

Chaos

The Chess HAppening Organisation System

Version 5.3

Jochen Wiedmann

Copyright ©1993 Jochen Wiedmann

Am Eisteich 9

72555 Metzingen (Deutschland)

Tel. 07123 / 14881

Internet: wiedmann@mailserv.zdv.uni-tuebingen.de

Permission is granted to make and distribute verbatim and modified copies of this manual and the program Chaos following the terms of the “GNU General Public License” provided the copyright notice and this permission notice are preserved on all copies and the “GNU General Public License” (in the file ‘COPYING’) is distributed as well.

The author gives **absolutely no** warranty that the program described in this documentation and the results produced by it are correct. The author cannot be held responsible for **any** damage resulting from the use of this software.

This software, the source (or parts of it) may explicitly be used on machines that are used to do research, development, construction, tests or production of arms and other military equipment. (Make chess, not war! ;-)

1 Survey

Chaos V5.3 is written completely in C and compiled using DICE V2.07.54-3.0. It manages swiss pairing (see Subsection 9.2 [Swiss pairing], page 19) as well as Round-Robin tournaments (see Subsection 9.1 [Round-Robin mode], page 18) and computes the DWZ's (German rating number) resulting from the tournament. It follows the rules of the *Turnierleiterhandbuch des Deutschen Schachbundes* (The official german chess federation's guide to managing chess tournaments). These rules may part in detail from the FIDE-rules (especially on managing the swiss pairing), but should conform in general. The algorithms should allow to do the pairings in a way satisfying everyone. Chaos is fast and therefore also suitable for tournaments with 15-minute or 5-minute games.

When having an odd number of players, Chaos selects one player, who receives a full point bye in the Swiss pairing mode. You are able to add players after drawing the first round and mark players that drop out. (Swiss pairing only).

All output of data (sorry: nearly all) goes to screen or printer, as the user decides. Tables are sorted by points only, by Buchholz, extended Buchholz or Sonneborn-Berger (see Subsection 6.5 [Preferences], page 12). The program creates tables of progress or cross-tables (the last one as T_EX-source only). All is done with a standard-Intuition GUI (Graphical-User-Interface), which should be usable straight forward. (This is no excuse for not reading this documentation, especially the section about precautions. see Subsection 3.2 [Precautions], page 3)

The number of players is limited only by RAM (approximately 3 KBytes of RAM per player are needed).

2 Installation

Chaos is developed using Kickstart 3.0 and should run on any Amiga using Kickstart 2.0 or higher. Chaos uses the `locale.library`, but supports other languages without it! Bad luck for people still using Kickstart 1.3 or lower — and another reason to upgrade. (Believe me: 2.x is a wonderful thing for programmers and users too.)

What Chaos needs is the `mui.library`, version 2.0 or newer. MUI (**M**agic **u**ser **i**nterface) is shareware and hence freely distributable; you find it for example on Aminet (Directory `'/pub/aminet/dev/gui'`) or on the Fish CDs. I admit, that using a shareware program as part of a really free program is a bit of contradiction. But the advantages of using MUI seemed overwhelming to me, MUI is cheap (15\$ for the registration) and after all: In principle it's possible, to use MUI without being registered. However, I strongly encourage any Chaos user to become registered.

A hard-drive is not required, but very useful (as always) and installing Chaos *and* MUI on one floppy disk could make problems. (I did not try it.)

Installation is rather simple: Just unpack the file `'Chaos.lha'`. (I suppose, you have done right now. :-) If not, use the following command:

```
lha x Chaos5.3bin.lha
```

(You will find LhA on the fish-disks for example.) If you want, you can unpack the sources (in `'Chaos5.3src.lha'`) as well.

The installation is finished now! But possibly you don't like to have a special Chaos-directory? To perform the installation you can use the Install-Skript in that case by simply double-clicking the Install icon. When running it in Novice-Mode it simply copies the program to `'c:'`, the default-icon to `'s:'` and the Catalogs you wish to `Locale:Catalogs`. The skript needs the program `Installer` which is part of the OS since Workbench 2.1. There is a freely distributable version of this program which you should find on the Fish-disks too or on Aminet (directory `'/pub/aminet/util/misc'`).

What the installer script does: When using `locale.library`, Chaos needs to find the Directory `'Catalogs'`. This should be a subdirectory of the directory from which the program is started or might be copied to `'locale:'` using the command

```
copy Catalogs locale:Catalogs all
```

Without `locale.library` you have to use the built in english language. Currently there is only a german catalogfile. The file `'src/NewCatalog.ct'` can be used to create new catalogfiles. (Please send them to me. I will include them in new versions of the program.) Note, that Chaos is able to use catalogs on Workbench 2.0! Possibly you have to create a directory `'Locale:catalogs'` in that case.

Another subdirectory should be called `'s'` containing the file `'Chaos_project.info'`, which is the project-icon, that tournaments will receive when being saved by Chaos, and the file `'Chaos_Menu'` which is required for the `Special` menu. It might be changed for example using `IconEd`. Another place to install this could be `'s:'`. In that case execute

```
copy s s: all
```

and remove the subdirectory `'s'`.

There is a subdirectory `'Rexx'` containing some example scripts. These show how to use Chaos from an ARexx program. Probably you wish to copy these as well.

It should be noted that Chaos uses some special characters which are not included in all fonts. Please ensure that your Workbench uses fonts which contain all 256 Amiga-characters!

The program is started from either CLI or workbench. One thing is important: Chaos uses algorithms working in a recursive way and needs very much stack therefore! Suggested are 50 Bytes per player and 4000 bytes for the system. A stack of 20000 bytes should be sufficient and is set as default in the Chaos-Icon. The stacksize is changed using the Info-menuitem in the workbench

screen or using the CLI-command `stack` depending on how you start Chaos. The latter should be done by inserting the following line in `'s:Shell-Startup'` or executing the line manually everytime before starting Chaos:

```
stack 20000
```

3 Management

This chapter contains some general hints and is not required to use the program. But you should take a look at the section about precautions.

3.1 Preparations

There are some additional problems and questions (besides the usual things like getting a room, the chess-sets and so on) when using a computer in a chess-tournament. You should check these:

- How many players will participate in the tournament? If you are expecting more than say 20, there will be a large number of people which need to be registered all at once. Note that the computer will accept input only from *one* assistant. This might result in an annoying delay. Try to be informed about the players in advance and encourage pre-registrations.
- Have a look at the playing-site: Is there a second room for the computer? If not: An ink- or laser-printer might be useful. (The sound of pin-writers is awful!) Do you have enough power plugs?
- If you are using another printer than your own, you should take a look at the printer-drivers. Which one could be used? And remember setting the preferences in the right way.
- Make sure that nobody unauthorized has access to the computer. (I have had very bad experiences with children.)

