

Computergestuetzte Mathematik zur Analysis

Lektion 5 (Programmsteuerung)

For Schleifen

```
> liste :=[a,b,x+y];
```

liste:= [a, b, x + y] (1.1)

```
> for k in liste do  
  k^3;  
od;
```

a^3
 b^3
 $(x + y)^3$

(1.2)

```
> for l from 1 to 7 do  
  l, evalf(sin(Pi/l))  
end do;
```

1, 0.
2, 1.
3, 0.8660254040
4, 0.7071067810
5, 0.5877852524
6, 0.5000000000
7, 0.4338837393

(1.3)

```
> out := 0;  
for le in [1, z,x^2, sin(y)] do  
  out := out+le;  
od;
```

out:= 0
out:= 1
out:= 1 + z
out:= x² + z + 1
out:= x² + z + 1 + sin(y)

(1.4)

```
> for k from 20 by -2 while k>8 do  
  k,sin(k),unapply(x^k,x),k, x->x^k;  
  for l from 1 to 3 do
```

```
    erg := l+k; print(l,k,l + k);
  od;
  k+erg;
od:
```

```
1, 20, 21
2, 20, 22
3, 20, 23
1, 18, 19
2, 18, 20
3, 18, 21
1, 16, 17
2, 16, 18
3, 16, 19
1, 14, 15
2, 14, 16
3, 14, 17
1, 12, 13
2, 12, 14
3, 12, 15
1, 10, 11
2, 10, 12
3, 10, 13
```

(1.5)

```
> i:=0;
  from 1 to 4 do
    i:= i+1;
  od
```

```
i:= 0
i:= 1
i:= 2
i:= 3
i:= 4
```

(1.6)

```
> i:=0;
  from 1 while i<4 do
    i:=i+1;
  end do;
```

```
i:= 0
i:= 1
i:= 2
i:= 3
i:= 4
```

(1.7)

▼ Einfache logische Aussagen

```
> a:= 3; b:=5;
a:= 3
b:= 5 (2.1)
```

```
> a<b;
3 < 5 (2.2)
```

```
> evalb(a<b);
true (2.3)
```

```
> is(a=b);
false (2.4)
```

```
> is(a<>b);
true (2.5)
```

```
> a ≠ b; # a "!=" b # a ungleich in 2D Maple Input
3 ≠ 5 (2.6)
```

```
> a:=true; b:=false; c:=true;
a:= true
b:= false
c:= true (2.7)
```

```
> a and b
false (2.8)
```

```
> a or b
true (2.9)
```

```
> `or` (a,b);
true (2.10)
```

```
> `xor` (a,b);
true (2.11)
```

```
> `xor` (a,c);
false (2.12)
```

```
> a := 1+I;
I^2
a:= 1 + I
-1 (2.13)
```

```
> is(a,real)
false (2.14)
```

```
> is(a,integer);
false (2.15)
```

```
> is(a,complex);
true (2.16)
```

```

|
|
|> ? property

```

true (2.16)

▼ If Abfragen

```

|> a:=3; b:=5;

```

a:= 3
b:= 5 (3.1)

```

|> if b<a then
|   a
|   else
|     b
|   end if;

```

5 (3.2)

```

|> for j from 4 to 8 do
|   if isprime(j) then
|     print(j,evalf(sin(Pi/j))):
|     fi:
|   od;
|>

```

5, 0.5877852524
7, 0.4338837393 (3.3)

▼ While Schleife

```

|> q:=1016:
|   while is(q,even) do
|     q:=q/2;
|   end do:
|   q;

```

127 (4.1)

▼ Maple Funktionen

```

|> restart;
|> myfun := proc(x,n,m)
|   description "Berechne sin(nx) cos(mx)";
|   sin(n*x) * cos(m*x);
| end proc;
|myfun:= proc(x, n, m)
|   description "Berechne sin(nx) cos(mx)";
|   sin(n*x) * cos(m*x)
| end proc

```

(5.1)

```
> myfun(1,2,3)
sin(2) cos(3) (5.2)
```

```
> myfun2 := proc (x, n, m)
  local a,b;
  description "Berechne sin(nx) cos(mx)";
  a := sin(n*x);
  b := cos(m*x);
  a*b;
end proc;
myfun2:=proc(x, n, m) (5.3)
```

```
  local a, b;
  description "Berechne sin(nx) cos(mx)";
  a:= sin(n*x); b:= cos(m*x); a*b
end proc
```

```
> myfun2(1,2,3);
sin(2) cos(3) (5.4)
```

```
> a;
a (5.5)
```

```
> Describe(myfun);
# Berechne sin(nx) cos(mx)
myfun( x, n, m )
```

```
> divide2en:= proc (n)
  local q;
  q := n;
  while type(q, even) do
    q := (1/2)*q
  end do;
  q;
end proc;
divide2en:=proc(n) (5.6)
```

```
  local q;
  q:= n; while type(q, even) do q:= 1 / 2 * q end do; q
end proc
```

```
> divide2en(1016);
127 (5.7)
```