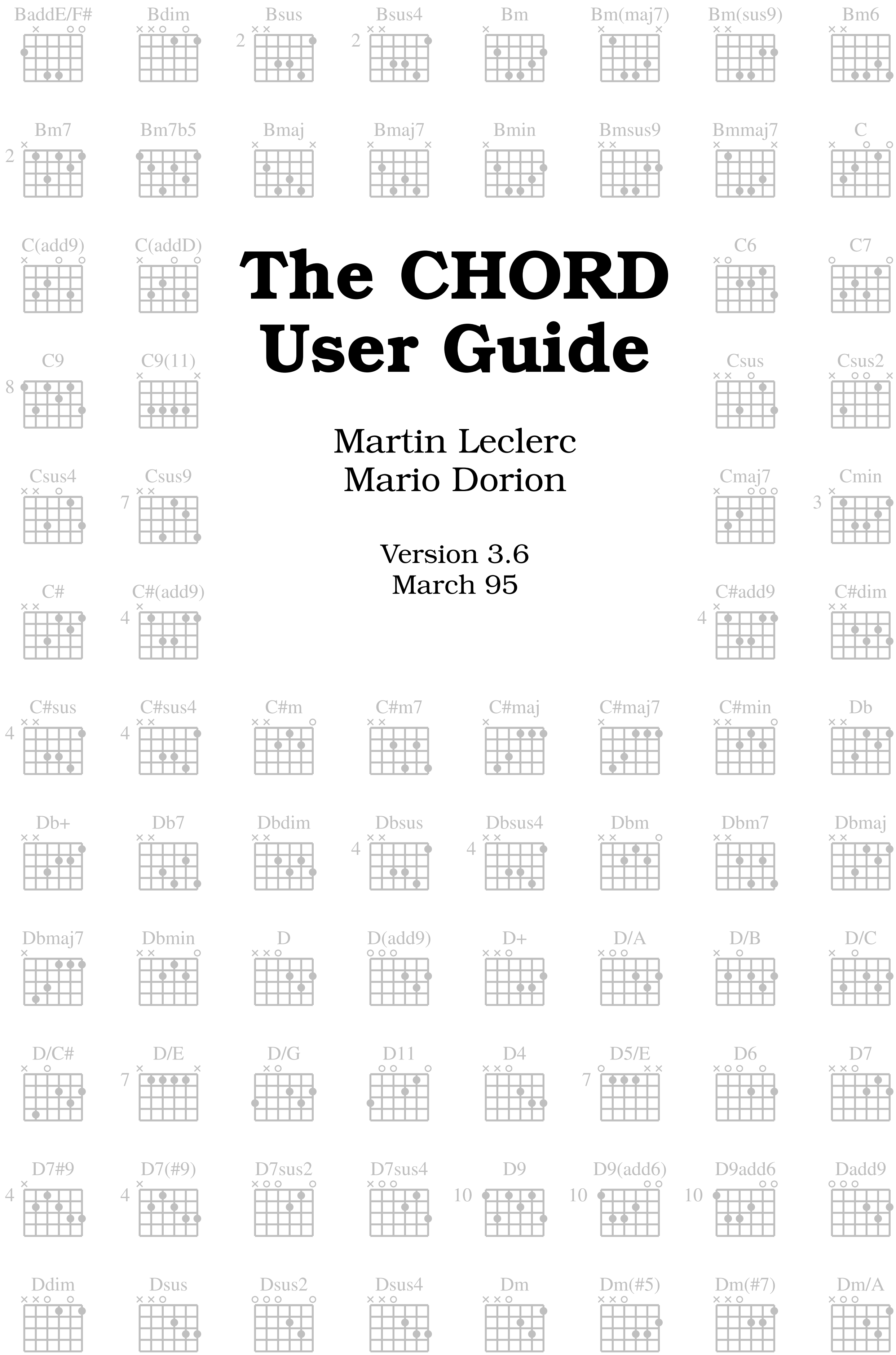


The CHORD User Guide

Martin Leclerc
Mario Dorion

Version 3.6
March 95

Welcome to the CHORD User Guide!

CHORD is a share-ware utility that was first created by lazy guitarists who got tired of turning pages in the middle of the songs they liked.

CHORD takes a file containing the lyrics and chords of a song and produces a PostScript version of that song. The song has now the following attributes:

Titles and sub-titles have been centered;
The lyrics appear in the font and size of your choice;
All chords name appear right above the right lyrics;
All chords used in your song appear as grids at the bottom of the page.

