# Digitální učební materiál

| Číslo projektu: | CZ.1.07/1.5.00/34.0548 |
|---|---|
| Název školy: | Gymnázium, Trutnov, Jiráskovo náměstí 325 |
| Název materiálu: | VY_32_INOVACE_159_IVT |
| Autor: | Ing. Pavel Bezděk |
| Tematický okruh: | Algoritmy |
| Datum tvorby: | září 2013 |
| Ročník: | 4. ročník a oktáva |
| Anotace: | Algoritmus XIX. – Algoritmus interpolačních polynomů II. Newtonův polynom |
| Metodický pokyn: | Při výuce nutno postupovat individuálně.<br>Části DUM – „ Pro hloubavé“, jsou určeny pro zájemce o studium na technických a matematicko-fyzikálních oborech vysokých škol. |
| Pokud není uvedeno jinak, je použitý materiál z vlastních zdrojů autora DUM. | |

esf evropský sociální fond v ČR
EVROPSKÁ UNIE
MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY
OP Vzdělávání pro konkurenceschopnost
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

| | | | |
|---|---|---|---|
| **Autor** | **Ing. Pavel Bezděk** | | |
| **Vytvořeno dne** | **23. 9. 2013** | | |
| **Odpilotováno dne** | **24. 4. 2014** | **ve třídě** | **4.A** |
| **Vzdělávací oblast** | **Informatika a informační a komunikační technologie** | | |
| **Vzdělávací obor** | **Informatika a výpočetní technika** | | |
| **Tematický okruh** | **Algoritmus** | | |
| **Téma** | **Algoritmus XIX. - Algoritmus interpolačních polynomů II. - Newtonův polynom** | | |
| **Klíčová slova** | **Algoritmus, Newtonův interpolační polynom** | | |

