Mandelmania

COLLABORATORS			
	TITLE : Mandelmania		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		April 10, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Mandelmania

1.1 Mandelmania Help

```
--------------------------------------------------------------------------
        *                                                     *
     *****      ------   M A N D E L M A N I A   -----    *****
   ** *******                                             ******* **
  **********    -----    VERSION 4.0  28-FEB-93    ----   **********
   ** *******                                             ******* **
     *****      ----    WRITTEN BY MARKUS ZEHNDER   ---    *****
        *                                                     *
--------------------------------------------------------------------------


              @{ " Overview " link Overview}
              @{ " Starting Mandelmania " link Start}
              @{ " Keystrokes " link Keystrokes}
              @{ " Fractal Types " link FractalTypes}
              @{ " Menus " link Menus}
              @{ " ARexx Interface " link ARexx}
              @{ " Animations " link Animations}
              @{ " Technical Information " link TechInfo}
              @{ " Credits " link Credits}
```

1.2 Mandelmania/General

```
              @{ " Author " link Author}
              @{ " Copyright " link Copyright}
              @{ " Disclaimer " link Disclaimer}
              @{ " History " link History}
              @{ " Overview " link Overview}
              @{ " Known Bugs " link Bugs}
```

1.3 Mandelmania/General/Author

Mandelmania was written by

Markus Zehnder
Inwilerriedstrasse 47
CH-6340 Baar
Switzerland

E-Mail: zehnder@amiga.physik.unizh.ch

Please note, that between the march 1 and june 1, I'm on a USA trip and
won't read my mail. So don't expect an answer if you send me mail in
this time.

Bug reports, criticism, suggestions, comments etc. are always welcome
(in English or German)

1.4 Mandelmania/General/Copyright

The program Mandelmania is copyrighted © 1990-1993 by the author. All rights
reserved worldwide.

Mandelmania is FREEWARE. This program may be freely distributed as long as
all executables and documentation remain unchanged and are included in the
distribution. You can use this program without paying a fee to the author.

You're not allowed to sell Mandelmania for for a more than nominal copying
fee. This fee MUST NOT be more than US $5.

Some people asked me to have an update service. If you want the next
version of Mandelmania, send me at least 5 or 10 (or what it is worth to
you) Dollars or Swiss francs. Don't forget to write me the version you
currently use.

1.5 Mandelmania/General/Disclaimer

The author cannot be held liable for the suitability or accuracy of this
software and/or this manual. Any damage directly or indirectly caused by the
use of this software is the sole responsibility of the user.

1.6 Mandelmania/General/History

Mandelmania 1.0:
 - first release

Mandelmania 1.1:
 - file requester added

Mandelmania 1.2:
 - color requester added

```
  – special Turbo version included

Mandelmania 2.0:
 – color cycling, proportional zooming and cursor key scrolling added.

Mandelmania 2.1:
 – bug fixed in Julia set for 68000 mode

Mandelmania 3.0:
 – complete rewritten version
 – faster calculation routines
 – FPU calculation routines added
 – several OS2.0 features supported
 – ARexx Port included
 – animation support

Mandelmania 4.0:
– Filerequesters go to Mandelmania screen
– Better colorcycling; now a interrupt server is used
– Better colormap handling (up to 256 colors should be possible)
– Bug fixed for 68000-mode. User could select FPU --> GURU.
– New About Window
– Error in Screenmode-requester fixed.
– Several bugs in palette requester found.
– Different Filerequester
– Palette can be saved and loaded
– Preferences can be saved and loaded
– Form file for ARexx output files. This feature allows more user inter-
  vention in the automatical ARexx file generation
– More choices in the iterations menuitem. Some people didn't find out that
  they can enter every iteration value using the Enter Range menuitem.
– New ZoomHandling (object before action)
– New Titlebar handling (several options possible)
– Buffermode for storing iterations of a Mandelbrot Set
– Fully font associative, font can be selected by a fontrequester.
– Saving pictures as brushes
– Lyapunovia algorithm implemented
– Paths for several directories (pictures, anims, palettes etc.) can now
  be edited and saved.
– internal 24-bit palette
– save requester flag
– bug fixed for 68000-mode. After loading Prefs saved on a 68020 machine,
  Mandelmania crashed running on a 68000 machine.
– CPM (Continous Potentional Method) implemented (two- and
  threedimensional)
– Full AGA support
– Only two tooltypes: PALETTE, SETTINGS
```

1.7 Mandelmania/General/Overview

```
Mandelmania is a fast Fractal calculation program.
The main features are:

 – 2.5 times faster than MandFXP
 – Needs Kickstart 2.1 (V38+ of asl.library)
```