Optionally, you can also generate an index of your songs, have the pages numbered, have songs transposed up or down, print in 2-up or 4-up modes and insert tablature and comments

You have a great many options on the final appearance of you songs. All of them are described in the section titled “Options and Switches”

For now, please read on to better understand what CHORD can do for you.

An Example

This is an example of a “source” file for CHORD, i.e. a file that is meant to be processed by CHORD.

```
{title:Love me tender}
{subtitle:Presley/Matson}

[G]Love me tender [A7]love me sweet [D7]Never let me [G]go
[G]You have made my [A7]life complete [D7]and I love you [G]so

{start_of_chorus}
[G]Love me [B7]tender [Em]love me [G7]true
[C]all my [Cm]dreams ful[G]fill
[G]For [Dm6]my [E+]dar[E7]ling [A7]I love you
[D7]and I always [G]will [Am7] [D7]
{end_of_chorus}

[G]Love me tender [A7]love me long [D7]Take me to your [G]heart
[G]For it's there that [A7]I belong [D7]and we'll never [G]part

{comment:Chorus}

[G]Love me tender [A7]Love me dear [D7]Tell me you are [G]mine
[G]I'll be yours through [A7]all the years [D7]till the end of
[G]time

{comment:Chorus}
```

Here are the important points about that file:

It can be created using any text editor you like;

All characters present in the ISO 8859-1 character set can be used.

We will explain the content of that file. But first, let’s look at the end result, on the next page.

Love me tender

Presley/Matson

```
G              A7             D7           G
Love me tender love me sweet  Never let me go
G                A7            D7             G
You have made my life complete and I love you so
```

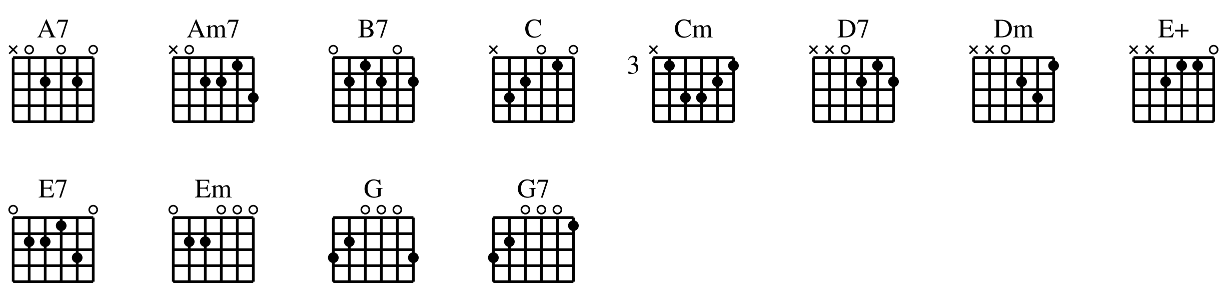
```
G       B7     Em      G7
Love me tender love me true
C      Cm     G
all my dreams fulfill
G   Dm E+ E7  A7
For my darling I love you
D7           G    Am7  D7
and I always will
```

```
G              A7           D7             G
Love me tender love me long Take me to your heart
G                    A7     D7              G
For it's there that I belong and we'll never part
```

Chorus

```
G              A7           D7              G
Love me tender Love me dear Tell me you are mine
G                     A7            D7              G
I'll be yours through all the years till the end of time
```

Chorus



How do I use CHORD?

Using CHORD is simple. four step are required:

1. Type in the lyrics of the songs: this can be done with any text editor.

```
% edit loveme
Love me tender love me sweet Never let me go
```

2. Insert the chord names between square brackets throughout the lyrics. The chord name will appear right above the letter that follows the closing bracket

```
Love me tender [A7]love me sweet [D7]Never let me [G]go
```

3. Insert any directive you want to give to CHORD: titles, sub-titles, chorus markers, comments, etc.

```
{title:Love me tender}
{subtitle:Presley/Matson}
```

4. Invoke CHORD with your file as an argument. This will produce the PostScript output. You can save this in a file for future printing, or send the output directly to the printer

```
% chord loveme >loveme.ps
```

or

```
% chord loveme | lp
```

Directives

A directive is a string of text that appears between french brackets (‘{‘ and ‘}’) and has a special meaning to CHORD. It contains a message that will affect the way CHORD processes your file. Examples of directive uses would be: changing the fonts, marking a chorus, and defining a title.
Directives must be alone on a line. Blanks before the opening bracket and after the closing bracket are not significant. Blanks inside a directive are ignored.

