Tutorial 1. Basics of working with GAP

1. For the 4th SCIEnce Training School, GAP 4.4.12 with packages as on June 29th, 2009 is installed can be loaded using one of the following commands:
`/zvol/gap/gap4r4p12/bin/gap.sh`
`/zvol/gap/gap4r4p12/bin/gap4.sh` (to load GAP from a workspace)

If you are using your own computer, you can install GAP yourself downloading its distribution from the page http://www.gap-system.org/. Note that besides the main archive **gap4r4p12**, it is recommended also to install packages (certain packages will essentially speed up some computations) from the latest archive **packages-<timestamp>**, and you may consider installing some other optional components such as additional tables of marks (**xtom**), tools for package developers (**tools**) and HTML-documentation optimised for MS Internet Explorer (**htmie**).

Detailed installation instructions for UNIX, Mac OS X and Windows could be found on the GAP homepage. Additionally, note that:
- if you are using Mac OS X or higher, it is better to perform UNIX-installation, because in this case you will be able to use packages that require UNIX-like systems;
- in Windows you can also install Cygwin (http://www.cygwin.com) to work in a UNIX-like environment and use some packages that require UNIX (this may require certain adaptation of packages, for which it is better to contact their authors);
- Windows users can also use experimental GAP distribution for Windows (http://www.gap-system.org/ukrgap/wininst/wininst.htm). It consists of two files - one with the core GAP system and optional components, and another with all packages;
- There are other alternative ways to obtain GAP, mentioned on the Downloads page.

2. Now find the GAP root directory (that is, **gap4r4**, or **gap4r4p12** in the RISC installation).

3. The directory **gap4r4/bin** contains scripts **gap.sh** to launch GAP in UNIX/Linux-like systems (thus, in Mac OS X UNIX-installation too) and scripts **gap.bat** and **gaprxvt.bat** to launch GAP in Windows. Additionally, in the RISC installation it contains the script **gap4.sh** to start GAP from a workspace. To start GAP:
- in the Mac OS UNIX-installation, you need first to start **Terminal**. You can open Finder and find it in **Applications/Utilities**
- in Linux systems, sh-scripts can be started immediately from the file manager, e.g. from Conqueror, just by clicking on their pictogram. Alternatively, you can start GAP from the Terminal window like described in (4)
- in Windows, you can launch **gap.bat** file to work with GAP in an MS-DOS window. Alternatively, you can use **gaprxvt.bat** to work in the RXVT shell. You can decide yourself which one is more convenient for you.

From now on, working with GAP looks similarly in all systems. You have the command line with tthe prompt **gap>**, after which you can type your command. You should remember that all commands should be finished with a semicolon (or double semicolon to suppress the output) and that GAP is case-sensitive. To finish your work with GAP you should type the command **quit;** and press ***<Enter>***.

4. Perform now simplest computations, for example, entering commands:

```
352/182;
2*(15+256)/17;
2^64;
2^20000 mod 100;
3 in [1,2,3];
2*2 >= 4;
```

5. One command can be written in several lines, the last line should be finished with a semicolon. Thus, if you forgot to type a semicolon and pressed ***<Enter>***, you can just type semicolon in the next line, and press ***<Enter>*** afterwards. Try to enter the following multi-lines command:

```
155/4545+
1234*5678+
Factorial(100)+
Sum([1..100]);
```

6. Remember that the input is case-sensitive, and that all names of functions are started with the capital letter. For example, the following error is caused by writing the name of the function in lowercase :

```
gap> factorial(100);
Variable: 'factorial' must have a value
gap>
```

7. After some errors you can see intermediate system prompt **brk>**. Here you and investigate local variables and other debugging information. To quit to the upper level, you should enter the command **quit;** (in this case it will not terminate the GAP-session), for example:

```
gap> Factorial(1/2);
Range: <last> must be an integer less than 2^28 (not a
rational) at
return Product( [ 1 .. n ] );
 called from
<function>( <arguments> ) called from read-eval-loop
Entering break read-eval-print loop ...
you can 'quit;' to quit to outer loop, or
you can replace <last> via 'return <last>;' to continue
brk> quit;
gap>
```