- On-line help using amigaguide.library
- Supports all Amiga graphic modes, incl. AGA modes and autoscroll screens
- Loading and saving using IFF format. Picture parameters are stored in a
  special chunk
- Entering new parameters using either the keyboard or the mouse (different
  zoom modes)
- Supported Fractal types:
      Mandelbrot Set LSM
      Julia Set LSM
      Mandelbrot Set CPM (two- and threedimensional)
      Julia Set CPM (two- and threedimensional)
      Lyapunov Space
- The graphic output window can be sized as you like
- Colormap can be changed by a comfortable colorrequester
- Built in colorcycling
- ARexx interface
- Creating animations automatically via ARexx scriptfile
- Easy scrolling by pressing the cursor keys

1.8 Mandelmania/General/Known Bugs

Unfortunately, Mandelmania 4.0 has still some bugs. I wanted to release
this version before I'm on holidays for three months.

- Describtion only as Amigaguide document
- Lyapunov only implemented in low precision (use FPU mode for high
  precision)
- AGA colorcycling doesn't work
- Colorcycling must be improved (flickers when mouse moves)
- Save as PPM doesn't work
- Special CPM and Lyapunov parameters can't be interpolated in animations
- Only full screen are saved for animations (not brushes)
- Buffer isn't valid anymore after scrolling
- Help produces two enforcer hits. This hits are produced in the amigaguide-
  library call "OpenAmigaGuide" (can change in newer amigaguide.libraries).
- Sometimes crashes unexpectedly :-))

1.9 Mandelmania/Keystrokes

```
Cursor keys : @{ " Scrolling " link Scrolling }
Help key    : @{ " Help " link MENU_Help }
Del key     : Remove rubberband or Julia constant mark


Numpad
------


'0'        : Redraw (or recalculate if needed)
'1'-'9'    : @{ " Scrolling " link Scrolling }
'+'        : Zoom in
'-'        : Zoom out
```

1.10   Mandelmania/Keystrokes/Scrolling

```
Pictures can be scrolled easily by pressing the cursor keys. All bitmaps will
then be moved 2 pixels in the selected direction and only two rows and/or
columns have to be calculated.
Only twodimensional pictures can be scrolled.
```

1.11   Mandelmania/Starting Mandelmania

```
            You should start Mandelmania from Workbench because of the
             tooltypes
            defined in the Mandelmania.info file.
```

1.12   Mandelmania/Starting Mandelmania/Workbench/Tooltypes

```
The following tooltypes are available:

                @{ " SETTINGS " link Settings}
                @{ " PALETTE " link Palette}
```

1.13   Mandelmania/Starting Mandelmania/Workbench/Tooltypes/SETTINGS

```
This tooltype defines your standard Mandelmania settings file.
```

1.14   Mandelmania/Starting Mandelmania/Workbench/Tooltypes/PALETTE

```
This tooltype defines your standard Mandelmania palette file.
```

1.15   Mandelmania/Fractal Types

```
Mandelmania 4.0 can calculate several different fractal types.

    @{ " Mandelbrot Set " link TYPE_Mandelbrot}
    @{ " Julia Set " link TYPE_Julia}
    @{ " Mandelbrot Set CPM " link TYPE_MandelbrotCPM}
    @{ " Julia Set CPM " link TYPE_JuliaCPM}
    @{ " Lyapunov Space " link TYPE_Lyapunov}
```

1.16   Mandelmania/Fractal Types/Mandelbrot Set

```
Formula
-------
```

```
z[n+1] = z$^2$[n]+c
```

```
  z[] and c are complex numbers
```

```
  z[0] = 0
  c    = complex coordinate of pixel
```

The recursiv formula is calculated until either |z|^2>8 or
n>maxiter (maximal number of iterations)

```
Parameters
----------
```

You can enter the following parameters for Mandelbrot Sets.

The first four parameters define the Mandelbrot Set range being displayed.

```
x min = Real part of the pixelcoordinates on the left edge of the screen
x max = Real part of the pixelcoordinates on the right edge of the screen
y min = Imaginary part of the pixelcoordinates on the top edge of the
        screen
y max = Imaginary part of the pixelcoordinates on the bottom edge of the
        screen
```

iterations = Maximal number of iterations being calculated.

color divide = number of successive iterations having the same color.

1.17   Mandelmania/Fractal Types/Julia Set

```
Formula
-------
```

```
z[n+1] = z$^2$[n]+c
```

```
  z[] and c are complex numbers
```

```
  z[0] = complex coordinate of pixel
  c    = complex constant (the same for the whole picture)
```

The recursiv formula is calculated until either |z|^2>8 or
n>maxiter (maximal number of iterations)

```
Parameters
----------
```

You can enter the following parameters for Julia Sets.