3.2 Precautions

Did you ever have contact with Murphy's Law? If not, drop this section: You won't in the future. Otherwise read on: What happens in case of a hardware-failure or a software-bug? No problem, if you are prepared!

Always having a backup of the program and all data should be your first principle. Pairings are always (at least a little bit) randomized, so make a backup after *each* pairing. Even better: Use different files for each round. (You might detect for example that the result of a game was entered incorrectly, while the pairings for the next round are already finished and saved. By loading the previous copy you can change the result and repeat the drawing with the correct result.)

Additionally you should have contact to another Amiga-user near you... So hardware-failures aren't problems.

But there might be a software-bug. :-((Please do not be alarmed that the author considers this, see it as a strength of the program! :-) You have to manage the tournament by hand in that case. The program is able to create the player-cards you need using the menuitem `Player-cards`. (Actually the program creates `TEX`-source. The output of `TEX` isn't very fast. I would like to have suggestions for doing this another way.) See Subsection 6.4 [Player-cards], page 11.

4 Starting the program

The program is started from either CLI or Workbench by double-clicking the program's icon or a project icon created by Chaos. The command-template of Chaos look's like the following:

```
Chaos FILE,MAKEICONS/S,DEFICON/K,LANGUAGE/K,NOWINDOW/S
```

This is what the arguments mean:

FILE is the name of a tournament-file you want to load. The program is initialized to a new tournament without name, if **FILE** is omitted. You should begin by selecting the menuitem `Project-New` in that case.

MAKEICONS forces Chaos to create Icons when saving tournaments. When started from CLI, Chaos will not create Icons otherwise.

DEFICON is the name of the icon, Chaos should use. Default is `'s/Chaos_Project'`. Please notice, that the ending `'.info'` is added automatically by `icon.library!` If the given Icon isn't found, Chaos try's to load `'s:Chaos_Project'`. If this is isn't present either, Chaos will use the Kickstart's default-icon. See Section 2 [Installation], page 1.

MENUFILE is the name of a file containing own menuitems which should appear in a menu called `Special`. (Default is `'s/Chaos_Menu'`.) See Subsection 8.2 [Own Menuitems], page 17.

LANGUAGE is the name of the language you wish to use. This is for users of Workbench 2.0 only. Don't use this when running Workbench 2.1 or higher! Default is english. Of course other languages require the installation of the catalog files.

NOWINDOW suppresses the program's window. This makes of course no sense, except if you use the program in batch mode from an `ARexx` script.

WINNERPOINTS
DRAWPOINTS

number of points that winning a game and draws give, respectively. Note, that these will be divided by two. See Subsection 5.1 [tournament-window], page 5.

The Workbench-Tooltypes are the same except for **MAKEICONS**: The default is to create Icons, when started from Workbench, so this command isn't needed. In that case you might use the Tooltype **NOICONS**. (Please notice, that the Project-icon's Tooltypes overwrite those of the Tool-icon.) **NOWINDOW** isn't allowed as tooltype. I don't see any sense in running the program without window from the workbench.

5 The program's input-windows

Input of data is done using standard-Intuition-gadgets. You should be able to work with it immediately. However, some principles should be noted:

- All windows have a CloseWindow- and a Cancel-gadget. You might cancel the program's actions everywhere. The **ESC** key might be used alternatively.
- All windows receiving input have an **OK**-gadget and are terminated using this. Additionally you can use the keys **CTRL RETURN** to terminate a window. However, this will not work if a string gadget is activated. A workaround is to set **CTRL RETURN** as **Gadget Off** key in the MUI preferences. This makes it possible to terminate the window by pressing **CTRL RETURN** twice.
- Some windows have string gadgets. The first string gadget is always activated, when the window appears. All text you enter will be received by that string-gadget, until you press one of the following keys:
 1. The **TAB**- and the **RETURN**-key will activate the next string-gadget.
 2. The **SHIFT-TAB** key will activate the previous string-gadget.
- MUI makes it possible to work without mouse. For example pressing the Tab key always activates the next gadget and Shift-Tab the previous. Listview gadgets are controlled by the cursor keys and the Return key can select a button like the Ok gadget. Note, that the keys can be changed in the MUI preferences. Please read details in the MUI documentation.

5.1 The Input-of-tournament-data-window

This window appears, when selecting the menuitem **Project-New**. See Subsection 6.1 [Project], page 8. The window contains a string-gadget, in which the tournament-name should be entered. This is the name which will appear on the main-window and in the printer-headlines. If you have different groups each playing their own tournament, you should use different names, for example "1st Metzinger Open (Women)" and "1st Metzinger Open (Men)".

Below of the tournament name you find gadgets where you may enter the points for winning a game and draws. They correspond directly to the **WINNERPOINTS** and **DRAWERPOINTS** arguments when starting the program. See Section 4 [Startup], page 4. These numbers are divided by two, thus if you want the usual 1 point for a win and a half point for a draw, you have to enter 2 and 1

point instead. (These are the defaults.) The idea is to allow something like 3 points for a win and 1 point for a draw. (Remember, that you would have to enter 6 and 2 instead.)

5.2 The Player-selection-window

This window is opened from the menuitems `Players-Modify` and `Players-Delete`. See Subsection 6.2 [Players], page 9. It shows a list of all players, which can be scrolled with the usual Arrow- and Scrollbar-Gadgets. You can select one player by clicking on his name or more than one by clicking on their names while holding the shift key pressed. Note, that it's just like selecting Workbench icons. The window is terminated by pressing the upper-left button of the window. (`Modify`, for example)

5.3 The Input-of-player-data-window

When selecting one of the menu-items `Players-Add` or `Players-Modify`, this window will appear. See Subsection 6.2 [Players], page 9. It uses the following gadgets:

Name The player's name is used, when pairing the first round. The players are sorted alphabetically by the program, therefore their names need to be entered in a uniform way. Suggested is last name, comma, first name, for example "Wiedmann, Jochen". See Subsection 9.2 [Swiss pairing], page 19.

You *must* enter a player's name and it must be unique! The program doesn't accept clicking the OK-gadget without a name! If needed, use something like 'Miller, Joe (2)'.
 You *must* enter a player's name and it must be unique! The program doesn't accept clicking the OK-gadget without a name! If needed, use something like 'Miller, Joe (2)'.

