TK-707

The TK-707 Rhythm Composer
Version 0.8
3 November 2004

by Chris Willing and Pierre Saramito

1 Installation

The TK-707 installation requires:

- C compiler: to compile the source code.
- TCL/TK version 8.0 or later. Tested up to TCL/TK 8.3. Available free from http://www.scriptics.com
- Alsa: audio installation (at least version 0.5.0), available free from http://www.alsa-project.org
- a sound card supported by Alsa.

and, at least one of these MIDI renders:

- an internal MIDI render hardware in your card.
- an external MIDI module with drum sounds (Tested Roland D-110 and TR-707).
- or any Alsa supported basic sound card together with the `timidity` or the `iiwusynth` MIDI synthesis software installed.

Clearly, if you have neither a MIDI card nor an external MIDI module, you can successfully manage MIDI files by a MIDI software sequencer. See below.

So far tested on

- Linux - Debian 3.0 (kernel 2.4.18) with SB Live ! sound card
- Linux - Slackware 7.0 (kernel 2.2.14) with ESS1868 sound card
- Linux - Mandrak (kernel 2.2.13-7mdk)
- Linux - Redhat 6.0 (kernel 2.2.5) with Ensonic 1370 sound card

1.1 Building `tk707`

```
configure
make
```

The program consists of Tcl/Tk for the gui and C code for outputting the midi data. The Tcl/Tk code is compiled with the C code. The compiled target `tk707` is a self contained binary.

1.2 Configure options

During the configuration step, you could also provide to the `configure` script some non-standard locations of the Alsa or Tcl/Tk libraries. On my Debian 3.0 installation, I use:

```
configure --with-tcl-includes=/usr/include/tcl8.3
```

A more general example writes:

```
configure --with-alsa-prefix=/usr/sound/lib \
  --with-alsa-inc-prefix=/usr/sound/include \
  --with-tcl-includes=/new/include   --with-tcl-libs=/usr/new/lib  \
  --with-tk-includes=/usr/new/include --with-tk-libs=/usr/new/lib
make
```

Enter `configure --help` for a complete list of available options.

It could also be useful to send compiler options to the `configure` script.

```
CFLAGS="-O0" configure
make
```

Indeed, by default, `gcc` compiler optimization flags are set to the maximum, i.e `-O9`. This is not supported for old `gcc` versions, where this feature failed on some combinations of Linux kernel versions and `gcc` version (e.g. Mandrake with kernel `Linux-2.2.13-7mdk` and `pgcc-2.91.66`). Either update our environment or use lower optimization flags.

1.3 Installation

```
make install
```

The default installation directory is `/usr/local/bin` but this can be changed by using an extra configuration option:

```
configure --prefix=/users/angela/music
make install
```

will build and install `tk707` into the installation directory `/users/angela/music/bin` as `tk707`.

1.4 Running

If installation was done as in previous step, and assuming the installation directory is in your execution path, then the program is started with

```
tk707
```

A sample song is in `demo.dat` which can be loaded from the `File->Load Demo` menu. After loading, click the `START` button and you should hear something if your hardware is set up correctly.

When `tk707` is first started, a suitable output port needs to be selected. A selection gui comes up before the main TK-707 window to make this choice. The list on the right hand side should contain one or more entries from which to choose (don't worry about the input port list on the left). Double click on the available output port you want to use, and it should be entered into the Selected Output box near the bottom. Now you can click on the OK button to use the selected output port for this invocation of TK-707. You can also click the SAVE button first to save this information (in `~/.tk707rc`) so that you don't need to make the selection next time you run TK-707. Note that `~/.tk707rc` is a plain text file which can be edited by hand if you want to.

Alternatively, if you know what port you want to use, run `tk707` with the `-p` option, e.g.

```
tk707 -p 65:0
```

Use of the `-p` option overrides any setting in the `~/.tk707rc` file.

A list of suitable ports can also be obtained with the `-l` option, e.g.

```
tk707 -l
```

1.5 Building the documentation

The documentation in `.info` format, suitable for emacs, is contained in the distribution. You can browse it:

```
info -f tk707.info
```

If you have the `texi2html` tool, the `html` version has been generated during the installation:

```
netscape tk707_toc.html
```

If you have TEX installed, a pretty version of the documentation is available:

```
make dvi
xdvi tk707.dvi
```

and also for printing:

```
dvips tk707.dvi -o tk707.ps
lpr tk707.ps
```

Now, read the documentation and enjoy `tk707`.

1.6 Comments, bug corrections and future versions

We are keen for people to try it and send comments and suggestions.