The first four parameters define the Julia Set range being displayed.

```
x min = Real part of the pixelcoordinates on the left edge of the screen
x max = Real part of the pixelcoordinates on the right edge of the screen
y min = Imaginary part of the pixelcoordinates on the top edge of the
        screen
y max = Imaginary part of the pixelcoordinates on the bottom edge of the
        screen

x const = Real part of the Julia constant
y const = Imaginary part of the Julia constant

iterations = Maximal number of iterations being calculated.

color divide = number of successive iterations having the same color.
```

1.18 Mandelmania/Fractal Types/Mandelbrot Set CPM

```
Formula
-------

z[n+1] = z$^2$[n]+c

  z[] and c are complex numbers

  z[0] = 0
  c    = complex coordinate of pixel
```

The recursiv formula is calculated until either |z[n]|^2>64 or
n>maxiter (maximal number of iterations)

Then, a continous potential of the point must be calculated. If
n>maxiter, then this potential is equal zero.
In the other case, the potential is calculated as follows:

```
          / log(|z$^2$[n]| \
p(z) = 0.5* | ----------- |
          |      n      |
          \      2      /
```

To vary the steepness of the graphic, linear and logarithmic scaling
are possible.

```
                  logscale
P(z) = linscale * p(z)
```

```
Parameters
----------
```

You can enter the following parameters for CPM Mandelbrot Sets.

The first four parameters define the Mandelbrot Set range being displayed.

```
x min = Real part of the pixelcoordinates on the left edge of the screen
x max = Real part of the pixelcoordinates on the right edge of the screen
y min = Imaginary part of the pixelcoordinates on the top edge of the
        screen
y max = Imaginary part of the pixelcoordinates on the bottom edge of the
        screen

iterations    = Maximal number of iterations being calculated.

log. scale    = logarithmic scale constant
color periods = number of color periods (used only in 2D view)
                  (=how many times is the full palette between iteration 0
                   and the maximal iteration)

lin. scale    = linear scale constant (used only in 3D view)
plateau [%]   = level of the plateau (cut level)  (used only in 3D view)
angle         = In 3D view, the graphic can be rotated.
                  angle = 0  -> front view
                  angle = 90 -> top view
```

1.19   Mandelmania/Fractal Types/Julia Set CPM

```
Formula
-------


z[n+1] = z$^2$[n]+c

  z[] and c are complex numbers

  z[0] = complex coordinate of pixel
  c    = complex constant (the same for the whole picture)
```

The recursiv formula is calculated until either |z[n]|^2>64 or
n>maxiter (maximal number of iterations)

Then, a continous potential of the point must be calculated. If
n>maxiter, then this potential is equal zero.
In the other case, the potential is calculated as follows:

```
           / log(|z$^2$[n]| \
p(z) = 0.5* | ----------- |
           |      n      |
           \      2      /
```

To vary the steepness of the graphic, linear and logarithmic scaling
are possible.

```
                     logscale
P(z) = linscale * p(z)
```

```
Parameters
----------

You can enter the following parameters for CPM Julia Sets.

The first four parameters define the Julia Set range being displayed.

x min = Real part of the pixelcoordinates on the left edge of the screen
x max = Real part of the pixelcoordinates on the right edge of the screen
y min = Imaginary part of the pixelcoordinates on the top edge of the
        screen
y max = Imaginary part of the pixelcoordinates on the bottom edge of the
        screen

x const = Real part of the Julia constant
y const = Imaginary part of the Julia constant

iterations    = Maximal number of iterations being calculated.

log. scale    = logarithmic scale constant
color periods = number of color periods (used only in 2D view)
                (=how many times is the full palette between iteration 0
                  and the maximal iteration)

lin. scale    = linear scale constant (used only in 3D view)
plateau [%]   = level of the plateau (cut level)  (used only in 3D view)
angle         = In 3D view, the graphic can be rotated.
                  angle = 0  -> front view
                  angle = 90 -> top view
```