Supported directives are:

start_of_chorus or **soc**
: Which indicates the start of a chorus (yep). The complete chorus will be highlighted by a change bar, to be easily located by the player.

end_of_chorus or **eoc**
: Marks the end of the chorus.

comment: or **c**:
: Will call the printing of the rest of the line, highlighted by a grey box. (Useful to call a chorus, for example)

comment_italic: or **ci**:
: Will print the comment in an italic font... well not really, it will print the comment in the font used for printing the chord names (which is normally italic unless you specified a different chord_font).

comment_box: or **cb**:
: Will print the comment inside a bounding box.

new_song or **ns**
: Marks the beginning of a new song. It enables you to put multiple songs in one file. It is not required at the beginning of the file.

title: or **t**:
: Specifies the title of the song. It will appear centered at the top of the first page, and at the bottom of every other page, accompanied there by the page number, within the current song.

subtitle: or **st**:

Specifies a string to be printed right below the title. Many subtitles can be specified.

define: name **base-fret** offset **frets** str1...str6

Defines a new chord called “name”. The keyword “base-fret” indicates that the number that follows (“offset”) is the first fret that is to be displayed when representing the way this chord is played.

The keyword “frets” then appears and is followed by 6 values. These values are the fret number [1 to n] for each string [str1 to str6] and are RELATIVE to the offset. A value of “-”, “X” or “x” indicates a string that is not played.

Keywords **base-fret** and **frets** are mandatory.

A value of 0 for a given string means it is to be played open, and will be marked by a small open circle above the string in the grid. The strings are numbered in ascending order of tonality, starting on the low E (the top string). On output, a chord defined in the user’s .chordrc file will have a small asterisk near its grid, a chord defined in a song will have two small asterix. At the beginning of every song, the default chords are reloaded and the user’s .chordrc file is reread. Chord definition of new chords inside the text of a song are only valid for that song.

The syntax of a {define} directive has been modified in CHORD 3.5. CHORD will attempt to recognize an old-format **{define}** and will accept it. It will, though, print a warning inviting you to modify your input file to use the new syntax (the exact **{define}** entry to use is provided as an example).

textfont: postscript_font

Same as -T command option.

textsize: n

Same as -t command option.

chordfont: postscript_font

Same as -C command option.

chordsize: n

Same as -c command option.

no_grid or **ng**

Will disable printing of the chord grids for the current song.

grid or **g**

Will enable the printing of the chord grids for the current song (subject to the limitation caused by the usage of the -g option). This directive will override the runtime -G option for the current song.

new_page or **np**

Will force a logical page break (which will obviously turn out to be a physical page break if you are not in either 2-up or 4-up mode).

new_physical_page or **npp**

Will force a physical page break (in any mode).

start_of_tab or **sot**

Will cause chord to use a monospace (i.e.: non-proportional) font for the printing of text. This can be used to enter 'tab' information where character positioning is crucial. The Courier font is used with the same point-size as the rest of the text.

end_of_tab or **eot**

Will stop using monospace font. The effect is implicit at the end of a song.

columns: n or **col:** n

Specifies the number of columns on the pages of the current song.

column_break or **colb**

Forces a column break. The next line of the song will appear in the next available column, at the same height as the last "columns" statement if still on the same page, or at the top of the page otherwise.

Another example:

This example uses some of the more advanced directives:

```
{title: Everybody Hurts}
{subtitle: REM}
{define: G base-fret 1 frets 3 2 0 0 3 3}
{define: D4 base-fret 1 frets - - 0 0 3 -}
{define: E base-fret 1 frets - 3 3 2 0 0}
[D]Intro [G] [D] [G]

[D]When your day is [G]long and the [D]night, the night is [G]yours a[D]lone
[D]When you're sure you've had e[G]nough of this [D]life, well [G]hang on
{start_of_tab}
       E(low)-3-2-0-
{end_of_tab}
[E]Don't let yourself [A]go, [E]cause everybody [A]cries [E]and everybody[A]
hurts some[D]times [G]
Sometimes everything is [D]wrong, [G]now it's time to sing a[D]long
When your day is night alone [G] (hold [D]on, hold on)
If you feel like letting go [G] (hold [D]on)
If you think you've had too [G]much of this [D]life, well hang [G]on

[E]Cause everybody [A]hurts, [E]take comfort in your [A]friends
[E]Everybody [A]hurts, [E]don't throw your [A]hands, oh [E]now, don't throw
your [A]hands
[C]If you feel like you're [D4]alone, no, no, no, you're not [A]alone
{start_of_tab}
                         D4-> E-0-----0-----0-----0--
                              B---3-----3-----3------
                              G-----0-----0-----0----
{end_of_tab}
[D]If you're on your [G]own in this [D]life, the days and nights are [G]long
[D]When you think you've had too [G]much, with this [D]life, to hang [G]on

{start_of_chorus}
[E]Well everybody [A]hurts, some[E]times
Everybody [A]cries, [E]and everybody [A]hurts,[N.C.] ... some[D]times [G]
{end_of_chorus}
But everybody [D]hurts [G]sometimes so hold [D]on, hold [G]on, hold [D]on
Hold on, [G]hold on, [D]hold on, [G]hold on, [D]hold on
[G]Everybody [D]hurts [G] [D] [G]
[D]You are not alone [G] [D] [G] [D] [G]
#Picking pattern for the chords:
# D G
#E----------2-----------2-------------3-----------3-----------------------
#B--------3---3-------3---3---------3---3-------3---3---------------------
#G------2-------2---2-------2-----0-------0---0---------------------------
#D----0-----------0-------------------------------------------------------
#A------------------------------------------------------------------------
#E------------------------------3-----------3-----------------------------
```

