IOMax 1.1
User Manual

Klaus Riedl, 2003/2004

IOMax 1.1

less torture, more pleasure

Klaus Riedl

The user manual describes the basic features of the IOMax-software for Axiom PowerTrain bicycle trainer.

Table of contents

1 IOMax

IOMax is a free software that can be used with the Axiom PowerTrain bicycle trainer. You can download the newest version at http://home.mnet-online.de/iomax .

This is just a tiny helpfile, because nobody likes help authoring for a freeware, especially me. I hope this little piece of software is intuitive enought to be described with just a few words.

If you have any suggestions, questions or any other feedback please contact me at iomax@mnet-online.de.

Klaus Riedl

1.1 Features

This software can be used as an alternative driver software for the Axiom PowerTrain bicycle trainer. The motivation for the implementation is to store the results in a database (that makes reporting end evaluation easyer) and to get some additional indicators.
So what do we have right now?

- The data of riders and tour history are now stored in an access- database (`iomax.mdb` in the database folder). The installer contains a JetEngine 3.6 database for Access database format. So you don 't need a ms-office on your computer, but if you have a license, you can easily generate new reports on the test dataset (table `history`). Please use the access frontend (or any other ODBC- datebase frontend) to edit rider data or add new riders (table `rider` in the `iomax.mdb`).
- You can store the the whole data of one tour in a file (`result.ini` in the result folder). The file contains complete time-, speed- und pulse- and power- history of a single tour. To generate this file please check the store binary menu item in the options menu.
- The course- file format has changed. The original files (binary, *.crs) are now converted to a text file format. The syntax of the text files is much like ini- files (so is the extension). The new course files can contain aditional informations like comments or pictures (have a look on the multimedia page). To edit course files just take a simple text editor like notepad and change the ini- file. To create a new course write an ini- file and store it in the course- folder. A conversion program (`*.crs` to `*.ini`) is also included.

- The software has some additional indicators for ascending speed, absolut altitude etc.
- You can use over 4100 ciclomaniac profiles with the iomax. For more information have a look on the import chapter.
- Import your Axiom PowerTrain testing data into the the iomax database. For more information have a look on the import chapter.

1.2 Import

You can import data of differet applications and types.

Course Files:
Cyclomaniac xml- files
Axiom course files

Training Results:
Axiom

1.2.1 Ciclomaniac

On the ciclomaniac page * you can download more then 4100 course files. The course- files are in xml- format. The specification for CicloManiacXML Profiles that corresponds to this DTD is available at: http://www.ciclomaniac.com/ciclomaniacxml/ciclomaniac.dtd *.

If you want to use ciclomaniac profiles with the iomax you have to make two steps:

- Download the specific profile. On the cyclomaniac web page you can find a text box and a search button. Enter the name of the profil (e.g. "pordoi") in the text box and press the `<search>` button. A table appears with a xml- link in the right coulmn. Select the xml- link. Now the content of the xml- file appears in your browser. Select the *`file/save as`* menu of your browser and store the xml- file on your computer.
- Open the iomax- application and select the *`file/import/ciclomaniac`* - menu. A file select popup appears on the screen. Select the xml- file and press <OK>. Now an iomax- course file ist created in the course directory and can be used with iomax.