Home address

Phone The player's addresses will not be needed in most cases. Of course they might be useful, if a tournament goes on for an extended period of time.

Chess-club

This will not be needed too in most cases. But you might wish to include this in the list of players.

Birthday A players birthday is needed only when computing the DWZ-ratings. It *must* be entered as dd.mm.yy or dd.mm.yyyy, where dd means day, mm means month and yy Year. I would like to use the excellent DOS-routine StrToDate, which might use the locale's date format instead of my own. But StrToDate fails when entering dates prior to 1.1.1978!

The program checks birthdays before accepting them and rejects incorrect dates.

Rating The rating is used to create an internal ranking, when pairing the first round (Swiss pairing only) and when computing the new DWZ of course. You might enter any

national rating, for example USCF-rating instead, if they are sorted descending. (Better players have higher numbers.) Of course you must not mix ratings of different systems, you should convert them instead.

Please notice that the official form of the DWZ is something like 1834-23, which means a rating of 1834 resulting from 23 tournaments. Chaos accepts both the official and the short form 1834.

ELO If players have an ELO-number, this rather than DWZ will be used by Chaos. (This seems wise to me, but could be handled in a different manner.) Please notice, that ELO-numbers should not be entered, if national ratings and ELO-numbers aren't equivalent. For example players which have the same DWZ or ELO should be expected to have the same playing strength.

For all these string-gadgets please have a look at Section 5 [Input-windows], page 5.

Senior

Junior

Woman

Junior (A)

Junior (B)

Junior (C)

Junior (D)

Junior (E) If you want to make separate tables for special groups (for example you can give a prize to the best woman), but all players are paired in one group, you can use these gadgets to mark the groups. This doesn't affect the pairings, it just adds the possibility of creating separate tables for these groups. See Subsection 6.4 [Table], page 11.

5.4 Setting games

Under certain circumstances you might wish to force special pairings. That's why this window appears before drawing the games of a new round in a Swiss pairing tournament. I don't expect you want this in most cases and so the window can be disabled using the menu item **Setting games**. See Subsection 6.5 [Setting games], page 12.

To the left of the window you find the list of all players except for withdrawn players. By double-clicking a player's name you put him into the list of set games on the right side of the window. Of course you need to double-click another player to make a game. Chaos doesn't allow you to pair players again. You will be asked, if you really want to pair players, if both had one color in the last two rounds. Using the **Delete** button you can remove games.

Normally Chaos should determine who will receive the white pieces. This makes the pairings easier in the following rounds. However, this can be changed using the cycle gadget below the list of

players. In that case the first player will receive the white pieces. (The order of setting is meaningless otherwise.) Matches with colors determined are indicated by a `(cd)` (colors determined) behind the names.

You can as well select a player for a one point bye: After all games have been selected, double-click the player for the one point bye. (You select an odd number of players.) Of course this isn't allowed, if the number of active players is even or if the player already had an one point bye.

5.5 The Input-of-results-window

By selecting a subitem of `Round-Input results` you create this window, which shows a list of one round's games. (see Subsection 6.3 [Input results], page 10) By using the usual Arrow- and scrollbar-gadgets you might scroll this list. There is always one game, which is activated and therefore highlighted (Kickstart 3.x) or shown beneath the List (Kickstart 2.x). You might activate another game by clicking it.

Some other gadgets are placed below. By clicking the right ones (`Played` and `Not played`) you tell the program, if a game really took place or was, for example, forfeited. (Default is, that the game was played.) The gadgets to the left (`White wins`, `Draw`, `Black wins`, `Result missing`) allow to enter the game's result. (Default is, that the result is missing.)

You may as well use the keyboard to enter results: The keys `W` (White wins), `D` (Draw), `B` (Black wins), `M` (Result missing), `P` (Game played) and `N` (Game not played) correspond to the gadgets above. The next game will automatically be activated after a result was entered. This allows a rather fast and comfortable input.

6 The program's menus

The program uses one main-window, which contains some informations: Tournament-name and -file, number of players and rounds and tournament-mode. It carries a menu-bar, while there is no input- or output-window.

6.1 The Project-menu

Most programs using a GUI have a menu like this. It's items are:

New This menu-item creates the Input-of-tournament-data-window. See Subsection 5.1 [Input of tournament data], page 5. If anything has changed since the last save/load, a requester appears and asks, if you want to save first.

- Load** You might load a tournament from disk using this using the `asl.library`. The same requester as above appears, if data has changed since the last save/load.
- Save** Here you save the tournament. Of course a tournament-file has to be set first. (This is done by loading a tournament.) `Project-Save as` is called without a tournament-file.
- Save as** Using this you might select another tournament-file with the `asl-requester` and save the tournament to it. I **strongly** encourage you to save to different files after each drawing. (see Subsection 3.2 [Precautions], page 3) This is supported by calling this menuitem automatically, if drawing is finished and using a name like `'chaos.5.cdat'` as default, where 5 is the number of rounds.
- About** Here some informations about the program and me (the author) is reported. I always select **About**, when exploring new programs hoping to see *interesting* information. Of course I am disappointed always. :-(Now I want to disappoint you! ;-)
- Quit** Oh no! You really need this? Think about it! Sure? Ok. No point in helping those people. Of course the program brings up a similar warning if you want to leave without saving modifications first.

6.2 The Players-menu

This menu's items are using the windows `Input of player-data` and `Player-selection`. They are described above. See Section 5 [Input-windows], page 5. Here are the menu-items:

- Add** Using this you enter new players before starting the tournament. When using the Swiss pairing, you may enter new players until the second round is paired. This of course is not in strict accordance with the rules, but some administrators will be happy about it. New players are given a one point bye, if the number of players was even before or have to play against the player who was free before (and he gets his bye removed). The new players are sorted in at the lower end of the internal list of rankings. Please note that you decide about the pairings of the new players, if you enter more than one! You should therefore follow the principles described in Subsection 9.3 [Late players], page 21.
- Import** You might wish to include players from a previous tournament in a new tournament without enter all data manually. This is possible by using this item: The `asl requester` appears, allowing to select an old tournament file and to get players from the player selection window.

You can create a player database by creating a new tournament, give it the name of your club, for example, and including the players here. See Subsection 8.2.2 [UpdateRatings], page 18.

Modify You are always able to modify players. Of course this should not happen after the first round and therefore a requester appears, asking you, whether you really want to do this.

Delete Players may be deleted before the first pairings, but not later. You can use this menu-item to mark players who dropped out instead, when using the Swiss pairing. Players which get the marked player as opponent, will receive a one point bye. Of course there will be at most one player receiving a free point per round. Dropped players are paired against each other to ensure this.