See the result on the next page. Look for the tablatures and the modified definitions of chords G, D4 and E.

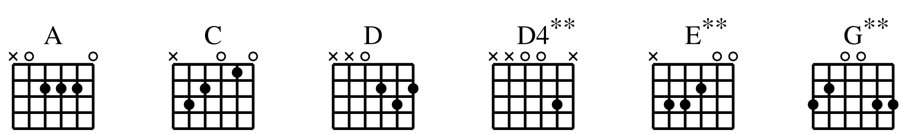
Everybody Hurts

REM

```
D       G      D      G
Intro

D                G            D                   G     D
When your day is long and the night, the night is yours alone
D                           G              D          G
When you're sure you've had enough of this life, well hang on
    E(low)-3-2-0-
E                  A   E                A     E             A          D    G
Don't let yourself go, cause everybody cries and everybody hurts sometimes
                           D     G                     D
Sometimes everything is wrong,   now it's time to sing along
                             G              D
When your day is night alone        (hold on, hold on)
                            G           D
If you feel like letting go       (hold on)
                            G            D               G
If you think you've had too much of this life, well hang on


E               A      E                    A
Cause everybody hurts, take comfort in your friends
E         A      E                A         E                     A
Everybody hurts, don't throw your hands, oh now, don't throw your hands
C                       D4                            A
If you feel like you're alone, no, no, no, you're not alone
              D4 ->    E-0-----0-----0-----0--
                       B---3-----3-----3------
                       G-----0-----0-----0----
 D                 G           D                             G
 If you're on your own in this life, the days and nights are long
D                             G               D             G
When you think you've had too much, with this life, to hang on
E              A      E
Well everybody hurts, sometimes
          A      E             A      N.C.      D     G
Everybody cries, and everybody hurts, ...      sometimes
              D     G                 D        G        D
But everybody hurts sometimes so hold on, hold on, hold on
     G        D        G        D
Hold on, hold on, hold on, hold on, hold on
G         D     G  D  G
Everybody hurts
D                 G  D  G  D  G
You are not alone
```



About the chords CHORD knows about:

CHORD knows of about 360 chords. Now, we KNOW your favorite chord won't be in there, and if it is, it will be all wrong. So we put in a facility for people to define their own chords. This facility will also let you redefine already defined chords. For the details, look in the "Directives" section of this guide, for the "define" directive.

About your preferences:

You can store your preferences about fonts and chords in a file that is unique to every user. This file is called ".chordrc" and should be in your home directory. For the list of the preferences you can have, see the "Directives" section.

Putting comments in your files

A line starting with a '#' is interpreted as a comment, and generates no output. (although all your comments are automatically mailed to the authors, and we read them at parties...)

Where to find CHORD files

CHORD having been around since 1991 there is a number of CHORD input files freely available if you have access to the Internet. **http://www.nada.kth.se/~f91-jsc** is the home page of a CHORD archive. **FTP.NEVADA.EDU** (131.216.1.11) contains chord and tab files for a great many pop and traditional songs. Many of the songs are in CHORD format. Look for file with the extensions CRD, .CHO, .CRDPRO, .CHOPRO or .CHORDPRO. The songs are in the directory /pub/guitar. This site is available for anonymous FTP.

The following USENET newsgroup are often used to post or request chord and tab files: **rec.music.makers.guitar.tablature**, **alt.guitar.tab**

The authors

CHORD has been developed by Martin Leclerc and Mario Dorion, both systems engineer with the Montréal (Canada) subsidiary of Sun Microsystems. We're still developing and supporting CHORD.