1.20 Mandelmania/Fractal Types/Lyapunov Space

```
Formula
-------

x[n+1] = r*x[n]*(1-x[n])

  x[] and r are real numbers

  r is either the x-coordinate or the y-coordinate of the current pixel
  and can change every iteration.
  The user can define this sequence by typing an 'A' for the x-coordinate
  and a 'B' for the y-coordinate. For example, the sequence "AB" would
  mean that r changes between x- and y-coordinate after every iteration.

This recursive formula needs a certain time to settle (if the function
is converging).

The Lyapunov exponent is calculated by:

L(x) = Average(log2(abs(r-2*r*x[n=1..maxiter])))

All exponents between the minimal Lypunov exponent and 0 are displayed.
```

```
Parameters
----------

You can enter the following parameters for Lyapunov Spaces.

The first four parameters define the graphics range being displayed.

x min = x-coordinates of the pixels on the left edge of the screen
x max = x-coordinates of the pixels on the right edge of the screen
y min = y-coordinates of the pixels on the top edge of the screen
y max = y-coordinates of the pixels on the bottom edge of the screen

lyaexpmin  = minimal Lyapunov exponent being displayed
sequence   = Lyapunov sequence (sequence of 'A's and 'B's)

settle     = Number of iterations for settling
iterations = Number of iterations for Lyapunov exponent calculation
             This value should be higher than the number of colors
             of the current screen.
```

1.21 Mandelmania/Menus

```
  Project Menu:
    @{ " Open " link MENU_Open}
    @{ " Save As " link MENU_Save_As}
    @{ " Titlebar " link MENU_Titlebar}
    @{ " About " link MENU_About}
    @{ " Help " link MENU_Help}
    @{ " Quit " link MENU_Quit}

  Calculation Menu:
    @{ " Buffer Mode " link MENU_Buffer_Mode}
    @{ " Reset Range " link MENU_Reset_Range}
    @{ " Enter Range " link MENU_Enter_Range}
    @{ " Zoom In " link MENU_Zoom_In}
    @{ " Zoom Out " link MENU_Zoom_Out}
    @{ " Zoom Mode " link MENU_Zoom_Mode}
    @{ " Type " link MENU_Type}
    @{ " CPM Type " link MENU_CPM_Type}
    @{ " Iterations " link MENU_Iterations}
    @{ " Machine " link MENU_Machine}
    @{ " Precision " link MENU_Precision}
    @{ " Recalculate " link MENU_Recalculate}

   Settings Menu:
    @{ " Default Screen " link MENU_Default_Screen}
    @{ " Screen Mode " link MENU_Screen_Mode}
    @{ " Windowsize " link MENU_Windowsize}
    @{ " Palette " link MENU_Palette}
    @{ " Open Font " link MENU_Open_Font}
    @{ " Paths " link MENU_Paths}
    @{ " Load Palette " link MENU_Load_Palette}
    @{ " Save Palette " link MENU_Save_Palette}
```

```
   @{ " Save Palette As " link MENU_Save_Palette_As}
   @{ " Load Settings " link MENU_Load_Settings}
   @{ " Save Settings " link MENU_Save_Settings}
   @{ " Save Settings As" link MENU_Save_Settings_As}

 Cycling Menu:
   @{ " Delay " link MENU_Delay}
   @{ " Forwards " link MENU_Forwards}
   @{ " Backwards " link MENU_Backwards}
   @{ " Off " link MENU_Off}

 ARexx Menu:
   @{ " Select Form " link MENU_Select_Form}
   @{ " New " link MENU_New}
   @{ " Append " link MENU_Append}
   @{ " Interpolate " link MENU_Interpolate}
   @{ " Close " link MENU_Close}
   @{ " Execute " link MENU_Execute}
```

1.22 Mandelmania/Menus/Open

Loads a fractal image. You can select the file using a standard asl-
filerequester. The file must include a MAND chunk to be loaded
successfully.

1.23 Mandelmania/Menus/Save As

Saves a fractal image. You can save the picture as a picture, brush or as a
ppm file. Additional to the standard ILBM chunks, some @{ " special chunks "
link SpecialChunk} is saved containing the constants representing the saved
picture. If the current fractal type is either CPM or Lyapunov, additional
to the MAND chunk, a 'CPM1' or 'LYA1' chunk is saved containing the extra
parameters.

```
Options
-------
```

Picture: The whole screen is saved, even if the fractal window size is
         smaller than the screen size.

Brush: The size of the saved picture is the same as the window size.
         You can change the window size using @{ " Windowsize " link ↩
            MENU_Windowsize}.

PPM: --> NOT YET IMPLEMENTED <---
         Save picture as a PPM P6 file. This is a 24 bit standard which
         can converted by a lot of image conversion programs (e.g. WASP,
         HamLabPlus)

1.24   Mandelmania/Menus/Titlebar

Specifies the function of the titlebar.

Off: No Titlebar is displayed.

No Coordinates: Titlebar is displayed, but no coordinates.

Window Coordinates: The window coordinates (x,y) of the mouse pointer
                        are displayed in the titlebar.