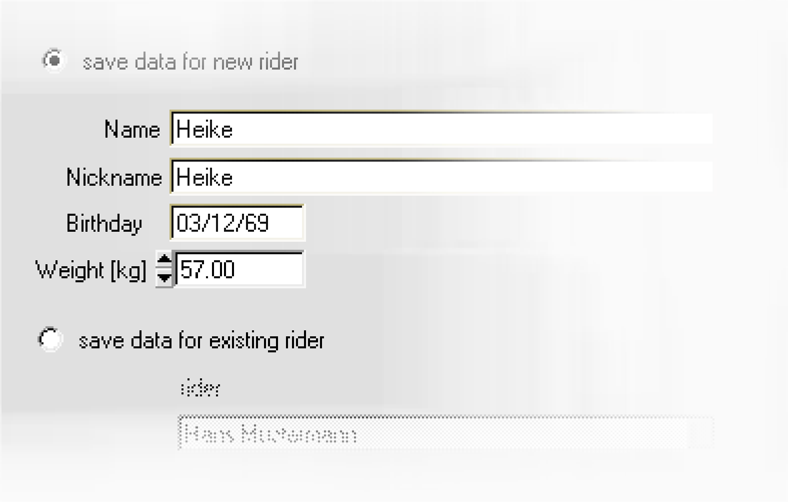
* under CicloManiac Concession

1.2.2 Axiom

From your axiom application you can import course files (*.crs), rider profiles (*.cfg) and training test files (*.prf).

To import axiom course files
- Select the *`file/import/axiom/course file (crs) ...`* menu in the iomax application. A file select popup appears.
- Select a course file and press <OK>. Now an iomax- course file ist created in the course directory and can be used with iomax.

To import rider profiles and training test files to the iomax database
- select the *`file/import/axiom/rider and data (cfg/prf) ...`* menu in the iomax application. A path select popup appears.
- select an axiom rider profile path. The directory must contain a rider profile (*.cfg) and some training test files (*.prf). Now a dialog appears that shows a rider profile.



● save data for new rider

Name: Heike

Nickname: Heike

Birthday: 03/12/69

Weight [kg]: 57.00

○ save data for existing rider

rider: Hans Mustermann

- The Dialog shows the rider profile that was stored in the axiom folder. You can edit the rider data or select an existing rider.
- If you select <save data for new rider> and press <OK>, a new rider entry is created in the database and the testing files are stored for the new rider.
- If you select <save data for existing rider> and press <OK> the testing files are stored for your selection of an existing rider.

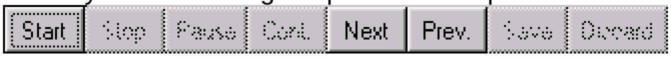
1.2.3 HAC4

To import HAC4- files into iomax you have to make two steps:

- Select the `file/import/HAC4/course file (tur) ...` menu in the iomax application. A file select popup appears.
- Select a HAC4- course file and press <OK>. Now an iomax- course file ist created in the course directory and can be used with iomax.

1.3 Keys

The keys are on the right top of the main panel.



You can select a key with mouse, computer- keyboard or with the tab-key of the navigator box on your bicycle. If you press the enter- key of the navigator box, the *selected* key in the program is pressed.

Start
Starts the selected tour. This key changes from *passive*- mode to *active*- mode. This Key is only active in *passive*- mode, if serial communication is ok and if a course is selected.

Stop
Break the actual tour. This key changes from active- mode to ready- mode. To return from *ready*- mode to the *passive*- mode you have to press Save (to save last data set) or Discard. This Key is only active in *running*- mode.

Pause
Interrupt the active tour. This key changes from *running*- mode to *pause*- mode. To continue the tour (return to running- mode) you have to press the Continue- key. This Key is only active in *running*- mode.

Continue

This key continues an interrupted tour (return to *running*- mode). This Key is only active in *pause*-mode.

Prev.
Show previous page. This key is always active.

Next
Show next page. This key is always active.

Save
Save the data of the last tour and return to *passive*- mode. This Key is only active in *ready*- mode.

Discard
Discard data of the last tour and return to *passive*- mode. This Key is only active in *ready*- mode.

1.4 Pages

With the tabs in the left bottom of the main page you can select the different pages of the program.

- Skalar display shows most of the skalar values
- LabStyle contains lab controls
- Grafik contains strip- charts and a tour- profile
- Picture- Page shows bitmaps

1.5 Connection

The connection indicator (right top) shows the state of the serial communication. If the display is **green**, the communication is OK, otherwise (**red**) the communication does not work. Possible reasons are:

- the navigator box is not connected with the computer
- the navigator box is connected to the wrong serial port. To solve the problem select another com port (menu options/com), change COM- Registry setting or connect the cable to another comport.