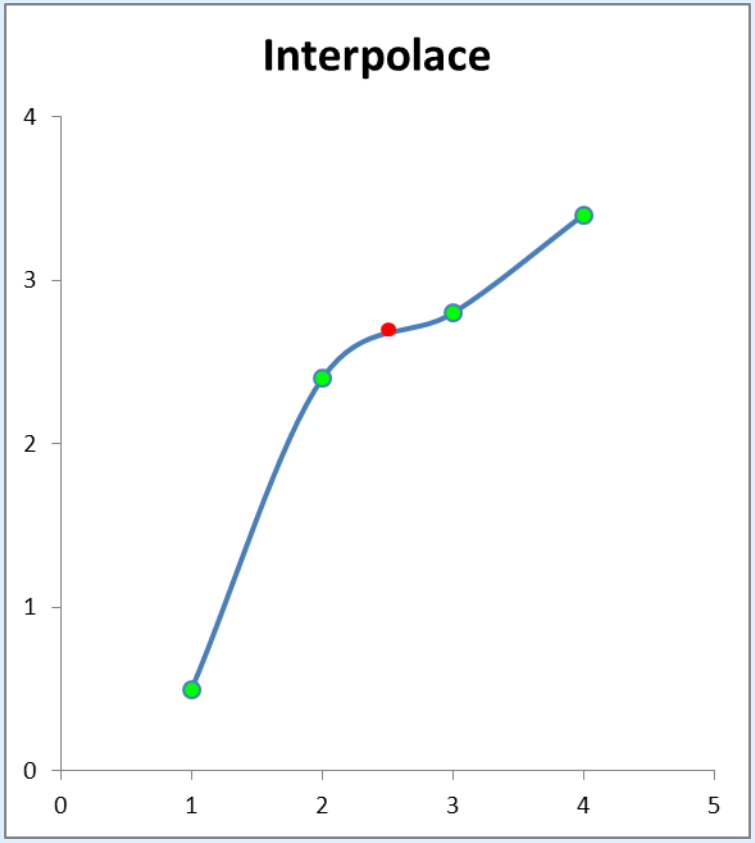
# Newtonův interpolační polynom

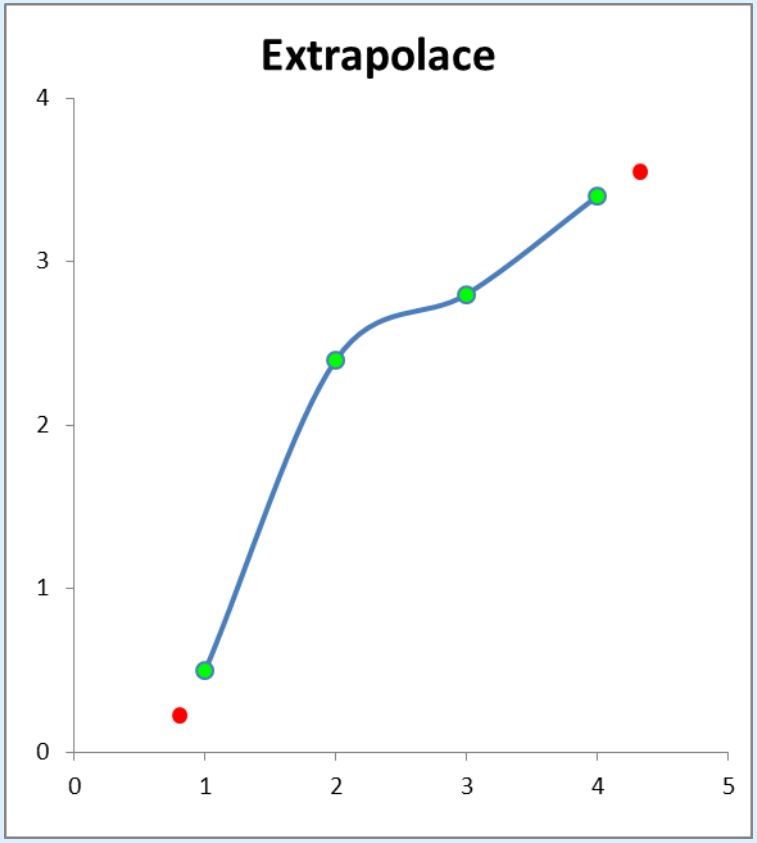
# Interpolace x extrapolace

***Interpolace*** je přibližný výpočet hodnot funkce v bodě, které leží uvnitř daného intervalu **int($x_0$,$x_1$,$x_2$, … $x_n$)**.

***Extrapolace*** je přibližný výpočet hodnot funkce v bodě, které leží vně intervalu **int($x_0$,$x_1$,$x_2$, … $x_n$)** z hodnot funkce v krajních bodech $x_0$ a $x_n$, popř. i z hodnot funkce v některých vnitřních bodech intervalu int($x_0$,$x_1$,$x_2$, … $x_n$). Chyba výpočtu může být velká. Ale pro x velmi blízká koncovým bodům intervalu int($x_0$,$x_1$,$x_2$, … $x_n$), lze dostat dobré výsledky.

# Interpolace x extrapolace




**Vypočteme bod funkce ● mezi známými body funkce ●**

**Vypočteme bod funkce ● mimo známé body funkce ●**

# Aproximace funkce dané několika body

Chceme-li aproximovat funkci danou svými body $x_0, x_1, x_2, x_3, \ldots. x_n$ (tzv. **uzly interpolace**), a požadujeme, aby aproximace procházela zadanými body, použijeme aproximaci interpolačním polynomem.

## Newtonův interpolační polynom

$$N_n(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) \ldots + a_n(x-x_0)(x-x_1)(x-x_2) \ldots (x-x_{n-1})$$

Tato interpolace nám poslouží k získání přibližné hodnoty funkce v libovolném bodě intervalu. Jsou-li uzly interpolace navzájem různá čísla, určují vstupní data jednoznačně koeficienty $a_0, a_1, a_2, \ldots a_n$ takové, že $N_n$ splňuje interpolační podmínky

$$N_n(x_i) = f(x_i) \quad i = 0,1,2,3,\ldots\ldots n$$

$N_n(x_i) = f(x_i) \quad i = 0,1,2,3,\ldots\ldots n$

$N_n(x_0) = a_0 = f(x_0)$

$N_n(x_1) = a_0 + a_1(x_1 - x_0) = f(x_1)$

$N_n(x_2) = a_0 + a_1(x_1 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = f(x_2)$