Complex Coordinates: The complex fractal coordinates of the mouse pointer
                        are displayed in the titlebar.

1.25   Mandelmania/Menus/About

An about window appears containing information about the current version
and free memory.

1.26   Mandelmania/Menus/Help

This command provides on-line help. You need "amigaguide.library" to use
this feature.

You can use help:

1. Select "Help" from the menu.
2. Press the "Help" key:
     Same function as "Help" menu. Additional you can get help when windows
     are open which provide help (for example palette-windows or enter-range-
     windows).
3. Select a Menu, and instead of releasing the right mousebutton, you
   press the "Help" key. You get then help on the menu command you selected.

1.27   Mandelmania/Menus/Quit

Leaves the program after asking for confirmation.

1.28   Mandelmania/Menus/Buffer Mode

Turns buffer mode on and off.
(MUST be on for three-dimensional CPM fractal types)

To speed up several operations and to make 3D CPM fractal types possible, a
data buffer for the values of each pixel is used. Several operations don't
need a recalculation of the fractal values, for example changing the number
of bitmaps, recalculation etc. After operations like this, the fractal
isn't recalculated, only a lookup in the table is needed.

1.29   Mandelmania/Menus/Reset Range

```
The range parameters are reset. The default values are:

Mandelbrot Set:  xleft   = -2.5
                 xright  =  1.5
                 ybottom = -1.75
                 ytop    =  1.75

Julia Sets:      xleft   = -2
                 xright  =  2
                 ybottom = -2
                 ytop    =  2

Lyapunov Space:  xleft   = -3
                 xright  =  3
                 ybottom = -3
                 ytop    =  3
```

1.30   Mandelmania/Menus/Enter Range

```
This command lets you enter new coordinates and picture parameters.  For
every fractal type, you have to enter different parameters.

@{ " Enter Mandelbrot " link TYPE_Mandelbrot}
@{ " Enter Julia " link TYPE_Julia}
@{ " Enter Mandelbrot CPM " link TYPE_Mandelbrot_CPM}
@{ " Enter Julia CPM " link TYPE_Julia_CPM}
@{ " Enter Lyapunov " link TYPE_Lyapunov}
```

1.31   Mandelmania/Menus/Zoom In

```
Zooms into an area.  But you first have to select the zoom area (before
choosing this command). This is done by selecting two points defining
the zoom area (see also @{ " Zoom Mode " link MENU_Zoom_Mode}).
```

1.32   Mandelmania/Menus/Zoom Out

```
Zooms out of an area.  But you first have to select the zoom area (before
choosing this command). This is done by selecting two points defining
the zoom area (see also @{ " Zoom Mode " link MENU_Zoom_Mode}).
```

1.33   Mandelmania/Menus/Zoom Mode

For several operations (zoom, windowsize) you have to select a
"rubberband".

You can select two different ways, how to draw a "rubberband":

Corner-Corner: The first selected point (left mouse button pressed)
               and the second selected point (left mouse button released)
               are opposite corners of the "rubberband".

Center-Corner: The first selected point (left mouse button pressed) is the
               center of the "rubberband" and the second selected point
               (left mouse button released) is a corner.

You can also turn proportional mode on and off. If this mode is turned on
the x- and y-range are proportional to the window width and height.
Therefore the width and height of a pixel are equal in (complex) fractal
coordinates and the proportion of the resulting fractal will look
reasonable (this is especially useful for Zoom In and Zoom Out).

1.34 Mandelmania/Menus/Type

You can select different Fractal types:

Mandelbrot Set: The default mandelbrot set is drawn immediately.

Julia Set: Before you start this command, you have to select a
                     "Julia Constant". This is done by clicking in the set.
                     A cross will appear where the Julia Constant is set.
                     The Julia Set will then be calculated immediately.

Mandelbrot Set CPM: A window pops up which asks you for several parameters.
                     The picture will be calculated after choosing OK.

Julia Set CPM: A window pops up which asks you for several parameters.
                     If you set a "Julia Constant" before executing this
                     command, this constant is set as a default.
                     The picture will be calculated after choosing OK.

Lyapunov Space: A window pops up which asks you for several parameters.
                     The picture will be calculated after choosing OK.

1.35 Mandelmania/Menus/CPM Type

You can select either two- or threedimensional CPM mode. If choosing
threedimensional mode, you must have the buffer mode turned on.

1.36 Mandelmania/Menus/Iterations

You can change the maximal number of iterations. You can choose from
different values. You should increase this value if you zoom into the
fractal image to see more details. If you select 'other' you can enter
the maximal number of iterations in a requester.