In either case there will be requesters which ensure that you don't delete players accidentally. Note, that **no** player is deleted, if you cancel **anywhere**.

6.3 The Round-Menu

This menu is designed to manage the menu after entering the players. It has just two menu-items:

Make pairings

This menuitem contains 3 subitems. By selecting one of these subitems the tournament is started and the first pairings are made using the corresponding mode: Either **Swiss pairing** (see Subsection 9.2 [Swiss pairing], page 19) or **Round Robin** according to the FIDE (see Subsection 9.1.1 [FIDE-system], page 18) or the **shift-system** (see Subsection 9.1.2 [shift-system], page 19). Please note, that it is not possible to change the pairing-mode later!

When playing a Round-Robin tournament, all games are paired at once and this menu-item is no longer needed. In a Swiss-pairing-tournament this needs to be selected again to create the pairings of another round. The program will not allow you to make new pairings until all results of the previous round are entered. If you need to proceed without waiting for all the results, you can use a simple trick: Enter all missing results as draw, make the pairings and mark the results again as missing. (Do not forget to re-modify the results, before making them public!)

If a pairing was successful, the menu-item **Project-Save as** is called automatically. I **strongly** recommend to use this! See Subsection 3.2 [Precautions], page 3.

Input results

The results are entered using this menu item. A requester appears asking for the round number, if you select it.

6.4 The Output-menu

All output except the player-cards and the cross-table can either be directed to screen, printer (Draft or Letter quality mode) or to a file. This is set in the preferences-menu. See Subsection 6.5 [Output-Device], page 12.

Output to the screen will appear in the Output-window. The text is arranged as a scrollable list and you may use Arrow- or Scrollbar-Gadgets to move the text up and down. For the printer please remember to set the printer-preferences. See Subsection 3.1 [Preparations], page 3. The `asl-filerequester` appears, if you want output to a file and select an output item and allows you to enter a filename.

Using the corresponding menuitems you can create the following lists:

List of players

You can select a **short** list containing only player-numbers, names, chess-clubs and ratings (DWZ and ELO) of the players or a **long** list containing all player-data.

List of rankings

This is the internal ranking used by the program. It is sorted by the official ratings before the first round, but changes later. There's no need to look at this list, except if you want to follow the program making pairings. See Subsection 9.2 [Swiss pairing], page 19. Please note that this list might look somewhat confusing, when results have been entered since the last pairing. The list isn't sorted until new pairings are created and therefore should be printed only immediately after creating them.

Table

The table can be sorted in different ways. This is set in the preferences-menu. See Subsection 6.5 [Table-mode], page 12. Additionally you can create separate tables of special groups (seniors, juniors, women, ...). See Subsection 5.3 [Input of player-data], page 6.

Round

This is the list of all games of one round. The number of the round is selected using the same requester as for entering results. See Subsection 6.3 [Enter results], page 10.

Table of progress

This table contains a list of all players, all games and results. It is designed to be used in the Swiss-pairing-mode and therefore disabled in the Round-Robin mode.

Cross-table

The cross-table is nearly the same as the table of progress, but designed for the Round-Robin mode and therefore disabled in the Swiss-pairing-mode. Another output mode is available here: Chaos can create $\text{T}_{\text{E}}\text{X}$ -source. If you select the subitem `TeX` the `asl-requester` appears, asking for a filename. The subitem `Ascii` works in the usual way.

Player-cards

The player-cards contain all information to manage the tournament manually. They are designed primarily for my purposes (debugging), but I decided to keep them in the public version for safety purposes. See Subsection 3.2 [Precautions], page 3. The output of the player-cards works in the same way as the output of the cross-table.

DWZ-ratings

The output of the DWZ-ratings is described in detail in another chapter. See Section 7 [DWZ-ratings], page 13.

6.5 The Preferences-Menu

Please notice, that the menu-items of this menu are valid for different periods.

Table-mode

You select the table-mode here, which determines how the tables are sorted. Please notice that the table mode is saved with the other tournament-data. Possible modes are (default is simple):

1. In the **simple** mode the table is sorted by points only.
2. Using the **Buchholz**-mode the table is sorted by points and Buchholz-points. A player's Buchholz-points are calculated as the sum of his opponent's points.
3. The **extended Buchholz**-mode is nearly the same as the Buchholz mode, except that extended Buchholz-points are included, which are the summarized Buchholz-points of the opponents.
4. The **Sonneborn-Berger**-mode is similar to the Buchholz-mode, except that the opponents points are multiplied by a constant depending on the result: 0 for a loss, one for a draw and 2 for a win.

Output-device

You set the output-device here. (Surprise! ;-) Possible devices are screen, printer using draft, printer using letter quality or output to a file. Default is screen, which is always set when the program starts.

Setting games

If this option is set (it is *not* when the program starts) and you select the menu item **Pairings** in a Swiss pairing tournament a window appears that allows you to set games. See Subsection 5.4 [Setting games], page 7.

Make icons

This shows, whether chaos will create an icon, if a tournament is saved. Default is to create icons, when started from Workbench and to omit this, when started from CLI. (This can be modified by using the **MAKEICONS**- or **DEFICON**-options or the

NOICONS-tooltype, respectively; see Section 4 [Startup], page 4) Selecting the menu-item toggles this mode.

6.6 The Special-Menu

This menu appears only if you have created a file containing own menuitems. See Subsection 8.2 [Own Menuitems], page 17.

7 The DWZ-report

The list of DWZ-ratings contains all players, who have played at least one rated opponent. The algorithm of computing the DWZ is similar to the ELO-algorithm and therefore can be used to get a rough impression of a players performance, even if another rating-system, for example USCF is used. For details of the algorithm see: Peter Zoefel, Karl-Heinz Glenz, *Das ELO-System*.

Age is a factor in determining the DWZ and the birthday of the player is required for an official rating-report. Therefore a requester appears, if a players birthday is missing when computing the DWZ. It offers the following answers:

1. One of the gadgets **20 or less**, **21-25**, **26 or more** makes the program suggest, that the player belongs to the corresponding age-group. The DWZ is computed in the normal way, but of course the birthday is missing on the DWZ-report.
2. The gadget **Modify** allows to modify the player's data, include the birthday and continue computing the DWZ.
3. **Drop** tells the program that this player should not be included in the DWZ-report.
4. **Cancel** breaks creating the DWZ-report.