8. When the method selection fails because there is no applicable method, an error as in the following example occurs and a break loop is entered:

```
gap> IsNormal(2,2);
Error, no method found! For debugging hints type
?Recovery from NoMethodFound
Error, no 1st choice method found for `IsNormal'
on 2 arguments called from
<function>( <arguments> ) called from read-eval-loop
Entering break read-eval-print loop ...
you can 'quit;' to quit to outer loop, or
you can 'return;' to continue
brk>
```

This only says, that the method selection tried to find a method for IsNormal on two arguments and failed. There are a few functions which can display further information to find out why this happened, see the manual for further details.

9. A very useful feature of GAP is the history of commands. To view the history and scroll commands, press up and down arrow keys. Now type 2 in the command line, and then, viewing the history, you will see only commands that were started with 2.

10. Using left and right arrow keys, you can move the cursor along the command line. You can delete symbols in it using ***<Delete>*** and ***<Backspace>***, and type new symbols. For example, try to type in the command line **F** just after the GAP prompt, find the previously entered line **`Factorial(100)+`** and edit it to compute 500!.

11. For fast moving cursor to the beginning or to the end of the command line, you can use ***Ctrl-A*** and ***Ctrl-B***. In an MS-DOS window ***<Home>*** and ***<End>*** also works. Other useful keyboard shortcuts are listed in the manual (see http://www.gap-system.org/Manuals/doc/htm/ref/CHAP006.htm#SECT008).

12. One of the GAP components is its documentation. It is contained in the **gap4r4/doc** directory and is provided in several formats. The file **gap4r4/doc/htm/index.htm** is the starting page for the manual in the HTML-format. You can create a bookmark in your browser and a shortcut on your desktop. Now open this file, go to the Index, and find how many entries are started from the word "group".

13. You need to know how to select, copy and paste the text in and from the GAP window, using the standard tools for your operating system. Try this first by copying and pasting GAP commands from the manual to the GAP session, and then by copying and pasting the content of the GAP session to some text file or email message.

You should be aware about the convention that within each GAP manual chapter all examples are ordered in such a way that if you will start to type them from the beginning of the chapter, you will obtain the same output like in the manual. If you will try to type an example just from the middle of chapter, it may not work or produce different output.

14. Besides HTML-manuals, there is a way to read the manual just from the GAP session. For example, type **?Factorial** after the GAP prompt (without the semicolon) and you will see the manual entry for this function. Then, type **??Sum** for the search within the manual. When search results will be displayed, type **?1** to see the first of discovered entries, etc.

You can change the viewer that will be used to display search results. For example, in Mac OS X you can enter in GAP the command

```
SetHelpViewer("safari");
```

and next time instead of text-only version of the manual displayed inside the GAP session, you will see an appropriate place of the HTML-version in the browser window.

15. A useful GAP feature is names completion. If already typed initial letters of the name of a variable (including function names) allows to uniquely determine its name, you can press ***<Tab>***, and the name will be completed. If there is no uniqueness, after the second pressing of ***<Tab>*** you will see the list of all possibilities.

For example, after the GAP prompt type **Fib** and press ***<Tab>***. Now compute the 100-th Fibonacci number, stating 100 in round brackets as an argument. Then type **Factori** and press ***<Tab>*** twice to see all manes starting from these letters.

16. You can save your GAP session in a text file. Type the command

```
LogTo("logfile.txt");
```

From now, all inserted commands and their output will be not only displayed on screen but also saved in the file named **logfile.txt** and located in your working directory. Now perform the following computations:

```
n:=20;
a:=2^(n+1)-1;
IsPrime(a);
Factors(a);
x:=n+10;
Factorial(x);
Phi(x);
Sigma(x);
Tau(x);
S:=SymmetricGroup(3);
l:=AsList(S);
t:=MultiplicationTable(l);
Display(t);
IsAbelian(S);
```

Then close the log file entering the command

```
LogTo();
```

and check the content of the log file in a text editor.