$N_n(x_n) = a_0 + a_1(x_n - x_0) \ldots\ldots \quad a_n(x_n - x_0)(x_n - x_1) \ldots\ldots\ldots (x_n - x_{n-2})(x_n - x_{n-1}) = f(x_n)$

Jedná se o soustavu rovnic

z 1. rovnice vypočítáme $a_0 = f(x_0)$

z 2. rovnice $a_1 = (f(x_1) - a_0)/(x_1 - x_0)$

atd., to ale není moc praktické. Pro zjednodušení výpočtu si zavedeme

Poměrné diference $f[x_i] = f(x_i)$ následně $f[x_i, x_{i+1}] = (f[x_{i+1}] - f[x_i])/(x_{i+1} - x_i)$

Platí, že $a_j = f[x_0, x_1, \ldots x_{n-1}, x_j]$

# Poměrné diference

**Poměrná diference 0. řádu**

$f[x_i] = f(x_i)$

**z diferencí 0.řádu vypočteme diference 1.řádu:**

**Poměrná diference 1. řádu**

$f[x_i,x_{i+1}]=(f[x_{i+1}]-f[x_i])/(x_{i+1}-x_i) = (f(x_{i+1})-f(x_i))/(x_{i+1}-x_i)$

**Z diferencí 1.řádu vypočteme diference 2.řádu:**

**Poměrná diference 2. řádu**

$f[x_i,x_{i+1},x_{i+2}]=(f[x_{i+1},x_{i+2}]-f[x_i,x_{i+1}])/(x_{i+2}-x_i)$

**Z diferencí 2.řádu vypočteme diference 3.řádu:**

**Poměrná diference 3. řádu**

$f[x_i,x_{i+1},x_{i+2},x_{i+3}]=(f[x_{i+1},x_{i+2},x_{i+3}]-f[x_i,x_{i+1},x_{i+2}])/(x_{i+3}-x_i)$

**Poměrná diference (k-1) -ního řádu**

$$f[x_i,x_{i+1},\ldots x_{i+k-2},x_{i+k-1}]=(f[x_{i+1},\ldots x_{i+k-2},x_{i+k-1}]-f[x_i,\ldots x_{i+k-3},x_{i+k-2}])/(x_{i+k-1}-x_i)$$

**Poměrná diference k -tého řádu pro $i+k \leq n$**

$$f[x_i,x_{i+1},\ldots x_{i+k-1},x_{i+k}]=(f[x_{i+1},\ldots x_{i+k-1},x_{i+k}]-f[x_i,\ldots x_{i+k-2},x_{i+k-1}])/(x_{i+k}-x_i)$$

**Poměrná diference n -tého řádu existuje pouze pro i=0**

$$f[x_0,x_1,\ldots x_{n-1},x_n]=(f[x_1,\ldots x_{i+k-1},x_n]-f[x_0,\ldots x_{n-2},x_{n-1}])/(x_n-x_0)$$

# Schéma výpočtu následných diferencí,

**koeficienty Newtonova polynomu jsou hodnoty na hlavní diagonále tabulky**

$$a_j = f[x_0, x_1, \ldots x_{n-1}, x_j]$$

| $x_i$ | $f(x_i)$ | 0.řád | 1.řád | 2.řád | 3.řád | 4.řád |
|---|---|---|---|---|---|---|
| $x_0$ | $f(x_0)$ | $f[x_0]$ | | | | |
| $x_1$ | $f(x_1)$ | $f[x_1]$ | $f[x_0,x_1]$ | | | |
| $x_2$ | $f(x_2)$ | $f[x_2]$ | $f[x_1,x_2]$ | $f[x_0,x_1,x_2]$ | | |
| $x_3$ | $f(x_3)$ | $f[x_3]$ | $f[x_2,x_3]$ | $f[x_1,x_2,x_3]$ | $f[x_0,x_1,x_2,x_3]$ | |
| $x_4$ | $f(x_4)$ | $f[x_4]$ | $f[x_3,x_4]$ | $f[x_2,x_3,x_4]$ | $f[x_1,x_2,x_3,x_4]$ | $f[x_0,x_1,x_2,x_3,x_4]$ |