1.6 Registry

Root: HKEY_LOCAL_MACHINE/Software/Bike/IOMax
Keys:

- Com: Comport settings
- Path: Recent Paths
- Rider: Last rider ID
- Data: Store tour data file

1.7 System Requirements

The software is tested on a scrap iron (266 MHz, Pentium II) with different OSs:

- WinNT 4, SP6
- WinNT 4 Server, SP6
- Win 2000, SP3

- Win XP

Best display resolution is 1024*768

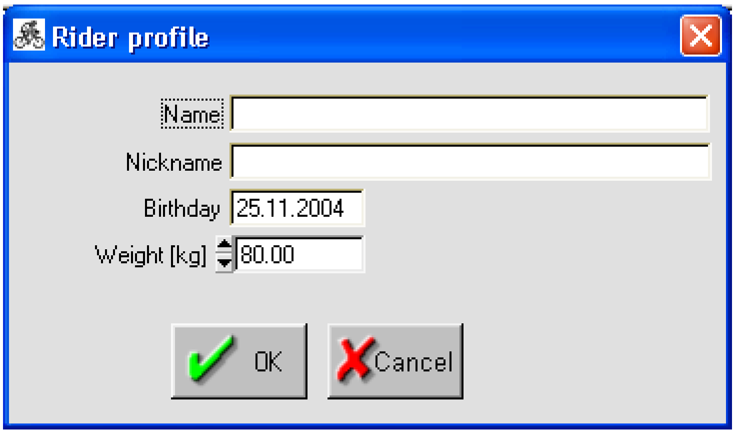
1.8 Riders

To manage the user data you have the following choices:

- add rider
- edit data oft the current rider
- select current rider

1.8.1 Add

To open the rider select dialog open the rider/add... - menu.



Rider profile

Name

Nickname

Birthday 25.11.2004

Weight [kg] 80.00

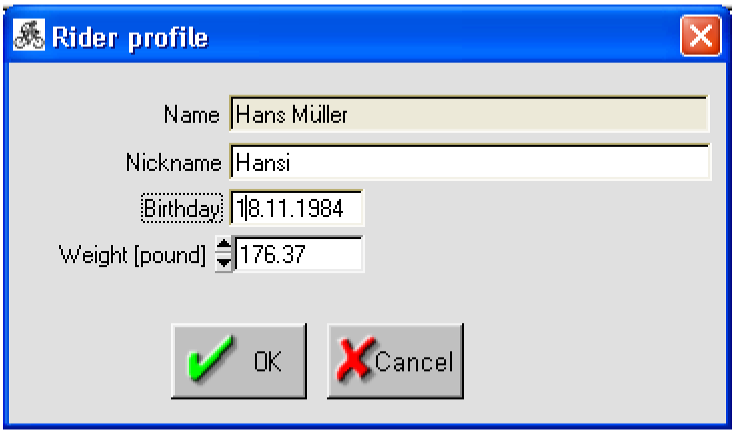
In the dialog you can enter the data of a new rider:

Item	Description	Example
name	full name of the rider	Hans Müller
nickname	rider nickname	Hansi
birthday	rider birthday. Please use the date format selected in the system settings of your computer	31.1.1984
weight	rider weight You can change the unit of this item with the units- dialog.	no lies, just the truth!

When you press <OK> a new rider with your data is created in the database and the new rider is selected as current rider.

1.8.2 Edit

To open the rider select dialog open the rider/edit... - menu.