Please, send comments, corrections, additions, bugs etc.. to chris@vislab.usyd.edu.au and pierre.saramito@imag.fr

Future versions from the TK-707 home page:

- Australie: http://www.vislab.usyd.edu.au/staff/chris/tk707
- Europe: http://www-lmc.imag.fr/lmc-edp/Pierre.Saramito/tk707

1.7 My card has no midi render...

Don't worry, you are able to hear midi file with a basic audio sound card: the `timidity` code is able to provides an Alsa sequencer interface which receives events and plays it in real-time.

`timidity` is a public domain code available at http://www.goice.co.jp/member/mo/timidity.

On this mode, `timidity` works purely as software (real-time) midi render. There is no scheduling routine in this interface, since all scheduling is done by Alsa sequencer core.

So, download `timidity` (at least version 2.10.0) and install as follow:

```
          configure --enable-server --enable-alsaseq --enable-alsatest \
--enable-audio=alsa,wav,au,aiff,list,oss
make install
```

`timidity` software emulates the best sound cards. It requires some Gravis Ultra Sound (GUS) patches describing musical instruments. The public domain EAW sound patch suite can be downloaded at http://www.stardate.bc.ca/eawpatches/html/default.htm This his a complete and high quality library, updated frequently. Especially, download the `gsdrums` percussion instrument library.

Install these files in the `timidity` directory, usually `/usr/share/timidity`.

For invoking Alsa sequencer interface, run `timidity` as follows:

```
timidity -iA -B2,8 &
```

The fragment size can be adjustable. The smaller number gives better real-time response. Then `timidity` shows new port numbers which were newly created (128:0 and 128:1 below).

Finally, run `tk707`:

```
tk707 -p 128:0
```

The `timidity` render is very flexible and rich, since you can add any instrument in your library by using GUS patch files. This modular approach is not always supported by all hardware renders. Thus, the software MIDI render solution is fruitful, and could be installed, even if you have a hardware or external MIDI solution.

2 Getting started

2.1 Terminology

Four song *tracks* are available. You can think of a track as a song. Tracks are labeled by roman numbers: I, II, III and IV.

Each track is made of *patterns*, each with 16 *steps*.s You could think of a pattern as a bar of music. At each step in a pattern, any of the available instruments can be set to play. Once you've made the patterns you want to use, you can combine them in any order into any of the four available tracks. Any particular pattern can be used in any of the tracks as many times as desired.

There are 64 patterns available in 4 *groups* of 16 each. Pattern groups are labeled by letters: A, B, C and D. The 4 groups are not at all related to the 4 tracks. Patterns from any group are available to any track.

2.2 Demonstrations

Some sample patterns and tracks are in `demo.dat` which can be loaded from the `File->Demo` menu. Something exists in each track of this file so click the `START` button and you should hear something if your hardware is set up correctly. Part of this set up is the *midi channel*; the default midi channel is 10 (the default for D-110 drum sounds). You can change the output midi channel via the Midi menu or Shift-click on the `MIDI CH` button; a new window will appear for midi channel selection.

A second demonstration file shows the alternate *727* latin sound map. With this sound map, the software emulates a Roland *727* latin percussion rhythm composer. Loaded from the `File->Demo` menu the file `son-montuno.dat`, and then load the latin percussion sound map `727.map` from the `Map->Load Standard Sound Map` menu. Set also the tempo to 200 bpm. Then, click the `START` button.

An alternate sound map for latin percussion is `7c7.map`. The `c` in `7c7` stands for cuba[1]. can be loaded in the `Map->Load Standard Sound Map`. This file has no Roland latin percussion rhythm composer equivalent, and has been designed for our pleasure. Click the `START` button and you will hear the file `son-montuno.dat` with another feeling.

Finally, a brazilian sound map `7b7.map` is presented. The `b` in `7b7` stands for brazil[2] and can be loaded together with the `carioca.dat` demonstration. A good tempo is 220 bpm for the samba carioca.

There is no more demonstration file for these sound maps, since it is time for you to compose your music !

2.3 Creating patterns

[1] An excellent introduction to afro-cuban rhythms can be founded in 'Les tumbaos de la salsa', by Daniel Genton, Editions Musicales Francaise, 2000. email: Topdjembe@hotmail.com.

[2] This sound map is indented for batucada music for carnaval.

2.3.1 Writing pattern

Shift-click the PATTERN button to enter Pattern Write mode. While START is not pressed, the 16 instrument/step keys at the bottom of the interface are dual purposed.

