


# Computeralgebra-Praktikum

Universität Siegen  
Mohamed Barakat




WS 2017/2018  
Abgabe bis Mi. 29.11.2017, 16:00 Uhr

---

## 1 Erste Schritte mit GAP

- (i) Starte ein Terminal.
- (ii) Starte GAP durch Eingabe von `gap` auf der Kommando-Zeile, gefolgt von einem :

```
$ gap
GAP 4.8.8
https://www.gap-system.org
Architecture: x86_64-linux-gcc--default64
Libs used:  gmp, readline
Loading the library and packages ...
Components: trans 1.0, prim 2.1, small* 1.0, id* 1.0
Packages:  Browse 1.8.7, FactInt 1.5.4, GAPDoc 1.6, IO 4.4.6, ...
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap>
```

- (iii) Gebe folgende Befehle nach der `gap>`-Prompt ein. Jede Zeile endet mit , jeder vollständige Befehl mit einem  vor dem .

```
gap> SetUserPreference("UseColorPrompt", true);
gap> SetUserPreference("HistoryMaxLines", 10000);
gap> SetUserPreference("SaveAndRestoreHistory", true);
gap> WriteGapIniFile();
```

Hinweis: Alle Befehlszeilen (samt `gap>`-Prompt) können mit der Maus kopiert und in GAP gepastet werden.

- (iv) Um GAP zu beenden gebe folgendes ein:

```
gap> quit;
$
```



- (v) Starte GAP nochmal und gebe folgende Befehle ein:

```
gap> 0 = 0;
true
gap> 0 = 1;
false
gap> 1 + 1;
2
gap> 2^(2^(2^2));
65536
<integer 200...736 (19729 digits)>
```

```
gap> 2^(2^(2^(2^(2^2))));
Error, Integer operands: <exponent> is too large
not in any function at line 6 of *stdin*
you can replace the integer <exponent> via 'return <exponent>;'
brk>
```

Zum Verlassen der sogenannten break-loop: entweder `quit;`  oder `Strg-d` -

```
brk> quit;
gap> Factorial(50);
30414093201713378043612608166064768844377641568960512000000000000
gap> a := 0;
0
gap> for i in [ 1 .. 100 ] do
>   a := a + i;
> od;
gap> a;
5050
```

- (vi)
- Mit  und  kann man nun in der Zeilen-History rückwärts und vorwärts blättern. Durch Eingabe des exakten Anfangs eines früheren Befehls kann das rückwärts/vorwärts-Blättern eingeschränkt werden.
  - Die `Tab`-Taste kann benutzt werden, um Befehle automatisch zu vervollständigen: Etwa wird der Befehlsanfang `LoadP` `Tab` zu `LoadPackage` vervollständigt.
  - Mit `Strg-a` gelangt man an den Anfang der Zeile, mit `Strg-e` ans Ende.
  - Mit `Strg-k` wird die restliche Zeile ab der Cursor-Position abgeschnitten.
- (vii) GAP hat ein online verfügbares reference manual:

<http://www.gap-system.org/Manuals/doc/ref/chap0.html>

Das inline-Hilfesystem erreicht man mit `?`: Etwa

```
gap> ?r: if statement
gap> ?r: for loop
gap> ?function
gap> ?1
```

Man beendet das Blättern der Hilfe mit `q` .

**Aufgabe 1.** Unter Zuhilfenahme des GAP-Hilfesystems: Programmiere eine GAP-Funktion `treppe`, die bei Eingabe eine positiven natürlichen Zahl  $n$  die Summe  $\sum_{i=1}^n i$  zurückgibt:

```
gap> List( [1..10], treppe );
[ 1, 3, 6, 10, 15, 21, 28, 36, 45, 55 ]
```

```
(viii) gap> LoadPackage( "RingsForHomalg" );
true
gap> ZZ := HomalgRingOfIntegers( );
Z
gap> IsIntegersForHomalg( ZZ );
true
gap> 1 in ZZ; -1 in ZZ;
true
true
gap> 1/2 in ZZ;
false
gap> QQ := HomalgFieldOfRationals( );
Q
gap> IsRationalsForHomalg( QQ );
true
gap> 1/2 in QQ;
true
gap> mat := HomalgMatrix( [ 1, 2, 3, 4, 5, 6 ], 2, 3, ZZ );
<A 2 x 3 matrix over an internal ring>
gap> R := HomalgRing( mat );
Z
gap> IsIntegersForHomalg( R );
true
gap> IsFieldForHomalg( R );
false
gap> Display( mat );
[ [ 1, 2, 3 ],
  [ 4, 5, 6 ] ]
```

**Aufgabe 2.** Sei  $R \in \{\mathbb{Q}, \mathbb{Z}\}$ . Programmieren Sie eine GAP-Funktion `normalize_pair`, die bei Eingabe einer Spalte  $\begin{pmatrix} a \\ b \end{pmatrix} \in R^{2 \times 1}$  eine Matrix  $U \in \text{GL}_2(R)$  zurückgibt, mit

$$U \begin{pmatrix} a \\ b \end{pmatrix} = \begin{cases} \begin{pmatrix} 0 \\ 0 \end{pmatrix} & a = b = 0, \\ \begin{pmatrix} 0 \\ d \end{pmatrix} & \text{sonst,} \end{cases}$$

mit  $d = 1$  falls  $R = \mathbb{Q}$  bzw.  $d = \text{ggT}(a, b)$  falls  $R = \mathbb{Z}$ .

Hinweis: `?MatElm`, `?HomalgIdentityMatrix`, `?QuoInt`