Name: Hans Müller

Nickname: Hansi

Birthday: 18.11.1984

Weight [pound]: 176.37

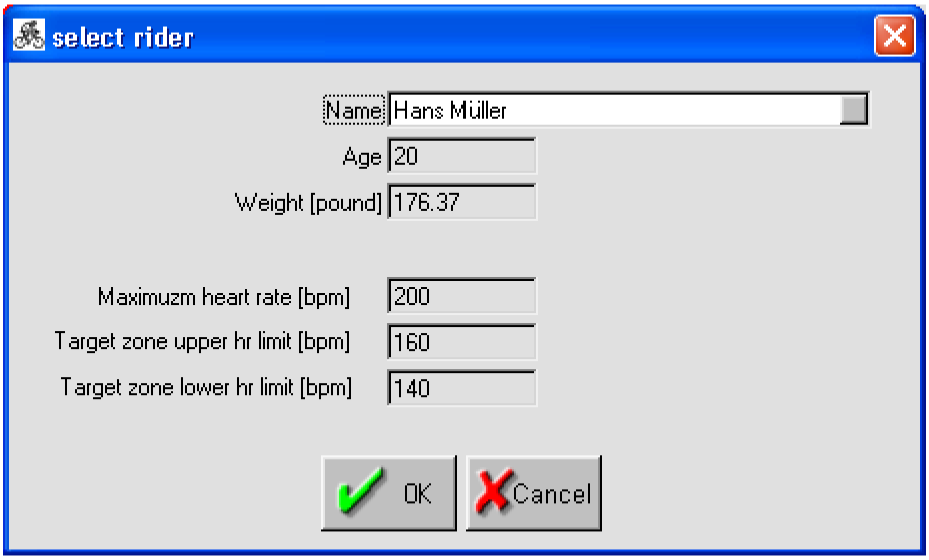
In the dialog you can change the data of the current rider except the rider name:

Item	**Description**	**Example**
nickname	rider nickname	Hansi
birthday	rider birthday. Please use the date format selected in the system settings of your computer	31.1.1984
weight	rider weight You can change the unit of this item with the units- dialog.	don't fib, contact your weighing machine

When you press <OK> the rider data are updated in the database.

1.8.3 Select

To open the rider select dialog open the rider/select... - menu.



Name Hans Müller

Age 20

Weight [pound] 176.37

Maximuzm heart rate [bpm] 200

Target zone upper hr limit [bpm] 160

Target zone lower hr limit [bpm] 140

With the name- combobox you can select a rider. Press <OK> to make your selection current.

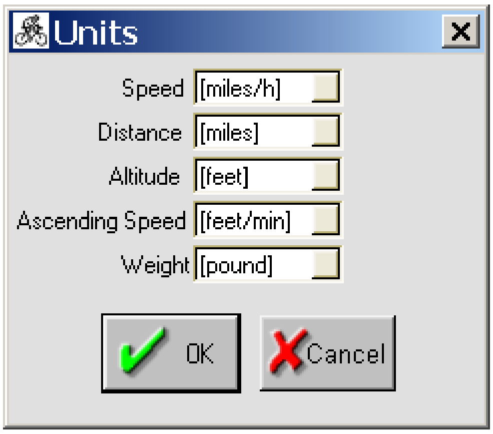
1.9 Options

You can change the following options of IOMax:
- Units
- Paths
- Comport

1.9.1 Units

To open the units- dialog select the options/units... - menu.

With the units dialog you can select units of
- speed ([km/h], [miles/h])
- ascending speed ([m/s], [feet/s], [km/h], [miles/h])
- distance ([km], [miles])
- altitude/elevation ([m], [feet])
- weight ([kg], [pund])



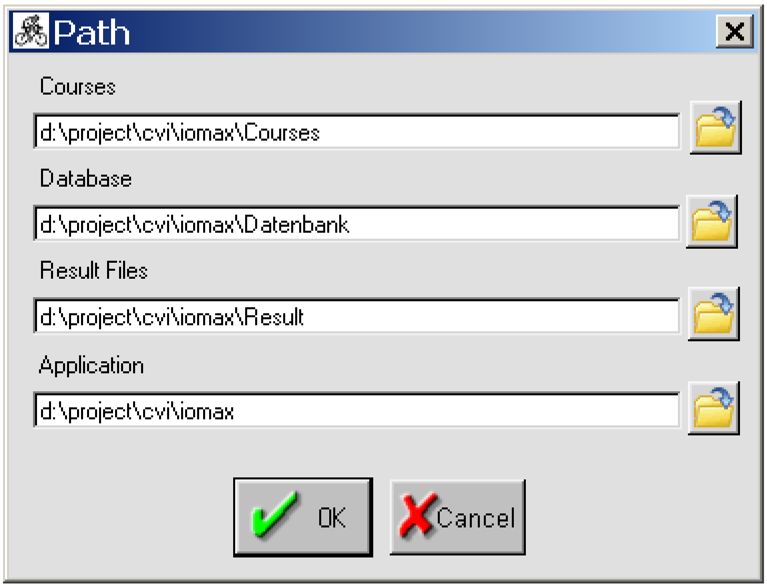
Speed [miles/h]
Distance [miles]
Altitude [feet]
Ascending Speed [feet/min]
Weight [pound]