The DWZ-report contains the following values:

| | |
|-----------|--|
| Ro | is the old DWZ. It should be entered in a form like 1834-23, where 1834 is the old DWZ and 23 is the number of rated tournaments. However, it is possible to enter the DWZ in the short form 1834. |
| W | are the points scored in games against opponents who already have a DWZ. |
| n | is the number of rated games. |
| We | is the win expectancy, that is the number of points the player is expected to score in order to keep the DWZ. |
| RH | is the so called H-number, the performance rating. |

Rn is the new DWZ in the same form as Ro. Previously unrated players get a provisional DWZ like 1457-(4), where 4 is equal to n. This provisional DWZ will be used in future tournaments. Once a player accumulates 9 games, his rating is established.

8 The ARexx-Port: Chaos as a batch program

Using the port name `CHAOS.1` you can control Chaos from other programs via ARexx. Even more: You don't need the GUI of Chaos, that's why you have the `NOWINDOW` option.

8.1 Possible ARexx commands

The ARexx commands correspond to menu items, you should therefore understand them immediately. Of course you cannot use an ARexx command, when the corresponding menu item is disabled. For example it is not possible to sort the table due to Buchholz points in a round robin tournament or make new pairings while the user enters results.

This are the commands:

NewTournament *NAME/A, FORCE/S*

Creates a new tournament with the given name. You are asked for confirmation, if the current tournament isn't saved. This can be suppressed using the `FORCE` option. See Subsection 6.1 [Project-New], page 8.

LoadTournament *FILE/A, FORCE/S*

Loads the given tournament. `FORCE` works like above. See Subsection 6.1 [Project-Load], page 8.

SaveTournament *FILE/A, ICON/S*

Saves the current tournament in the given file. You can create an icon using the `ICON` option. See Subsection 6.1 [Project-Save], page 8.

AddPlayer *NAME/A, STREET/K, VILLAGE/K, CHESSCLUB/K, BIRTHDAY/K, PHONE/K, RATING/K, ELO/K, FLAGS/K, NOUSER/S*

Enters a new player. A name must be present. Flags are a string containing one or more of the characters s (Senior), j (Junior), w (Woman) and a-e (Junior A, ..., Junior E). `RATING` is the DWZ or another national rating. See Subsection 5.3 [Player-input], page 6. See Subsection 6.2 [Players-Add], page 9. The option `NOUSER` suppresses error requesters.

ModifyPlayer *PLAYER/A, NAME/K, STREET/K, VILLAGE/K,*

CHESSCLUB/K, BIRTHDAY/K, PHONE/K, RATING/K, ELO/K, FLAGS/K

Modifies the data of the player whose name is *PLAYER*. Only the given fields are modified, for example

```
'ModifyPlayer "Widmann, Jochen" NAME "Wiedmann, Jochen"'
```

modifies the name only. See Subsection 6.2 [Players-Modify], page 9. NOUSER suppresses error requesters.

DeletePlayer *PLAYER/A, FORCE/S*

deletes the given player (or marks him as withdrawn in a Swiss pairing tournament). You are asked for confirmation without the *FORCE* option. See Subsection 6.2 [Players-Delete], page 9.

SetPlayer *PLAYER/A, INIT/S, FORCE/S*

Sets the given player for the next pairings. For example if you want to force the pairings of players 3-5 and 2-4 and give player 17 an one point bye you execute the *SetPlayer* command for players 3, 5, 2, 4 and 17 in that order and call *DoPairings* after that. You see: It works like the interactive setting of games. See Subsection 5.4 [Setting games], page 7.

You can remove the games that have been set so far using the *INIT* option. This must be called **before** executing *DoPairings* of course.) You might enter a new player together with the *INIT* option.

Chaos declines pairing the same players a second time or giving a one point bye to a player who already had one. You are asked for confirmation, if players should be paired, who had one and the same color in the last two rounds. (This can be suppressed with the *FORCE* option.) Normally Chaos determines, who will receive the white pieces. However, this can be changed using the *SETCOLOR* option. It **must** be given when setting the second player who will receive the black pieces. It is ignored when setting the first player. See Subsection 6.3 [Round-Pairings], page 10.

DoPairings *SWISSPAIRING/S, ROUNDROBIN/S, ROUNDROBINSHIFT/S,*
NOUSER/S

This command creates new pairings. You must give one of the options *SWISSPAIRING*, *ROUNDROBIN* (FIDE system) or *ROUNDROBINSHIFT* (shift system) before creating the first round. The setting games window will be opened without the *NOUSER* option and allows the user to enter new games.

Chaos declines to draw new games, if you gave a player an one point bye, but the number of active players is even, See Subsection 6.3 [Pairings], page 10.

EnterResult *WHITE/A, BLACK/A, RESULT/A/N, NOTPLAYED/S*

The result (2 = white wins, 1 = draw, 0 = black wins, -1 = result missing) of the match between players *WHITE* and *BLACK* is entered using this command. The *NOTPLAYED* option should be used, if the match didn't took place. See Subsection 6.3 [Round-Enter results], page 10.

The following output commands need a filename. The given file will receive the output in Ascii format. You can use the names 'prt:' for output to the printer in draft mode or 'prt:LQ' for letter quality.

PlayerList *FILE/A, SHORT/S*

Prints the list of players. The long format is used without the *SHORT* option. See Subsection 6.4 [List of players], page 11.

InternalRatings *FILE/A*

Prints the internal ratings. See Subsection 6.4 [Internal ratings], page 11.

Table *FILE/A, TABMODE/K/N, PLRMODE/K/N*

Prints the current table. The table is sorted due to the *TABMODE* (0 = simple, 1 = Buchholz, 2 = extended Buchholz, 3 = Sonneborn-Berger) and *PLRMODE* determines, which players should be included into the table (0 = all, 1 = seniors, 2 = juniors, 3 = women, 4, ..., 8 = Juniors A, ..., Juniors E). Defaults are simple and all. See Subsection 6.4 [Table], page 11.

Round *FILE/A, NUMBER/N*

Prints the given round; default is the last round. See Subsection 6.4 [Round], page 11.

TableOfProgress *FILE/A, TABMODE/K/N*

Prints the table of progress. See the table command for *TABMODE*. See Subsection 6.4 [Table of progress], page 11.

CrossTable *FILE/A, TEX/S*

Prints the crosstable. \TeX source is created with the *TEX* option. It is possible, to use this option together with the filename 'prt:', but this doesn't call \TeX and prints the source on the printer instead, that's why this probably doesn't make much sense. See Subsection 6.4 [Cross table], page 11.

PlayerCards *FILE/A, TEX/S*

Prints the player cards. See the *CrossTable* command for the *TEX* option. See Subsection 6.4 [Player cards], page 11.