# Výpočet Newtonova polynomu

**Funkce *f* dána svými hodnotami v bodech $x_0=0$, $x_1=1$, $x_2=-1$, $x_3=3$,**
**$f(0)=1$, $f(1)=2$, $f(-1)=2$, $f(3)=0$.**

**Vypočítáme přibližnou hodnotu f(2) pomocí Newtonova interpolačního polynomu:**

| i | $x_i$ | $f(x_i)$ | 0.řád | 1.řád | 2.řád | 3.řád |
|---|---|---|---|---|---|---|
| 0 | 0 | f(0) | $f[x_0]=1$ | | | |
| 1 | 1 | f(1) | $f[x_1]=2$ | $f[x_0,x_1]= 1$ | | |
| 2 | -1 | f(-1) | $f[x_2]=2$ | $f[x_1,x_2]= 0$ | $f[x_0,x_1,x_2]= 1$ | |
| 3 | 3 | f(3) | $f[x_3]=0$ | $f[x_2,x_3]=-1/2$ | $f[x_1,x_2,x_3]=-1/4$ | $f[x_0,x_1,x_2,x_3]=-5/12$ |

$$a_j=f[x_0,x_1, \ldots x_{n-1},x_j]$$

Byly zadány hodnoty funkce : $x_0=0, x_1=1, x_2=-1, x_3=3,$

$f(0)=1, f(1)=2, f(-1)=2, f(3)=0,$

a vypočetli jsme poměrné diference:

$a_0= f[x_0]=1,\ a_1= f[x_0,x_1]= 1,\ a_2=f[x_0,x_1,x_2]=1,\ a_3=f[x_0,x_1,x_2,x_3]=-5/12$

$$N_n(x) =a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)\ldots+a_n(x-x_0)(x-x_1)(x-x_2)\ldots\ldots(x-x_{n-1})$$

$$a_j=f[x_0,x_1,\ \ldots x_{n-1},x_j]$$

$$N_3 (x) =a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)+a_3(x-x_0)(x-x_1)(x-x_2)$$

$$N_3(x)= f[x_0]+f[x_0,x_1]\ (x-x_0)+f[x_0,x_1,x_2]\ (x-x_0)(x-x_1)+f[x_0,x_1,x_2,x_3]\ (x-x_0)(x-x_1)(x-x_2)$$

$$N_3(x)=1 + 1(x-0) + 1(x-0)(x-1) – (5/12)(x-0)(x-1)(x+1)$$

$$N_3(x)=1 + x + x^2 – x - (5/12)x^3 + (5/12)x = (-5/12)x^3 + x^2 + (5/12)x + 1$$

$$N_3(x) = (-5/12)x^3 + x^2 + (5/12)x + 1$$

$$N_3(2) = (-5/12)2^3 + 2^2 + (5/12)2 + 1=(-40+48+10+12)/12=30/12=2,5$$

Naprosto stejný polynom i hodnotu f(2)=2,5 dostaneme při použití Lagrangeova interpol. polynomu

# Výhody Newtonova interpolačního polynomu oproti Lagrangeovu interpolačnímu polynomu

**Koeficienty Newtonova polynomu (poměrné diference $a_j=f[x_0,x_1, \ldots x_{n-1},x_j]$) nezávisejí na x, proto je možné je vypočítat při daných vstupních hodnotách funkce „jednou provždy“. A přibydou-li nám další body funkce, dopočítají se další koeficienty Newtonova polynomu, ale původní zůstávají.**

**Přidáme-li totiž k původním uzlům interpolace $x_0, x_1, x_2, x_3, \ldots . x_n$ další bod $x_{n+1}$ různý od všech ostatních uzlů, tak k původním koeficientům Newtonova interpolačního polynomu $N_n(x)$**