Your selection of units is stored when you press <OK>

These units are used for any kind of visualisation like online displays and reports but in general all data are stored in metric format.

1.9.2 Paths

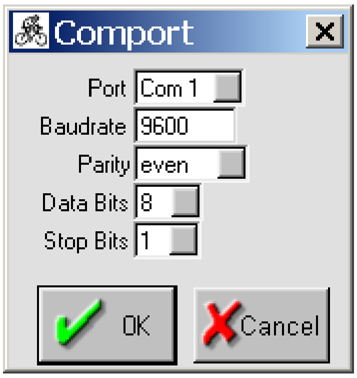
To open the units- dialog select the options/paths ... - menu.



Courses
d:\project\cvi\iomax\Courses
Database
d:\project\cvi\iomax\Datenbank
Result Files
d:\project\cvi\iomax\Result
Application
d:\project\cvi\iomax

1.9.3 Comport

To open the comport- dialog select the options/comport... - menu.



Port Com 1

Baudrate 9600

Parity even

Data Bits 8

Stop Bits 1

1.10 Reports

1.10.1 History by date

To open the *History By Date* select the reports/history by date... - menu.
The report *history by date* shows a complete history for the current user, that is a list of all rides ordered by date.

The list contains information about:

- Date
- Course
- Time
- Dist
- Elevation
- AvgSpeed
- AvgHR
- AvgPerf

At the bottom of the list you can find a column statistic with sum, average and maximum of the column.
All informations are displayed in the selected units.

1.10.2 Course History

To open the *Course History* select the reports/course history... - menu.
The report *course history* shows a complete history for the current user on the current course ordered by date.

The list contains information about:

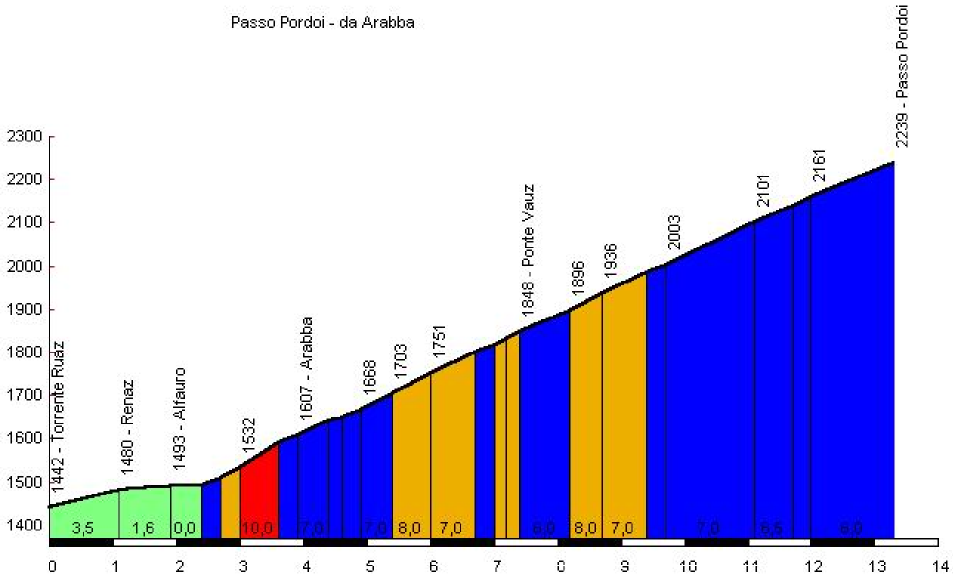
- Date
- Time
- AvgSpeed
- AvgHR
- AvgPower
- AvgPerf

At the bottom of the list you can find a column statistic with sum, average and maximum of the column.
All informations are displayed in the selected units.

