Computergestuetzte Mathematik zur Analysis

Lektion 11 (9. Januar)

```
> restart:
```

Implizite Funktionen

```
> g := solve(F(x, y), y);
> diff(g, x);
> diff(g, x$2);
> alias(alpha = g);
> diff(alpha, x);
> normal(diff(alpha, x$2));
```

Das gleiche zu Fuss:

```
> diff(F(x, y(x)), x);
> isolate(??=0, diff(y(x),x)); #Aufloesen nach d/dx y(x)
> diff(F(x,y(x)),x$2);
> isolate(%=0,diff(y(x),x$2));
> normal(algsubs(??,rhs(??))); #Ersetze d/dx y(x) (1.8) in der
  RHS von (1.10)
```

Taylor-Entwicklung in mehreren Veraenderlichen

```
> f := (1-y^2)*exp(-x^2-y);
> mtaylor(f, [x=0, y=0], 4);
> for n from 1 to 9 do;
>   p[n] := mtaylor(f, [x=0, y=0], n);
> od:
> for n from 1 to 6 do;
>     'n' = n, 'p' = p[n];
> od;
> optionen := x = -1 .. 1, y = -1 .. 1, view = 0 .. 1.5,
  orientation = [-30, 70], axes = boxed, style = patchcontour,
  shading = zhue;
> plot3d(f, optionen, title = "f");
> plot3d(p[3], optionen, title = "p[3]");
> plot3d(p[6], optionen, title = "p[6]");
> plot3d(p[9], optionen, title = "p[9]");
```

Noch etwas Datenstrukturen

```
> liste1 := [28, 2^6, 0, -3];
> is_pos := t -> evalb(t > 0);
> map(is_pos, liste1);
> select(is_pos, liste1);
> f := cos(y^2);
> has(f, y);
> has(f, y^2);
> has(f, y^3);
> liste := [x^3, exp(-4), x^2 + 15];
> test1 := t -> has(t, x^3);
> select(test1, liste);
> test2 := t -> has(t, exp);
> select(test2, liste);
> test3 := t -> has(t, 15);
> select(test3, liste);
> A := [seq(a[j], j = 1 .. 5)];
> B := [seq(b[j], j = 1 .. 4)];
> zip(F, A, B);
> G := (a,b) -> (a,b);
> zip(G, A, B);
> zip(G, 1, B);
```