$f[x_0], f[x_0,x_1], f[x_0,x_1,x_2], f[x_0,x_1,x_2,x_3], \ldots f[x_0,x_1,x_2, \ldots\ldots x_{n-1},x_n]$

**přibude jeden koeficient $f[x_0,x_1,x_2, \ldots x_{n-1},x_n,x_{n+1}]$ a vznikne nový polynom $N_{n+1}(x)$**

**s koeficienty**

$f[x_0], f[x_0,x_1], f[x_0,x_1,x_2], f[x_0,x_1,x_2,x_3], \ldots f[x_0,x_1,x_2, \ldots\ldots x_{n-1},x_n], f[x_0,x_1,x_2, \ldots x_{n-1},x_n,x_{n+1}]$

**U Lagrangeova interpolačního polynomu, po přidání dalších uzlů interpolace, musíme přepočítat celý polynom.**

Prohloubavé

# Algoritmus Newtonova interpolačního polynomu

**Hodnotu vypočtenou v předchozím řádku (poměrnou diferenci (k-1). řádu) použiji k výpočtu ( poměrné diference k. řádu) v dalším řádku.** Ale musím začít počítat od $a_4$. Od $a_1$ by to nešlo, protože bych počítal pom. diferenci k. řádu pomocí pom. diferencí k. řádu. (Správně pro výpočet pom. diference k. řádu použiji pom. diference (k-1). řádu.)

Např. pro výpočet diferencí 2. řádu používám diference 1. řádu $a_3=(a_3-a_2)/(x_3-x_1)$

Poměrná diference 2. řádu=(rozdíl pom. diferencí 1. řádu)/(rozdíl bodů)

Červeně výsledné hodnoty koeficientů.

$$N_n(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) \ldots\ldots + a_n(x-x_0)(x-x_1)(x-x_2)\ldots\ldots(x-x_{n-1})$$

Poměrná diference 0. řádu

$a_0=f(x_0)$; $a_1=f(x_1)$; $a_2=f(x_2)$; $a_3=f(x_3)$; $a_4=f(x_4)$;

Poměrná diference 1. řádu

$a_4=(a_4-a_3)(x_4-x_3)$; $a_3=(a_3-a_2)(x_3-x_2)$; $a_2=(a_2-a_1)(x_2-x_1)$; $a_1=(a_1-a_0)(x_1-x_0)$;

Poměrná diference 2. řádu

$a_4=(a_4-a_3)(x_4-x_2)$; $a_3=(a_3-a_2)(x_3-x_1)$; $a_2=(a_2-a_1)(x_2-x_0)$;

Poměrná diference 3. řádu

$a_4=(a_4-a_3)(x_4-x_1)$; $a_3=(a_3-a_2)(x_3-x_0)$;

Poměrná diference 4. řádu

$a_4=(a_4-a_3)(x_4-x_0)$;