1.11 Courses

Courses describe the profile of a section of road. You can find a very complete list of european passes in the internet under Archivo Salite d'Europa . Other sources are the Portus Index or The Mountain Site.
The ciclomaniac page offers more then 4000 course files in xml- format for download. These files can be imported into the iomax- format. Have a look on the import chapter.

Below is a sample profil from Archivo Salite d'Europa ...



To create a new course for the IOMax- software you have to write a new course file and store it in the IOMax- courses- folder. The next chapters describe the course file format and the used ini syntax.

1.11.1 File Format

Course files are text files with an ini- syntax (not case sensitive!). Course files consist of one header section and a arbitrary number of step sections. The steps contain the course profile (distance/altitude) and additional information (picture/comment). The order of the steps does not matter because they are sorted by distance after reading.

Sample:

```
[Header]
  Name="Test01"
  Author="Klaus Riedl"
  File="Test01.ini"
  Steps=2
[Step]
  dist=0.000
  alt=500
  comment="Start"
  Pict="picture\\test\\pc00.jpg"
[Step]
  dist=0.1
  alt=500
  comment="Anfang der Steigung"
  Pict="picture\\test\\pc01.jpg"
[Step]
  dist=0.2
  alt=501
  comment="jetzt wirds steiler"
  Pict="picture\\test\\pc07.jpg"
[Step]
  dist=0.3
  alt=506
  comment="wir sind übern Berg, auf zum Schlußsprint"
  Pict="picture\\test\\pc02.jpg"
[Step]
  dist=0.4
  alt=506
  comment="Zielankunft"
```

1.11.1.1 Header

Items:

item	description	type	option
name	name of the course. This name should be unique. It is used to identify the coures in the database.	string	mandatory
type	type of course file	string	optional
author	author name	string	optional
file	file name	string	optional
steps	number of profile steps	integer	optional
description	description text for the course	string	optional
origin	starting point of the tour	string	optional
city	city next to the tour	string	optional
region	region of the tour	string	optional
country	country of the tour	string	optional
language	language of the description text	string	optional
eMail	mail address of the author	string	optional
URL	web page of the author	string	optional
Phone	phone number of the author	string	optional
PicFlags	information how to display the picture: • 0: center picture, no resize • 1: size picture to control Default value is 0.	int	optional

Sample:

```
[Header]
  Name="Test01"
  Author="Klaus Riedl"
  File="Test01.ini"
  Steps=5
  Description="Ein Test für die course- Dateien\n"
              "Hier wird alles mal ausprobiert"
```

1.11.1.2 Step

Items:

item	description	type	option
dist	distance in [km]	float	mandatory
alt	altitude in [m]	float	mandatory*
comment	comment text. This text is intended to describe a special point or range of your course. The text is displayed in the comment text box.	string	optional
pict	picture file name, relativ to course-directory. Picture files may be of the following types: jpg, png, bmp, dib, wmf, pcx	string	optional
picflags	information how to display the picture: • 0: center picture, no resize • 1: size picture to control If this item is missing, the picflags of the header section are used. If picflags are missing in the header too, a default value of 0 is taken.	int	optional
label	label in the map	string	option

* If the altitude is missing in a step it is calculated automatically. There has to be at least one step with an altitude item.

Sample:

```
[Step]
  dist=15.3
  alt=506
  comment="wir sind übern Berg, auf zum Schlußsprint"
  pict="picture\\test\\pc02.jpg"
```

1.11.2 ini-syntax

Syntax:

```
  ini-file
    section-list

  section-list
    section-definition opt section-list

  section-definition
    [ name ] item-list

  item-list
    item-definition opt item-list

  item-definition
    name = literal
```

Remark:
The syntax is not case sensitive!