1.37   Mandelmania/Menus/Machine

Defines the machinetype. You can choose either 68000, 68020/30/40 or FPU.

The subitems which aren't available on your computer are ghosted. If there
is a 68040 in your system, the FPU mode is also selectable.

68000 and 68020/30/40 mode is fix point arithmetics. This is much faster
then floating point arithmetics (except 68040) but the precision is lower.

1.38   Mandelmania/Menus/Precision

Defines the precision. This makes only sense in 68000 or 68020/30/40 mode.

When low precision is selected, 16 bit fix point arithmetics are used
for calculation.

When high precision is selected, 32 bit fix point arithmetics are used
for calculation.

When auto mode is on, Mandelmania chooses either low or high precision
by itself.

1.39   Mandelmania/Menus/Recalculate

Recalculates the picture. If the buffer mode is on and the buffer contents
are valid, this is just a redraw.

1.40   Mandelmania/Menus/Defaultscreen

Sets the screen to its default value. The default values are in the file
Mandelmania.prefs or the file defined in the tooltype @{ " SETTINGS " link ←
    Settings}.

1.41   Mandelmania/Menus/Screen Mode

Opens a new screen. For this purpose a screen mode requester is opened
where you can enter the new screen parameters. You can select the
screentype in the listview gadget and the width, height and depth of the
screen. If you specify width and hight greater than the maximal visible
size, the screen will be autoscroll. You can also select the standard
amiga overscan modes using the cycle gadget. If the width and height are
smaller than the nominal screensize, they are automatically replaced by the
nominal values.

1.42   Mandelmania/Menus/Windowsize

This command lets you size the graphic window. You have to draw a
"rubberband" first. See also @{ " Zoom Mode " link MENU_Zoom_Mode}.

1.43   Mandelmania/Menus/Palette

If you choose this function you will see a color requester. The currently
selected color is on the left side of the palette gadget. You can select a
color by clicking in the palette gadget on the color. Use the three slider
gadgets to change the currently selected color. There are also certain
functions to modify the colormap:

```
Spread   :  spread colors between the currently selected and the next
            selected
Copy     :  copy the currently selected color to the next selected
Exchange :  exchange the currently selected color with the next selected
```

To cancel the operation, just press the right mouse button.

```
Undo     : undo the last operation
Ok       : accept the palette changes
Cancel   : change to the palette you had before
```

1.44   Mandelmania/Menus/Open Font

All Mandelmania requesters and windows are fully font-sensitive. To
change the screen font you can use this command. The screen will then
be closed and reopened, and a recalculation is executed unless you have
the buffer mode turned on.

1.45   Mandelmania/Menus/Paths

Lets you change different default paths (for pictures, animations, ARexx
scripts, palettes, preferences, form files and the ARexx IO window).

1.46   Mandelmania/Menus/Load Palette

Loads a new palette. This can be any file containing a CMAP chunk.
The filename can be selected by a file requester.

1.47   Mandelmania/Menus/Save Palette

Saves the current palette as the default palette "Mandelmania.palette".

1.48   Mandelmania/Menus/Save Palette As

Saves the current palette. The filename can be selected by a file requester.

1.49   Mandelmania/Menus/Load Settings

Loads a new settings. This can be any file containing a CMAP chunk.
The filename can be selected by a file requester.

1.50   Mandelmania/Menus/Save Settings

Saves the current settings as the default settings "Mandelmania.prefs".

1.51   Mandelmania/Menus/Save Settings As

Saves the current settings. The filename can be selected by a file requester.

1.52   Mandelmania/Menus/Delay

You can enter the cycling delay. The value you enter is the number of delays
between two color-rotations. So you have the maximal speed if you enter 0.
The colorcycling won't be started until you choose either @{ " Forward " link ←
   MENU_Forward} or
@{ " Backward " link MENU_Backward}.

1.53   Mandelmania/Menus/Forward

Starts the colorcycling. The colormap is cycled forwards.

1.54   Mandelmania/Menus/Backward

Starts the colorcycling. The colormap is cycled backwards.

1.55   Mandelmania/Menus/Off

Turns off colorcycling.

1.56   Mandelmania/Menus/Select Form

You can select a form file by a filerequester. A form file describes how
ARexx files are created when selecting the commands @{ " New " link MENU_New}, @{ ←
    " Append " link MENU_Append},
@{ " Interpolate " link MENU_Interpolate} and @{ " Close " link MENU_Close}. The ←
    default form file is
"CreateAnims.form" which is used for creating animations.

1.57   Mandelmania/Menus/New

Creates new ARexx scriptfile and writes ARexx header (@HEAD part of the form
file). Only one scriptfile can be opened at one time.