# Newtonův interpolační polynom -jednoduchý program v Pascalu

```
program newton;
uses crt;
const n=5;
type pole=array[0..n] of  real;
var a,x,clen:pole;
     pocet,m,r, i,j,k,p,radu: integer;   y,bod:real;
begin
clrscr;
write('Zadej pocet bodu, maximalne 6:'); read(pocet);
writeln;          {vypln prvky pole nulami, kdyz nevyuzijes cele pole}
if pocet<n+1 then
            for  i:=0 to N do begin a[i]:=0; x[i]:=0; end;
m:=pocet-1; r:=pocet-1;
for i:=0 to m do  begin
              write(' Zadej bod:x[',i,']:');  read(x[i]);
              write('      f(x[',i,']):');   read(a[i])
                    end;
```

```
k:=1;
for j:=r downto 1 do begin   for i:=n   downto 1  do  begin
                                        p:=i+k-1;
                                        if (k<=p) and (p<=M) then
                                             a[p]:=(a[p]-a[p-1])/(x[p]-x[p-k]);
                                                        end;
                                k:=k+1;
                        end;

clen[0]:=a[0]-a[1]*x[0]+a[2]*x[1]*x[0]-a[3]*x[2]*x[1]*x[0]-a[4]*x[3]*
x[2]* x[1]* x[0]-a[5]* x[4]* x[3]* x[2]* x[1]* x[0];

clen[1]:=a[1] -a[2]*( x[1]+x[0]) +a[3]*( x[2]*x[1]+x[2]*x[0]+
x[1]*x[0])  -a[4]*(x[3]*x[2]*x[1] +x[3]*x[2]*x[0] + x[3]*x[1]*x[0]+
x[2]*x[1]*x[0]) +a[5]*(x[4]*x[3]*x[2]*x[1]+x[4]*x[3]*x[2]*x[0]+
x[4]*x[2]*x[1]*x[0]+ x[3]*x[2]*x[1]*x[0]);

clen[2]:=a[2] - a[3]*(x[2]+ x[1]+x[0])+ a[4]*(x[3]*x[2]+x[3]*x[1]+
x[2]*x[1]+x[2]*x[1]+ x[1]*x[0]) -a[5]*(x[4]*x[3]*x[2]+
x[4]*x[3]*x[1]+ x[4]*x[2]*x[1]+ x[4]*x[2]*x[0]+x[4]*x[1]*x[0]+
x[3]*x[2]*x[1]+x[3]*x[2]*x[0] + x[3]*x[1]*x[0]+ x[2]*x[1]*x[0]);
```

```
clen[3]:=  a[3] -a[4]*(x[3]+x[2]+ x[1]+x[0]) +a[5]*(x[4]*x[3]+
x[4]*x[2]+ x[4]*x[1]+ x[4]*x[0]+ x[3]*x[2]+ x[3]*x[1]+ x[3]*x[0]+
x[2]*x[1]+ x[2]*x[0]+ x[1]*x[0]);

clen[4]:=a[4] -a[5]*(x[4]+x[3]+x[2]+x[1]+x[0]);

clen[5]:=a[5];
writeln;
writeln;
writeln;
writeln('Newton…v interpolacni polynom:');
writeln;
writeln('pomerne diference:');
writeln;
for i:=0 to pocet-1 do write('a',i,' = ',a[i]:5:5,'    ');
writeln;
writeln;
```

```
writeln;
radu:=pocet-1;

writeln('Polynom  ',radu:2,'. radu:');
writeln;
writeln('N',pocet-1,'(x) = ',clen[5]:7:5,'*x^5 + ', clen[4]:7:5,'*x^4 + ',
clen[3]:7:5,'*x^3 + ',clen[2]:7:5,'*x^2 + ',clen[1]:7:5,'*x + ',clen[0]:7:5);
 writeln;

write('Hodnota v bode: ');
read(bod);
y:=a[0]+a[1]*(bod-x[0])+a[2]*(bod-x[0])*(bod-x[1])+a[3]* (bod-x[0])* (bod-x[1])*
(bod-x[2])+a[4]* (bod-x[0])* (bod-x[1])* (bod-x[2])+a[4]* (bod-x[0])* (bod-x[1])*
(bod-x[2])* (bod-x[3])+a[5]* (bod-x[0
])* (bod-x[1])* (bod-x[2])* (bod-x[3])* (bod-x[4]);
writeln;
write('f(',bod:3:3,') = ',y:3:3);
repeat until keypressed;
end.
```