DWZReport *FILE/A*

Prints the DWZ report. It isn't possible, to suppress the requester asking for a players birthday, if this is missing. See Section 7 [DWZ-ratings], page 13.

Window *ON/S, OFF/S*

Switches the main window on or off. See Section 4 [Startup], page 4.

8.2 Inserting ARexx commands into an own menu

Once you have created an own ARexx script you probably wish to execute this from Chaos itself. This is possible by creating an own menu calling this script.

Chaos is looking for a file called '`s/Chaos_Menu`' at startup. (This name can be changed by using the option `MENUEFILE`. See Section 4 [Startup], page 4.) If this isn't found, Chaos is looking for '`s:Chaos_Menu`' as well.

A menu called '`Special`' will be created, if one of this files is found. The file defines some menuitems, which will be inserted into the menu. Each menuitem is represented by two subsequent lines in the file: One for the label and one for a shell command which will be executed, if the corresponding menuitem is selected. (See '`s/Chaos_Menu`' for an example.)

Notes:

1. Chaos is not ready to receive commands until the Shell command terminates. Especially Chaos cannot execute ARexx commands while your Shell command is running! Hence you probably need to use the command `Run`.
2. Intuition cannot maintain more than 31 menuitems.

Two scripts are available yet. These should supply you with examples, but possibly might be useful for themselves. Both contain a description of how to use them

8.2.1 PrintTable: Formatting own tables

A swiss user group wanted a special formatted table: The points should be behind the players and the chessclub should appear as well. This script realizes it. It makes Chaos to write the current table and the list of players into special files. This are read by the script and the modified table gets printed.

There should be no problem to understand the script, except for one point: All this data doesn't fit on one line in any case! (Player and chessclub names may be up to 30 characters long.) The script solves this by cutting these names eventually.

8.2.2 UpdateRatings: Exporting new computed ratings into a database

The menuitem `Import` makes it possible to have a player database, as already mentioned. (see Subsection 6.2 [Players], page 9) This script makes it possible, to update the ratings of the database. You have to supply the script with a database name, either by calling it with `'UpdateRatings.rexx databasename'` (The name must be inserted in `'s/Chaos_Menu'` in that case.) or by selecting the database with the asl requester, if the script is called without a database name. (This assumes the existence of `c:RequestFile`.)

The script stores the current tournament in `'t:CurrentTournament'`, makes Chaos to write the new ratings into a file and load the database. The ARexx command `ModifyPlr` is used to update the ratings in the database which is saved again finally and the current tournament is restored.

9 Algorithms

The algorithms are described in detail in parts 2.1 (Round-Robin mode) and 2.2 (Swiss pairing) of the *Turnierleiterhandbuch*.

9.1 Round-Robin tournament

In an Round-Robin tournament each player meets all the other participants in the tournament once. In order to determine the pairings each player is assigned a random number, his pairing number. Assume that the pairing numbers are $1, 2, \dots, n$.

If n is odd, we add an imaginary player with number $n+1$. His opponent will be free for the corresponding round. Without loss of generality we may therefore assume that n is even and $k = n/2$;

There are two different ways to determine the pairings: The FIDE-system and the shift-system.

9.1.1 The FIDE-system

This is the system suggested by the FIDE. It is well known, but not very clear and needs more organization therefore. A player will hardly know his next opponent or the place where he will play.

The games of the first round are: $1 - n$, $2 - (n-1)$, $3 - (n-2)$ and so on.

In the following rounds we start by getting the opponent of player n . The opponents are: $k+1$ (round 2), 2 (round 3), $k+2$, 4 , $k+3$ and so on. The players $2, 3, \dots, k$ have the white-pieces chessmen, $k+1$, $k+2, \dots$ the black ones.

All other players are determined as follows: The next round's opponent is the player whose pairing number follows after the pairing number of the last round's opponent. Player n is not included in this and pairing number 1 follows after pairing number $n-1$. Pairings where a player should play himself mean that he plays player n instead.

9.1.2 The shift-system

This system has a big advantage: Every player will know his next opponent, the place where they will meet and the colors. The expense of management will be rather small except for the first round where you need to tell the players where they should sit down.

The games of the first round are as follows: 1 - $k+1$ (board 1), $k+2$ - 2 (board 2), 3 - $k+3$ (board 3), $k+4$ - 4 (board 4) and so on. The lower half players sit side by side, player 2 to the right of player 1.

After each round the players are shifted clockwise. The boards remain unchanged. An exception is the last board when having an even number of participants: Player n keeps his place and turns his board. (Player n is the one using the white pieces at the last board in the first round when having an even number of boards and the black pieces otherwise.) All boards remain unchanged when the number of players is odd: The opponent of the virtual player n is free in that case.

The shift-system isn't very popular outside Germany, as far as i know. An example may be useful to illustrate what i mean. Let the first round look like this:

| | | |
|---|---|---|
| 4 | 5 | 6 |
| 1 | 2 | 3 |

(Player 3 might be written alone in the diagram above, if there were only 5 players and he would be free.) When the first round is finished, players 1-5 are shifted clockwise and circular while player 6 keeps his place and turns his board. This leads to the following position:

| | | |
|---|---|---|
| 1 | 4 | 6 |
| 2 | 3 | 5 |

The shift will now be repeated after each round.

9.2 Swiss pairing

The Swiss pairing is needed, if there are more players than rounds. The principle of the algorithm is to pair players which are approximately of equal strength. In the end the best player should score the most points of course.

Each round begins by building an internal list of rankings. When starting the tournament, the players are sorted by the DWZ (or ELO-number, if there is one). Players without DWZ and ELO-number are placed at the end of the list. Players with equal rating (and players without a rating) are sorted alphabetically.

Let n be again the number of players. If n is odd, one player is selected randomly receiving a full point bye. In the first round this is a player without DWZ. (If there are fewer than 5 players without DWZ, the weakest ones are added to the group to ensure that this is a real drawing.) Without loss of generality we may therefore assume that n is even and k is $n/2$.

We start by determining randomly whether player 1 receives white or black. If he gets white, the games of round 1 are: 1 - $k+1$, $k+2$ - 2, 3 - $k+3$ and so on. If player 1 gets black the pairings are the same, but the colors are reversed.

Beginning with round 2 the players are sorted by points. The group of players with the same number of points are sorted by the last internal rankings. (So the influence of the DWZ on the rankings gets smaller with every round.)

