391487745606800\, t^{67}$$
$$- 14392452349329364524417042867353209283400\, t^{68}$$
$$+ 24844941897715102689492866270213591548800\, t^{69}$$
$$- 39378298887506343210113505336549060890880\, t^{70}$$
$$+ 57403576547280373552771095807504616932480\, t^{71}$$
$$- 77069616660700501529183415667483050511200\, t^{72}$$
$$+ 95401244035786174993783705072824598017600\, t^{73}$$
$$- 108965933258265760392429801798807883987200\, t^{74}$$
$$+ 114898522957882451791573202118965202115392\, t^{75}$$
$$- 111874877616885545165479170484255591533408\, t^{76}$$
$$+ 100586803052144845803128125310519512867200\, t^{77}$$
$$- 83488001773575114180374151492302045817600\, t^{78}$$
$$+ 63937014016472081112818179307395870531200\, t^{79}$$
$$- 45141736620250546923619042114445877556080\, t^{80}$$
$$+ 29351417226335746559390159070298142608480\, t^{81}$$
$$- 1035090093500472043472513275012228416 0\, t^{90}$$
$$+ 253085215169071461178718905154088816 0\, t^{91}$$
$$- 54347567414355218333712172103608680 0\, t^{92}$$
$$+ 10129295718804556822053881539023840 0\, t^{93}$$
$$- 1613870965588850715834413585138880 0\, t^{94}$$
$$+ 215440191100138444841053296772128 0\, t^{95}$$
$$- 23439094865061358582244224417338 0\, t^{96}$$
$$+ 1996157787701325921392381281620 0\, t^{97} - 124814034489357764950283501760 0\, t^{98}$$
$$+ 509552245179617138054608572 00\, t^{99} - 10191044903592342761092171 44\, t^{100}$$
$$- 17550808097572909036299387012924145068800\, t^{82}$$
$$+ 9634442095730911662245672539474173008400\, t^{83}$$
$$- 4845026075558762935487181645031384261800\, t^{84}$$
$$+ 2226364923301742968279230873564594567360\, t^{85}$$
$$- 931966712079799382070375714515411679360\, t^{86}$$
$$+ 354099740541175371859433940061730491200\, t^{87}$$
$$- 121585844571570659010702440107191243600\, t^{88}$$
$$+ 37532708968870422237484727224100431200\, t^{89}]$$

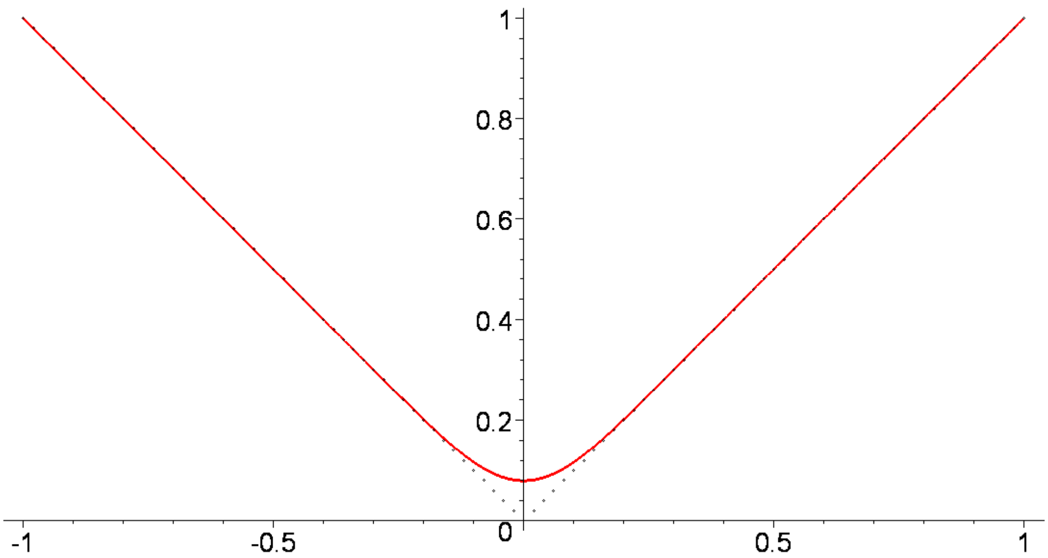
```
> krivka:=plot([op(bezier),t=0..1],thickness=3):
> display([body,krivka],scaling=constrained);
```




```
> Digits:=100;
```

$$Digits := 100$$

```
> krivka:=plot([op(bezier),t=0..1],thickness=3):
> display([body,krivka],scaling=constrained);
```