# Newtonův interpolační polynom

## jednoduchá verze

## C++

```
#include <iostream>
using namespace std;
const int n=50;
double x[n],a[n],clen[n];   double bod,y;   int pocet,m,r,i,j,k,p,radu;
int main()
{ cout<<" Zadej pocet bodu, maximálne 50: "; cin>>pocet; cout<<endl;
                                                    //vypln prvky pole nulami
if (pocet<n+1)  for( i=0;i<n+1;i++) { a[i]=0; x[i]=0;}
m=pocet-1; r=pocet-1;
for (i=0;i<m+1;i++) { cout<<"  Zadej bod: x["<<i<<"] = "; cin>>x[i];
                cout<<"          f(x["<<i<<"]) = ";cin>>a[i];
              }
k=1;                                                  // dif[0]=a[0];
for (j=r; j>=1;j--){ for (i=n;i>=1;i--) { p=i+k-1; if ((k<=p) &&  (p<=m))   a[p]=(a[p]-a[p-1])/(x[p]-x[p-
k]); }
                                                    //  dif[k]=a[k];  z[j]=x[m]-x[m-k];
              k=k+1;
                  }
```

```
clen[0]=a[0]-a[1]*x[0]+a[2]*x[1]*x[0]-a[3]*x[2]*x[1]*x[0]-a[4]*x[3]*x[2]*x[1]*x[0]-
a[5]*x[4]*x[3]*x[2]*x[1]*x[0];
clen[1]=a[1]-a[2]*(x[1]+x[0])+a[3]*(x[2]*x[1]+x[2]*x[0]+x[1]*x[0])-
a[4]*(x[3]*x[2]*x[1]+x[3]*x[2]*x[0]+x[3]*x[1]*x[0]+x[2]*x[1]*x[0])+a[5]*(x[4]*x[3]*x[2]*x[1]+
x[4]*x[3]*x[2]*x[0]+x[4]*x[2]*x[1]*x[0]+x[3]*x[2]*x[1]*x[0]);
clen[2]=a[2]-a[3]*(x[2]+x[1]+x[0])+a[4]*(x[3]*x[2]+x[3]*x[1]+x[2]*x[1]+x[2]*x[1]+x[1]*x[0])-
a[5]*(x[4]*x[3]*x[2]+x[4]*x[3]*x[1]+x[4]*x[2]*x[1]+x[4]*x[2]*x[0]+x[4]*x[1]*x[0]+x[3]*x[2]*x[
1]+ x[3]*x[2]*x[0]+x[3]*x[1]*x[0]+x[2]*x[1]*x[0]);
clen[3]=a[3]-
a[4]*(x[3]+x[2]+x[1]+x[0])+a[5]*(x[4]*x[3]+x[4]*x[2]+x[4]*x[1]+x[4]*x[0]+x[3]*x[2]+x[3]*x[1]+
x[3]*x[0]+x[2]*x[1]+x[2]*x[0]+x[1]*x[0]);
clen[4]=a[4]-a[5]*(x[4]+x[3]+x[2]+x[1]+x[0]);
clen[5]=a[5];
cout<<endl<<endl<<endl;
```

```
cout<<"   Newtonuv interpolacni polynom: "<<endl;
cout<<"   pomerne diference: "<<endl;
for (i=0; i< pocet;i++) cout<<"   a "<<i<<" = "<<a[i]<<endl<<endl<<endl;
radu=pocet-1;
cout<<"   Polynom  "<<radu<<". radu:"<<endl<<endl;
cout<<"   N"<<pocet-1<<"(x) = "<<clen[5]<<" *x^5 + "<< clen[4]<<" *x^4 + "<<clen[3]<<" *x^3 + "<<clen[2]<<" *x^2 + "<<clen[1]<<" *x + "<<clen[0]<<endl<<endl;
cout<<"   Vypocet hodnoty f(x) v bode x "<<endl;  cout<<"   Zadani bodu x: "; cin>>bod;
cout<<endl;
y=a[0]+a[1]*(bod-x[0])+a[2]*(bod-x[0])*(bod-x[1])+a[3]*(bod-x[0])*(bod-x[1])*(bod-x[2])+a[4]*(bod-x[0])*(bod-x[1])*(bod-x[2])*(bod-x[3])+a[5]*(bod-x[0])*(bod-x[1])*(bod-x[2])*(bod-x[3])*(bod-x[4]);
cout<<"   Hodnota f("<<bod<<") = "<<y<<endl;
return 0;
}
```

# Použité zdroje

**PŘIKRYL, Petr. *Numerické metody analýzy*. Praha: Státní nakladatelství technické literatury, 1988. MATEMATIKA PRO VYSOKÉ ŠKOLY TECHNICKÉ, sešit XXIV.**