1.58   Mandelmania/Menus/Append

Adds the @APPEND part of the form file to the ARexx file under construction.
By default this is a routine which adds the current image parameters to the
ARexx script file. You must do this first before selecting @{ " Interpolate " ←
    link MENU_Interpolate}.

WARNING: The rangewidth (xright-xleft) or rangehight (ytop-ybottom) must
          be at least 1E-16. If the precision is higher a "Precision too
          high for animation!" error message is shown.

1.59   Mandelmania/Menus/Interpolate

Adds the @INTERPOLATE part of the form file to the ARexx file under
construction. By default this is a routine which interpolates between the
picture you appended before and the current picture. You are asked after
the number of steps you want to interpolate. All picture parameters are
interpolated (except type).

ATTENTION: In version 4.0, Lyapunov and CPM parameters are not yet inter-
           polated. This feature will be implemented in the next version.

1.60   Mandelmania/Menus/Close

Adds the @TAIL part of the form file to the ARexx file under construction
and closes it.

1.61   Mandelmania/Menus/Execute

Executes a ARexx scriptfile. You must run RexxMast (located in the System
folder of the Workbench) before selecting this command.

1.62   Mandelmania/ARexx Interface

Mandelmania opens an ARexx port called 'rexx_mandelmania'. You can send
the following ARexx commands to that port:

```
@{ " defaultscreen " link xdefaultscreen}
@{ " calculate " link xcalculate}
@{ " interpolate " link xinterpolate}
@{ " openfractal " link xopenfractal}
@{ " savefractal " link xsavefractal}
@{ " openanim " link xopenanim}
@{ " saveanim " link xsaveanim}
@{ " closeanim " link xcloseanim}
@{ " screenmode " link xscreenmode}
@{ " windowsize " link xwindowsize}
@{ " machine " link xmachine}
```


1.63   Mandelmania/ARexx Interface/defaultscreen

```
arguments : -
function  : Set the screen to its default value.
```


1.64   Mandelmania/ARexx Interface/calculate

```
arguments : xleft xright ybottom ytop xconst yconst iterations type
function  : Calculates Mandelbrot-Set with the specified parameters
```

If no arguments are specified, this command will recalculate the fractal image
using the last parameters.

1.65   Mandelmania/ARexx Interface/interpolate

```
arguments : nr steps xleft1 xright1 xbottom1 xtop1 xconst1 yconst1 iterations1
                   xleft2 xright2 xbottom2 xtop2 xconst2 yconst2 iterations2
                   type
```

```
function : Calculates one picture. This function is used to calculate
           interpolated pictures, where the set has <steps> pictures and we
           are currently calculating the picture <nr> of the series. The other
           parameters are for the definition of the start and the destination
           picture.
```

1.66   Mandelmania/ARexx Interface/openfractal

```
argument  : filename
function  : Load a fractal image
```

1.67   Mandelmania/ARexx Interface/savefractal

```
argument  : filename
function  : The current fractal image will be saved under the specified
            filename.
```

1.68   Mandelmania/ARexx Interface/openanim

```
argument  : animationfilename
function  : Opends a new animationfile
```

1.69   Mandelmania/ARexx Interface/saveanim

```
argument  : -
function  : Appends the current picture to the open animationfile
```

1.70   Mandelmania/ARexx Interface/closeanim

```
argument  : -
function  : Closes the open animationfile
```

1.71   Mandelmania/ARexx Interface/screenmode

```
Argument  : DisplayID Width Height Depth
```

```
Function  : Opens a new screen using the specified screen parameters.
            The fractal image won't be calculated
```

```
Example   : screenmode '8000'x 704 240 3
```

1.72   Mandelmania/ARexx Interface/windowsize

```
argument  : leftedge topedge width height
function  : Sizes the graphicwindow to the specified values. The fractal image
            is not calculated!
```

1.73   Mandelmania/ARexx Interface/machine

```
argument  : machinetype
function  : Sets the machinetype to either 68000, 68020 or FPU
```