Beginning with the highest group of players (those which have the most points), we try to pair the players of one group against each other. We divide each group in 4 subgroups: First we divide the group in an upper and lower half (by using the rankings of the previous round), then we separate those which had white in the previous round and those which had black. Let's look at an example:

| | |
|-------------------|-------------------|
| Upper half, white | Upper half, black |
| 1 | 2 |
| 3 | 5 |
| 4 | 6 |
| | 7 |
| Lower half, white | Lower half, black |
| 8 | 11 |
| 9 | 12 |
| 10 | 13 |
| | 14 |

We try to pair players of the upper half and players of the lower half, as we did in the first round. Additionally we would like to pair players of the left and the right side, because this allows more possible pairings in the next round. But this is not always possible: In the above example we need to pair at least two players from the right half.

We begin by getting an opponent for player 1. As described above, we try player 11. Player 2 is paired with player 8, player 3 with 12, player 4 with 14, 5 with 9, 6 with 10 and finally 7 with 14.

This assumes, that all games can be paired. Of course this is not always true. A pairing is allowed, if

1. The same opponents have not been paired in the previous rounds.
2. The players did not have the same colors in the previous two rounds. This rules out having three times black (or white) in a row.
3. It is possible to pair the rest of the players following 1. und 2. rules

It is allowed, that a player has the black pieces 4 times in 5 rounds! (Not my selection, i personally would prefer to prohibit this. But i follow the rules of the *Turnierleiterhandbuch*.)

If the number of players in the group is odd, there is one player left at the end. This player is moved to the next group. If the number of all players is odd, one player is left at the end. This player gets a full point bye. (Of course no player may get more than one bye.)

If it isn't possible to pair the players of a group, the lowest player is moved to the next group. This process needs to be repeated possibly.

9.3 Late players

When using the Swiss pairing, it is possible to add players after pairing the first round. If there is only one player to be added, simply enter him. But be aware that the order in which you input the new players determines the pairings, if there is more than one new player. In this case you are encouraged to follow these steps:

1. If the number of players was odd until now, add the free player to the new players.
2. Build the internal list of rankings of the new players and make the pairings as described above. See Subsection 9.2 [Swiss pairing], page 19. (If you think, this is too silly: Draw lots! :-)
3. If there was a free player until now, modify his data to the data of the first new player. (Not needed, of course, if he himself is the first. :-)
4. Enter the new players. The last player should be the one, which gets a bye now. (If there is one.)

10 (Pre)-history

V1.0 Initial version (1981) of the program, written on an Exidy Sorcerer (Operating System CP/M, Z80-processor) in — Basic. I admit. ;-) All data on disk, so the program was rather slow. Rather simple algorithms, which has not changed up to version 4.

V2.0 My first Computer (a C64) needed a new version, written in Assembler.

- V3.0 My first Amiga, an A1000 (Sigh!) got the next version, written using AmigaBasic. First version allowing Round-Robin mode.
- V4.0 Using my first C-Compiler (Aztec-C V3.4a) this version was created.
- V4.2 Some suggestions by Frank-Thomas Elpelt: Adding of late players, marking players as withdrawn. But the player's needs are raising: Some of them flamed the way of handling the colors. (O.k., they have a point.)
- V5.0 Running on Kickstart 2.x only GUI created using GadToolsBox (suggested by Kai Bolay). Algorithms now as described in the *Turnierleiterhandbuch*. DWZ-ratings introduced. This was a beta-version and never released.
- V5.1 International version using locale.library. Some minor suggestions of the beta-testers introduced.
- V5.1a Minor bugfixes; localizing for Workbench 2.0
- V5.2 User interface now using MUI; ARexx port, setting of games introduced; Cross table and player cards available in ascii format
- V5.3 Bugfixes; using some advantages of MUI 2.0; introduced winner- and drawpoints

11 Future of the program

Many things could be done: International ranking-systems like USCF or ELO, Double-Round-Robin-tournaments. The pairing could be done in assembler to increase the speed.

But *I* don't need these changes. I will not introduce them (nore will I develop the program any more) without feed-back from the users. (Especially the international rankings couldn't be done without help because I don't know the algorithms.) So it is in *your* hands.

Credits

My thanks go to

Stefan Stuntz

for creating MUI. I cannot help people still don't liking the user interface.

Reinhard Spisser und Sebastiano Vigna

for the Amiga-version of texinfo. This documentation is written with it.

The Free Software Foundation

for the initial version of texinfo and many other excellent programs.

My Betatesters

Kai Bolay (kaib), Frank Geider, Franz Hemmer, Juergen Lang (Quarvon), Christian Soltenborn and Volker Zink (The Zinker).

My friend Mathias Moersch

for his help in the manual's english translation. Spending nearly 6 years in the U.S. finally does some good!

The people of #AmigaGer

for answering many stupid questions and lots of fun, for example PowerStat (Kai Hoffmann), ZZA (Bernhard Moellemann), Stargazer (Petra Zeidler), stefanb (Stefan Becker), Tron (Mathias Scheler), ill (Markus Illenseer) and jow (Juergen Weinelt).

Commodore

for the Amiga and Kickstart 2.0. Keep on developing it and I'll be an Amiga-user for the next 8 years too. ;-)

Douglas Adams and Tom Sharpe

for creating Arthur Dent and Henry Wilt. My favourite (anti)-heroes!

and to my parents for their bath-tub:

There's no better place to think ;-)

References

Das ELO-System

(The ELO-System), from Peter Zoefel, Friedrich Froebel-Str. 5, 35041 Marburg, Germany and Karl-Heinz-Glenz, Friedenau 22, 46284 Dorsten 21, Germany; published by the authors themselves

Turnierleiterhandbuch des Deutschen Schachbundes

(The official german chess federation's guide to managing chess-tournaments), from Ernst Schubart and Helmut Noettger, de Gruyter, Berlin, New York, 1987

All algorithms and rules follow those books.