The syntax may contain comments. The comment syntax is like in ANSI- C:

- A single line comment starts with `//` and ends at `<new line>`
- A multi line comment starts with `/*` and ends at `*/`

Sample:

```
[Header]
  Name="Test01"
  Author="Klaus Riedl"
  File="Test01.ini"

//Strings können auch über mehrere Zeilen geschrieben werden
  Description="Ein Test für die course- Dateien\n"
              "Hier wird alles mal ausprobiert"

/*
  Der Item 'Steps' dient ausschließlich zur Information
  Er sollte die Anzahl der Stützpunkte im Höhenprofil angeben
  Der Item kann fehlen oder falsch sein
*/
  Steps=2

/*
  Ab hier folgen die Stützpunkte:
  Die Liste wird beim Einlesen vom Programm nach der Entfernung (Dist) sortiert,
  d.h. die Stützpunkte können in beliebiger Reihenfolge aufgeschrieben werden.
*/
[Step]
  dist=0.000
  alt=500
  comment="Start"
  Pict="picture\\test\\pc00.jpg"
```

1.11.2.1 section-definition

item-definition

Syntax:

```
  section-definition
    [ name ] item-list

  item-list
    item-definition_opt item-list
```

Description:

Sets the values of a section.

Sample:

```
[Header]
  Name="Test01"
  Author="Klaus Riedl"
  File="Test01.ini"
  Steps=2
```

1.11.2.2 item-definition

item-definition

Syntax:

```
 item-definition
    name = literal
```

Descrioption

Sets the value of an item. The literal must hav the same type as the item.

Sample:

```
dist = 1.5
```

1.11.2.3 constants

Syntax:

```
literal
  string-literal
  floating-point-constant
  character-constant
  integer-constant
  oct-hex-escape
```

1.11.2.3.1 string-literal

string literal

Syntax:

```
string-literal :
    " s-char-sequenceopt" L" s-char-sequenceopt"

s-char-sequence :
    s-char
    s-char-sequence s-char

s-char :
    any member of the source character set except the double
    quotation mark ("), backslash (\), or newline character

escape-sequence
```

Description:

A string literal consists of zero or more characters from the source character set surrounded by double quotation marks ("). A string literal represents a sequence of characters that, taken together, form a null-terminated string.

Because the double quotation mark (") encloses strings, use the escape sequence (\") to represent enclosed double quotation marks. The single quotation mark (') can be represented without an escape sequence. The backslash character (\) is a line-continuation character when placed at the end of a line. If you want a backslash character to appear within a string, you must type two backslashes (\\).

Strings may contain escape-sequences. For example

```
"\t\t\"Name\"\\\tAddresse\n\n"
```

generates the output:

```
"Name"\       Addresse
```

1.11.2.3.2 character-constant

character-constant

Syntax:

```
character-constant
  'c-character'

c-character
  all chars except abostroph( ' ), Backslash( \ ) und Newline.
  escape-Sequenz
```

```
escape-sequence
  simple-escape-sequence
  octal-escape-sequence
  hexadecimal-escape-sequence

simple-escape-sequence
  one of
  \' \" \? \\ \a \b \f \n \r \t \v

octal-escape-sequence
  \octal-digit
  \octal-digit octal-digit
  \octal-digit octal-digit octal-digit

hexadecimal-escape-sequence
  \xhexadecimal-digit
  hexadecimal-escape-sequence hexadecimal-digit
```

Sample:

```
c='X'  //normal character
c='\r' //Escape-Sequence for carriage return
```

Escape-Sequenz	**ASCII- Wert**	**Zeichen**
\\	92	backslash
\?	63	questionmark
\'	39	inverted comma
\"	34	quotation mark
\0	0	Null
\n	10	newline
\t	9	horizontaler Tab
\v	11	vertikaler Tab
\b	8	backspace
\r	13	carriage return
\f	12	formfeed
\a	7	alert