1.74   Mandelmania/Animations

To calculate an animation, you have to go through the following steps:

```
 a) Start Mandelmania
 b) Choose a form file (@{ " Select Form " link MENU_Select_Form}) (not absolutely ←
     necessary, because
    Mandelmania loads a default form file during startup).
 b) Choose @{ " New " link MENU_New} from the ARexx menu and enter a new filename  ←
     for the
    scriptfile. If you don't specify fileextension ('.rexx'), they will be
    appended automatically.
 c) Now you must have a start picture. You can load one from disk or get one
    by zooming and scrolling through the fractal image.
    If you found a suitable start picture, you must choose @{ " Append " link  ←
       MENU_Append} from the
    ARexx menu.
 d) Now you need a destination picture.  You can get it similar to the start
    picture. If you got one, choose @{ " Interpolate " link MENU_Interpolate} from ←
         the ARexx menu. You are
    asked after the number of interpolations you want to calculate between
    the start and the destination.
 e) If you want to continue your animation, you can select another fractal
    image and interpolate multiple times.
    The destination picture of the last interpolation is automatically the
    start picture of the new interpolation.
    If you want a cut you have to append the picture after the cut.
 f) Choose @{ " Close " link MENU_Close} from the ARexx menu.
 g) You have created an ARexx scriptfile. Start it by choosing @{ " Execute " link ←
     MENU_Execute}
    or type rx <rexxfilename> in a shell and the animation will be
    calculated.
```

If you want to interrupt the calculation, send a Ctrl-C to the scriptfile and
then you can abort the current calculation (using the menu).

This is the only possibility to have a controlled interruption. All open
files will be closed and the animation is made loopable.

1.75 Mandelmania/Technical Information

Mandelmania 3.0 was compiled using AztecC 5.2. The very time-wasting
calculation routines are written in assembler.

```
                @{ " Special Chunks " link SpecialChunk}
                @{ " Precision " link Precision}
```

1.76 Mandelmania/Technical Information/Special Chunks

To store specific picture parameters, a special chunks are built in. The
most important one called "MAND". The structure of this chunk is as follows:

```
struct MAND_Chunk {
  USHORT LeftEdge,TopEdge,Width,Height;  /* windowcoordinates */
  long double xleft,xright,ybottom,ytop; /* range */
  long double xconst,yconst;             /* Julia constant */
  ULONG Iterations;                      /* max number of Iterations */
  ULONG Type;                            /* Type, see below */
};
```

Remember that a 'long double' is 12 Bytes (=96 bits) long.

Mand_Chunk->Type can be one of the following:

```
#define TYPE_MANDELBROT 0
#define TYPE_JULIA 1
#define TYPE_MANDELBROT_CPM2D 2
#define TYPE_JULIA_CPM2D 3
#define TYPE_MANDELBROT_CPM3D 4
#define TYPE_JULIA_CPM3D 5
#define TYPE_LYAPUNOV 16
```

I also need special chunks for saving the parameters for Lyapunov and
CPM fractals:

```
struct CPM1_Chunk {
  double LogScale;
  double LinScale;
  USHORT ColorPeriods;
  USHORT PlateauLevel;
  USHORT SeaLevel;
  USHORT Angle1;
  USHORT Angle2;
};

struct LYA1_Chunk {
  LONG LyaExpMin;
  LONG Settle;
  UBYTE Sequence[];
};
```

As you can see, the length of the "LYA1" chunk is variable.

I looked for an "official" standard for special fractal chunks, but I
didn't found. If there is a special chunk and you have further
information about it, please send it to me.

1.77   Mandelmania/Technical Information/Precision

There are 3 different precisions in Mandelmania: @{ " 16-, 32-bit integer " link ←
   Integer} and
@{ " FPU " link FPU}. For calculating FPU, you need a mc68881, mc68882, mc68040 ←
   or higher,
of course.

1.78   Mandelmania/Technical Information/Precision/Integer

In integer mode, internal coordinate calculation are made 64-bit wide (to
avoid some problems, for example instability of animations near the
precision limit). The fractal calculation is done either in 16-bit or
32-bit.

Mandelmania switches automatically between 16- and 32-bit precision.
But it can be also forced to calculate either low- or high-precision
by choosing @{ " Precision " link MENU_Precision } from the menus.

1.79   Mandelmania/Technical Information/Precision/FPU

All FPU numbers are extended real numbers (64-bit mantissa, 15-bit exponent).
If the numbers are converted to or from ascii (e.g. 'Enter range', 'Append',
'Interpolate') only double real numbers are used (C-restriction).

If your calculation mode is 68020 (integer), Mandelmania switches to FPU
mode, if higher precision is necessary and FPU available (and switches
back to 68020 mode if not necessairy any more).

It is different if you select FPU mode explicitly. Then you will always
stay in FPU mode. That's why the 68020-subitem is still checked, if
Mandelmania switched to FPU mode automatically.

Maximal precision is reached when the difference between two neighbourpixels
is 5E-19.

1.80   Mandelmania/Credits

Special thanks to Christian Schneider, Andreas Baumgartner, Leo Zimmermann,
Thomas Meyer and Stephan Babotai for beta testing.
Thanks also to all the people sending me bug reports, suggestions etc.