Num. Math.—9

Lagrange Interpoláció

- **Adjuk meg az Lagrange alapinterpolációs polinomokat, majd ezek segítségével állítsuk elõ a Lagrange interpolációs polinomot! Próbáljuk ki a következõ adathalmazokon:**

  x={1,2,3} , y={2,5,10}
  x={1,2,3,4,5}, y={0,6,24,60,120}

- **Javaslat**

$$\text{LagrBase}[j\_, x\_List, var\_] := \left(\prod_{k=1}^{j-1} \frac{var - x[[k]]}{x[[j]] - x[[k]]}\right) \left(\prod_{k=j+1}^{\text{Length}[x]} \frac{var - x[[k]]}{x[[j]] - x[[k]]}\right)$$

$$\text{LagrInterp}[x\_List, y\_List, var\_] := \sum_{i=1}^{\text{Length}[x]} y[[i]]\ \text{LagrBase}[i, x, var]$$

- **Példa**

```
x0 = {1, 2, 3}; y0 = {2, 5, 10};
```

```
LagrInterp[x0, y0, x]
```

$(2 - x)\ (3 - x) + 5\ (3 - x)\ (-1 + x) + 5\ (-2 + x)\ (-1 + x)$

```
Expand[%]
```

$1 + x^2$

```
Expand[InterpolatingPolynomial[Transpose[{x0, y0}], x]]
```

$1 + x^2$

```
? InterpolatingPolynomial
```

InterpolatingPolynomial[data, var] gives a polynomial in the variable var which provides an exact fit to a list of data. The data can have the forms {{x1, f1}, {x2, f2}, ... } or {f1, f2, ... }, where in the second case, the xi are taken to have values 1, 2, ... The fi can be replaced by {fi, dfi, ddfi, ... }, specifying derivatives at the points xi. More…

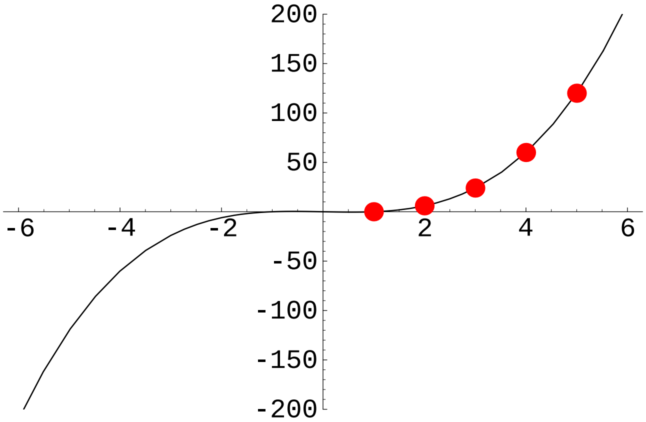
■ Feladat

Ábrázoljuk a pontokat és az interpolációs polinomot!

```
x1 = {1, 2, 3, 4, 5}; y1 = {0, 6, 24, 60, 120};

Expand[LagrInterp[x1, y1, x]]
```

$-x + x^3$

```
Plot[LagrInterp[x1, y1, x], {x, -6, 6}, ImageSize → {300, 300},
  Epilog → {RGBColor[1, 0, 0], PointSize[.03], Map[Point[#] &, Transpose[{x1, y1}]]},
  PlotRange → {-200, 200}];
```



Hermite Interpoláció

A különbség: a helyettesítési értékek mellett a deriváltak is adottak (bizonyos pontokban)

- **Oldjuk meg a problémát, ha adott 1 pont (a), továbbá fa, fda, fdda, azaz a pontban a 0. 1. és 2. derivált**
  x1=a, f(x1)=fa f'(x1)=fda, f''(x1)=fdda
  Milyen polinomot kapunk? Cos[x], a=1 esetén?

- **Javaslat**

```
InterpolatingPolynomial[{{a, {fa, fda, fdda}}}, x]
```

$$fa + (-a + x)\left(fda + \frac{1}{2}\, fdda\, (-a + x)\right)$$

- **Oldjuk meg a problémát, ha adott a,b (a<b) valós paraméterek esetén 3 pont a, a+b,b illetve a másodikban a derivált értéke is!**
  Használjuk ezt f közelítõ integrálásra [a,b]−n! Melyik közelítõ numerikus integrálási eljárást kapjuk?
  Vessük össze ezt azzal a formulával, amikor a
  deriváltra vonatkozó információt a középsõ pontban nem használjuk!

  x1=a f(x1)=fa
  x2=(a+b)/2 f(x2)=fm f'(x2)=fdm
  x3=b f(x2)=fb

- **Javaslat**

IP1: Lagrange interpolációs polinom
IP2: Hermite interpolációs polinom

```
IP1 = LagrInterp[{a, (a + b) / 2, b}, {fa, fm, fb}, x];

Integrate[IP1, {x, a, b}]
```

$$-\frac{1}{6}\,(a - b)\,(fa + fb + 4\, fm)$$

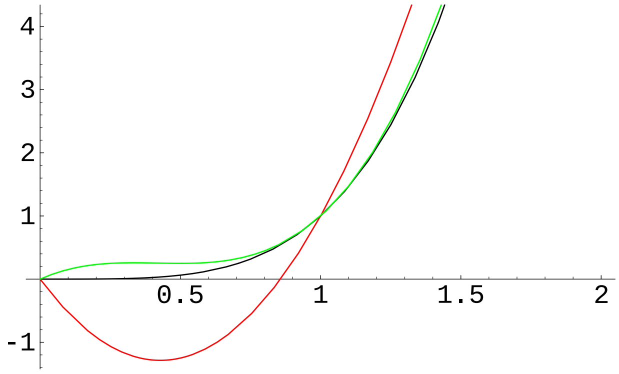
```
IP2 = InterpolatingPolynomial[{{a, fa}, {(a + b) / 2, {fm, fdm}}, {b, fb}}, x];

Simplify[Integrate[IP2, {x, a, b}]]
```

$$-\frac{1}{6}\,(a - b)\,(fa + fb + 4\, fm)$$

A két integrál értéke megegyezik (Simpson formula)

■ Abrázoljuk együtt az "elõzõ" két, interpolációs polinomot és a függvényt [a,b] intervallumon. A vizualizációs függvény neve legyen VisHerInterp[f,a,b,var]

Teszteljük [0,2]−n pl. az x^4 függvénnyel.



■ Javaslat

■ A Lagrange interpolációs polinom Newton féle alakja.

Elõny: lehet hozzávenni további osztópontokat. Határozzuk meg Newton féle alakkal, majd vegyük hozzá {4,11}

xi {1,2,3}, yi {2,5,10}

1 2 3 1
2 5 5
3 10

Osztott differenciák táblázata

■ Javaslat

```
Expand[2 + 3 (x - 1) + 1 (x - 1) (x - 2)]
```

$1 + x^2$

```
Expand[InterpolatingPolynomial[Transpose[{{1, 2, 3}, {2, 5, 10}}], x]]
```

$1 + x^2$

- **Feladat Ugyanez (Newton elõállítás) az alábbi adathalmazra.**
  xi {1,2,3,4,5}
  yi {0,6,24,60,120}