>

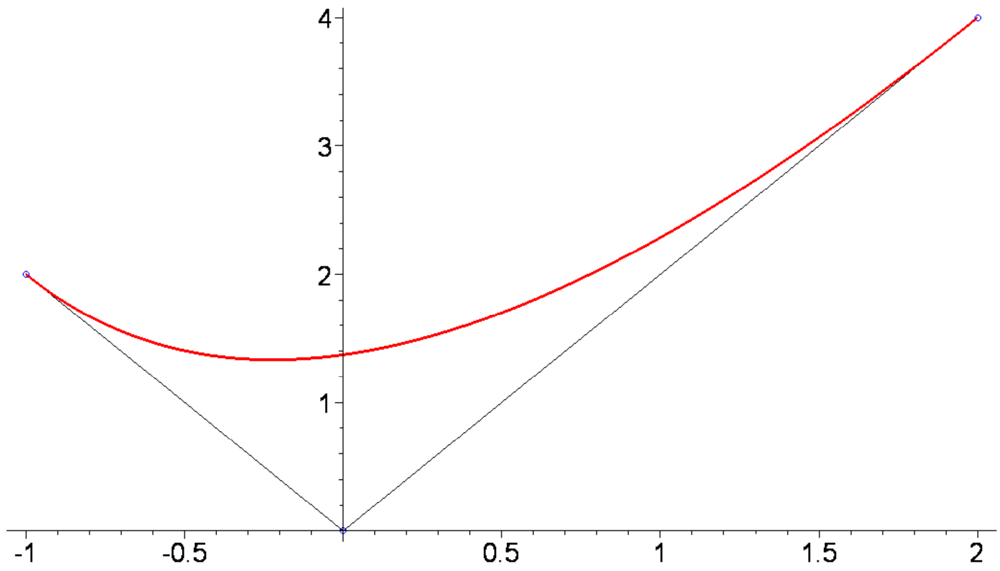
Cvičení

1) Napište proceduru, která vypočítá Lagrangeův interpolační polynom procházející danými body. Volání procedury by mohlo vypadat následovně:

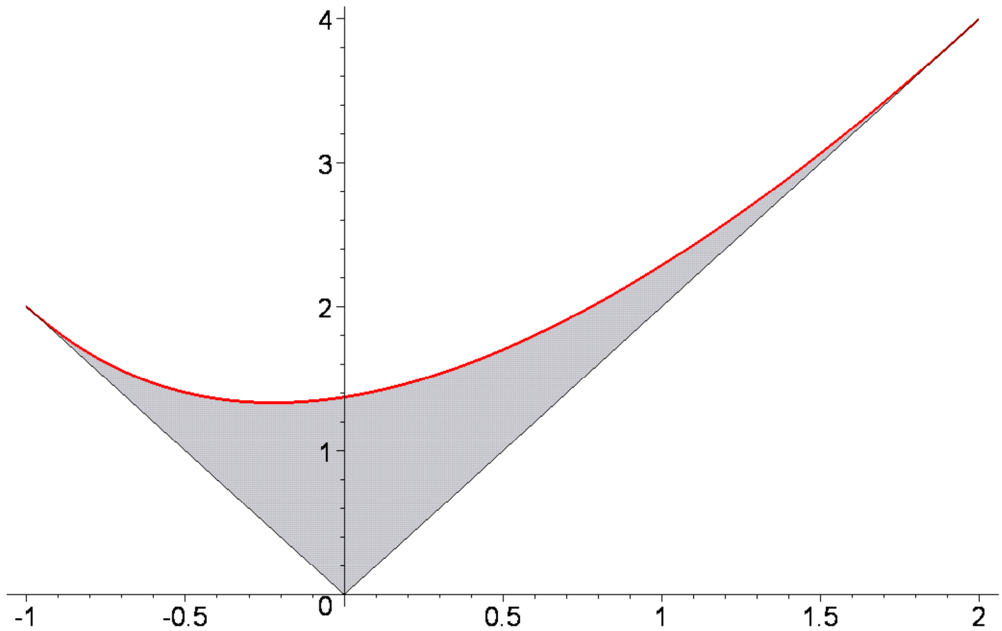
```
poly([[-1,0],[3,2],[6,5],[4,2]]);
```

Na vstupu tedy bude seznam bodů, kterými má polynom procházet (počet bodů není pevně stanoven). Proceduru napište bez použití příkazů z balíku CurveFitting.
Poté svou proceduru prověřte na seznamu bodů $[[1, 2], [3, 7], [2, -7], [-1, 1], [4, 1]]$. K ověření správnosti použijte příkaz PolynomialInterpolation z balíku CurveFitting. Jaká je hodnota tohoto interpolačního polynomu v bodě 470 ?

2) Na přednášce jsme si ukazovali kvadratickou Bézierovu křivku určenou body $[-1, 2]$, $[0, 0]$ a $[2, 4]$. Můžete si všimnout, že tato křivka je grafem jisté funkce (viz obrázek níže).



Určete tuto funkci. Jakou hodnotu má tato funkce v bodě 0 ?



Jaký je obsah vybarvené plochy?

3) Vytvořte animace ilustrující vznik kvadratické a kubické Bézierovy křivky (viz přednáška).

>