Feel free to contact us via e-mail to suggest improvements and report bug fixes. Our email addresses are **Martin.Leclerc@Sun.Com** and **Mario.Dorion@Sun.Com**

Copyright

The authors spent a great deal of time making it as feature-full and usable as possible. If you like this program, please take the time to send a postcard from your home town or locality to each of the two authors and let them know what you think of CHORD. Please send the cards to the following addresses:

Martin Leclerc	Mario Dorion
19 Browning	18 Georgia Crescent
Dollard des Ormeaux	Pointe Claire
Quebec, H9G 2K5	Quebec, H9R 5V7
CANADA	CANADA

How do I install CHORD on my machine?

CHORD is distributed in source format. That means you will need to compile it on your machine. Doing so requires a C compiler. This compiler can implement one of two norms about the C language: ANSI or K&R. Both types of compilers will do, but we prefer the first type.

The first step is to modify the file called “Makefile” to let the system know of some basic preferences:

1. If you are running on a system that has a routine called “getopt”, make sure the line that says

```
GETOPT = getopt.o
```

is commented OUT. To do so, add a ‘#’ at the very beginning of the line. If you do not know what we are talking about you should leave the line untouched.

2. Select the directory in which CHORD will be installed, by modifying the value of the variable BINDIR. In case of doubt, leave the line unchanged.

3.Select the directory in which the manual page for CHORD will be installed, by modifying the value of the variable MANDIR. In case of doubt, leave the line unchanged.

Save the Makefile file, and invoke the “make” utility

```
% make
```

This should build the CHORD utility correctly. If you have problems at this point, first try to contact someone who may be an experienced system administrator or programmer for assistance. If all else fails, send a detailed message to the authors, and they will do their best to help

If the “make” succeeded, invoke

```
% make install
```

to have the utility installed in the right spot. You could also manually install the CHORD binary in the location of your choice.

Included with the distribution are two sample files, under the “songs” subdirectory.

Options and Switches

Here are the options you can specify when invoking CHORD;

-A

Will print the “About CHORD...” message.

-a

Automatically single spaces lines that have no chords.

-c chord_font size

Sets the size, in points, of the font used to display chords to the specified integer value.

-C chord_font

Sets the font used to print chords to the specified name. That name must be known to your PostScript Interpreter.

-d

Generates a text Chord Chart of all internally known chords as well as chords defined in the $HOME/.chordrc file. Chords defined in the .chordrc file are identified with the “(local)” caption. The printout is suitable for input to the .chordrc file.

-D

Generates a PostScript Chord Chart of all internally known chords as well as chords defined in the $HOME/.chordrc file. chords defined in the .chordrc file are identified with a small asterisk after the chord grid.

-s grid_size

Sets the size of the chord grids.

-G

Disable printing of the chord grids for the whole input file(s). The effect can be disable for any particular song by the usage of the grid or g directive.

- g

Disable printing of grids for “easy” chords.

Whether a built-in chord is easy or not has been arbitrarily decided by the authors. The general rule was that any chord in its major, minor, 7th or minor 7th was “easy” while everything else (maj7, aug, dim, sus, etc...) was “tough”. All chords defined in the $HOME/.chordrc file or in the input file are defined as “tough”.

-h

Prints a short options summary.

-i

Generates a table of contents with the song titles and page numbers. It implies page numbering through the document. Index pages are not numbered.

-L

Places the odd page numbers on the left.

-l

Prints only the lyrics of the song.

-o filename

Sends output to <filename>.

-p first_page

Numbers the pages consecutively starting with first_page (e.g. 1). Without this option, each song restarts the page numbering at 1, and page numbers are only put on subsequent pages of multiple page songs.

-P paper_size

Sets the paper size. Recognizes “us” and “a4”.

-R

Places the odd and even page numbers in the lower right and left corners respectively (for two-sided output). The default is all page numbers on the right.

-t text_font size

Sets the size, in points, of the font used to display the lyrics to the specified integer value. The title line is displayed using that point size + 5. The sub-title is displayed using that point size -2.

-T Text_font

Sets the font used to print text to the specified name. That name must be known to your PostScript Interpreter.

-V

Prints version and patch level

-x halftones

Sets up transposition to that number of halftones. Can not be zero. All chord names must be build in the following way in order to be recognized:

{note-name}[#|b][^/]* ['/' {note-name}[#|b][^/]*]

That is, a valid note name, possibly followed by '#' or 'b', followed by other modifier ('7', 'm', etc...). Many such construct can make up a chord name, as long as they are separated by '/'.

{note-name} must appear in the list 'A','B','C','D','E','F','G'.

-2

Prints two logical pages per physical page.

-4

Prints four logical pages per physical page.