1.11.2.3.3 integer-constant

integer-constant

Syntax:

```
integer-constant
  decimal-constant integer-suffixopt
  octal-constant integer-suffixopt
  hexadecimal-constant integer-suffixopt

decimal-constant
  digit1-9
  decimal-constant digit

octal-constant:
  0
  octal-constante octal-digit

hexadecimal-constant:
  0x   hexadecimal-digit
  0X   hexadecimal-digit
  hexadecimal-constant hexadecimal-digit

digit 1-9
  1 2 3 4 5 6 7 8 9

octal-digit
  0 1 2 3 4 5 6 7

hexadecimal-digit
  0 1 2 3 4 5 6 7 8 9
  a b c d e f
  A B C D E F

integer-Suffix
  l L
```

Sample:

```
  i=157     //dezimal-constant
  j=0365    //oktal-constant
  k=0192    //error in oktal-constant because of '9'
  l=0x3fff  //hexadezimal-constan
```

1.11.2.3.4 oct_hex_escape

Syntax:

```
octal-escape-sequence:
  \ octal-digit
  \ octal-digit octal-digit
  \ octal-digit octal-digit octal-digit

hexadecimal-escape-sequence:
  \x hexadecimal-digit
  hexadecimal-escape-sequence hexadecimal-digit
```

Description:

Octal escape sequences, specified in the form *ooo*, consist of a backslash and one, two, or three octal characters. Hexadecimal escape sequences, specified in the form \x*hhh*, consist of the characters \x followed by a sequence of hexadecimal digits. Unlike octal escape constants, there is no limit on the number of hexadecimal digits in an escape sequence.
Octal escape sequences are terminated by the first character that is not an octal digit, or when three characters are seen.
For example:

```
och = '\076a';  // Sequence terminates at a
ch = '\233';     // Sequence terminates after 3 characters
```

1.11.2.3.5 floating-point-constant

floating-point-constant

Syntax

```
floating-point-constant:
     fractional-constant exponent-partopt floating-suffixopt
     digit-sequence exponent-part floating-suffixopt

fractional-constant:
    digit-sequenceopt.  digit-sequence
    digit-sequence .

exponent-part:
    e  signopt digit-sequence
    E  signopt digit-sequence

sign: one of
    + –

digit-sequence
    digit
    digit-sequence digit

floating-suffix: one of
    f l F L
```

Description:
Floating-point constants specify values that must have a fractional part. These values contain decimal points (**.**) and can contain exponents.

Floating-point constants have a "mantissa," which specifies the value of the number, an "exponent," which specifies the magnitude of the number, and an optional suffix that specifies the constant's type. The mantissa is specified as a sequence of digits followed by a period, followed by an optional sequence of digits representing the fractional part of the number. For example:

```
18.46
38.
```

The exponent, if present, specifies the magnitude of the number as a power of 10, as shown in the following example:

```
18.46e0      // 18.46
18.46e1      // 184.6
```

If an exponent is present, the trailing decimal point is unnecessary in whole numbers such as 18E0.

1.11.3 Fiets index

There are several ways in which to classify the difficulty of mountains. Based on a formula used in the Dutch magazine Fiets

FIETS-index = [H^2 / D*10] + (T - 1000):1000

H = difference in height
D = distance in meters
T = top of mountain in meters
[The second part of the formula does only apply to mountains above 1000 meters]

1.12 Versions

Version	Date	Description
1.0.0.0	1.1.2004	First release of IOMax
1.1.0.0	9.3.2004	Extended user interface
1.1.0.2	13.4.2004	Bugfix slope calculation, xml- file import
1.1.0.9	18.11.2004	Selectable units: ⇨Menu *options/units...*
1.1.0.10	29.11.2004	User management included: ⇨Menu *rider/add ...* ⇨Menu *rider/edit ...* Reports included: ⇨Menu *reports/History by date ...* ⇨Menu *reports/Course history ...* Registry information moved to ⇨HKEY_CURRENT_USER\Software\IOMax
1.1.0.11	4.2.2005	HAC4- import included: ⇨Menu file/import/HAC4/course file (tur) ... Bugfix with selectable units

Index