Index

A

| | |
|--------------------------------|----|
| About (item) | 9 |
| Add (item) | 9 |
| AddPlayer, ARexx command | 14 |
| Algorithms | 18 |
| ARexx | 14 |
| ARexx commands | 14 |
| Arguments | 4 |
| Ascii (subitem) | 11 |

B

| | |
|------------------------------------|----|
| Backup | 3 |
| batch program | 14 |
| Birthday (gadget) | 6 |
| Buchholz (subitem) | 12 |
| Buchholz, extended (subitem) | 12 |

C

| | |
|---------------------------------|----|
| Cancel | 5 |
| Chess-club (gadget) | 6 |
| CLI | 4 |
| Compiler | 1 |
| Credits | 22 |
| Cross-table (item) | 11 |
| CrossTable, ARexx command | 16 |

D

| | |
|-----------------------------------|----|
| Delete (item) | 10 |
| DeletePlayer, ARexx command | 15 |
| DoPairings, ARexx command | 15 |
| Draft | 12 |
| DWZ-ratings | 13 |
| DWZ-ratings (item) | 12 |
| DWZReport, ARexx command | 17 |

E

| | |
|----------------------------------|----|
| ELO (gadget) | 7 |
| EnterResult, ARexx command | 16 |

F

| | |
|----------------------|----|
| FIDE-system | 18 |
| File (subitem) | 12 |
| Free point | 20 |
| Future | 22 |

H

| | |
|------------------------------|----|
| History | 21 |
| Home address (gadgets) | 6 |

I

| | |
|--------------------------------------|----|
| Import (item) | 9 |
| Input results | 10 |
| Input-windows | 5 |
| Installation | 1 |
| InternalRatings, ARexx command | 16 |

J

| | |
|----------------------------|---|
| Juniors (A) (gadget) | 7 |
| Juniors (B) (gadget) | 7 |
| Juniors (C) (gadget) | 7 |
| Juniors (D) (gadget) | 7 |
| Juniors (E) (gadget) | 7 |
| Juniors (gadget) | 7 |

L

| | |
|-------------------------------------|----|
| Late players | 21 |
| Letter Quality | 12 |
| Limitations | 1 |
| List of players (item) | 11 |
| List of rankings (item) | 11 |
| Load (item) | 9 |
| LoadTournament, ARexx command | 14 |

M

| | |
|----------------------------|----|
| Make icons (item) | 12 |
| Make pairings (item) | 10 |
| Menuitems, building | 17 |
| Menus | 8 |

| | |
|----------------------------------|----|
| Modify (item) | 10 |
| ModifyPlayer, ARexx command..... | 15 |

N

| | |
|-----------------------------------|----|
| Name (gadget)..... | 6 |
| New (item)..... | 8 |
| NewTournament, ARexx command..... | 14 |
| Number of players | 1 |
| Number of rounds | 1 |

O

| | |
|----------------------------|----|
| Options..... | 4 |
| Output (menu) | 11 |
| Output device (item) | 12 |
| Own Menuitems | 17 |

P

| | |
|---------------------------------|----|
| Pairings | 10 |
| Phone (gadget) | 6 |
| player database | 9 |
| player database, updating..... | 18 |
| Player-cards (item) | 12 |
| Player-input | 6 |
| Player-selection | 6 |
| PlayerCards, ARexx command..... | 16 |
| PlayerList, ARexx command | 16 |
| Players (menu) | 9 |
| Precautions..... | 3 |
| Preferences (menu) | 12 |
| Preparations | 3 |
| Printer (Draft) (subitem) | 12 |
| Printer (LQ) (subitem)..... | 12 |
| PrintTable.rexx | 17 |
| Program language | 1 |
| Project (menu) | 8 |

Q

| | |
|------------------|---|
| Quit (item)..... | 9 |
|------------------|---|

R

| | |
|----------------------|----|
| Rating (gadget)..... | 7 |
| References | 23 |
| Requirements | 1 |

| | |
|----------------------------|----|
| Result-input | 8 |
| Room..... | 3 |
| Round (item)..... | 11 |
| Round (menu)..... | 10 |
| Round, ARexx command | 16 |
| Round-Robin | 18 |

S

| | |
|-------------------------------------|----|
| Save (item)..... | 9 |
| Save as (item) | 9 |
| SaveTournament, ARexx command | 14 |
| Screen (subitem)..... | 12 |
| Seniors (gadget) | 7 |
| SetPlayer, ARexx command..... | 15 |
| Setting games (item)..... | 12 |
| Setting games (window)..... | 7 |
| shift-system | 19 |
| Simple (subitem) | 12 |
| Sonneborn-Berger (subitem) | 12 |
| Special (Menu) | 13 |
| Stack..... | 2 |
| Startup..... | 4 |
| Survey | 1 |
| Swiss pairing | 19 |

T

| | |
|-------------------------------------|----|
| Table (item)..... | 11 |
| Table of progress (item)..... | 11 |
| Table, ARexx command | 16 |
| Table-mode (item)..... | 12 |
| TableOfProgress, ARexx command..... | 16 |
| TeX (subitem)..... | 11 |
| Todo | 22 |
| Tooltypes | 4 |
| Tournament-name (gadget) | 5 |

U

| | |
|-------------------------|----|
| UpdateRatings.rexx..... | 18 |
|-------------------------|----|

W

| | |
|----------------------------|----|
| Window, ARexx command..... | 17 |
| Women (gadget) | 7 |
| Workbench | 4 |

Table of Contents

| | | |
|----------|--|-----------|
| 1 | Survey | 1 |
| 2 | Installation | 1 |
| 3 | Management | 3 |
| | 3.1 Preparations | 3 |
| | 3.2 Precautions | 3 |
| 4 | Starting the program | 4 |
| 5 | The program's input-windows | 5 |
| | 5.1 The Input-of-tournament-data-window | 5 |
| | 5.2 The Player-selection-window | 6 |
| | 5.3 The Input-of-player-data-window | 6 |
| | 5.4 Setting games | 7 |
| | 5.5 The Input-of-results-window | 8 |
| 6 | The program's menus | 8 |
| | 6.1 The Project-menu | 8 |
| | 6.2 The Players-menu | 9 |
| | 6.3 The Round-Menu | 10 |
| | 6.4 The Output-menu | 11 |
| | 6.5 The Preferences-Menu | 12 |
| | 6.6 The Special-Menu | 13 |
| 7 | The DWZ-report | 13 |
| 8 | The ARexx-Port: Chaos as a batch program | 14 |
| | 8.1 Possible ARexx commands | 14 |
| | 8.2 Inserting ARexx commands into an own menu | 17 |
| | 8.2.1 PrintTable: Formatting own tables | 17 |
| | 8.2.2 UpdateRatings: Exporting new computed ratings into a database | 18 |
| 9 | Algorithms | 18 |
| | 9.1 Round-Robin tournament | 18 |

| | | |
|-----------|-----------------------------------|-----------|
| 9.1.1 | The FIDE-system..... | 18 |
| 9.1.2 | The shift-system..... | 19 |
| 9.2 | Swiss pairing..... | 19 |
| 9.3 | Late players..... | 21 |
| 10 | (Pre)-history..... | 21 |
| 11 | Future of the program..... | 22 |
| | Credits..... | 22 |
| | References..... | 23 |
| | Index..... | 24 |