Clicking on a key changes which pattern, 1-16, (within the current group) is selected for writing. A small lamp above one of the *group pattern* buttons indicates the current group. The current group may be changed by clicking group pattern button A, B, C or D.

Shift-clicking on the *instrument/step* keys selects which instrument is selected to be written into the current pattern. Click the START button; now as you click of the instrument/step keys, the selected instrument, e.g. Cowbell, will be added at that step. If the selected instrument has already been set at that step, then clicking there will remove it. To select another instrument, press the STOP/CONT button then select another instrument using Shift-click on the appropriate instrument/step key.

Continue adding instruments at the required steps to complete the pattern. Make all the patterns you will need for a new song.

A pattern may be cleared of all notes while in Pattern Write mode; clicking on the CLEAR button will raise a dialog window to confirm clearing of the current pattern.

2.3.2 Copy/paste and copy/merge

Another method of creating new patterns is to copy an existing pattern and pasting it to a new location, then modifying the new version. This is an efficient method of creating a number of patterns which are small variations of each other. Select the pattern to be copied and press CRTL-C (CONTROL key and C key together).

Now select a new (empty?) pattern location and press Ctrl-V or Ctrl-B to *paste* in the copied notes.

Using Ctrl-V will clear the destination location of any notes it already contains whereas Ctrl-B will *merge* its current contents with the new note information.

2.3.3 The pattern display grid

The third method uses the pattern display grid. Double-click near an instrument/step intersection to add the instrument at that step. If the instrument already exists at that step then it is deleted. This is probably the easiest way of creating patterns.

2.4 Playing patterns

Click the PATTERN button to enter Pattern Play mode. While the START button is depressed, the currently selected pattern will play. Clicking on different pattern keys at the bottom of the interface will select a new pattern (1,..,16) in the current group. If not currently playing, the new pattern is displayed immediately. If a pattern is already playing, the new pattern will play when the current pattern is finished.

Selecting a different group also changes the current pattern, i.e. if pattern 7 in group I is selected, then when group 2 is selected the new pattern will be pattern 7 in group II. Sixteen patterns are available in each of the four groups for a total of sixty-four patterns.

2.5 Creating tracks

Shift-click the TRACK button to enter Track Write mode.

To start from an empty track, delete anything already there (Shift-click on the CLEAR button) if necessary. Now select a pattern to add to the track and click on the enter key as many times as that pattern is required. Then select the next pattern to add and click on the enter key as many times as that pattern is required.

E.g., selecting pattern 3 and clicking on the ENTER button 4 times, then selecting pattern 9 and clicking on the ENTER button 2 times will result in a track of six measures - 4 of pattern 3 followed by 2 measures of pattern 9.

Changes to a track being created can be made. See Section 2.6 [Editing tracks], page 7.

The currently selected pattern can be heard playing if the START button is clicked. This enables an audible preview of selected patterns prior to adding them to the track.

2.6 Editing tracks

Shift-click the TRACK button to enter Track Write mode.

Track editing is only possible in Track Write mode.

Deletions of tracks (or parts of them) require confirmation (via a dialog window).

Shift-click on the CLEAR button deletes all of the current track.

Ctrl-click on the CLEAR button deletes the current measure from the track.

Shift-Ctrl-click on the CLEAR button deletes the rest of the current track, including the current measure. On some systems (including mine!) this actually requires the Ctrl key as well i.e. Shift + Ctrl keys + mouse click. Consider this a minor bug to be fixed sometime.

Adding patterns to a track involves selecting a pattern then a click or Shift-click on the ENTER button. Think of Shift-click on ENTER as an insert operation into the track before the current measure. An ordinary click on ENTER is actually a replacement operation i.e. it will replace the current measure with the currently selected pattern. However if you're at the end of the current track this behaves like an append operation - it adds new measures to the end of the track.

The currently selected pattern can be heard playing if the START button is clicked. This enables an audible preview of selected patterns prior to adding them to the track.

Different parts of the track can be accessed by clicking on the BACK and FWD buttons. Shift-click on the LAST MEAS button to go to the end of the track. Note that this goes to the measure just past the end of the track (ready to write the next measure). An ordinary click on the LAST MEAS button will momentarily show the last (+1) measure while the button is held down.

2.7 Playing tracks

Click on the TRACK button to enter the Track Play mode. Then click on the START button and the current track will be played.

Select another track, e.g. track IV, by Shift-clicking a the corresponding track button: Shift-IV. Then enter START for playing the song.

2.8 Tempo changes

The tempo may be changed in either Track or Pattern mode. The tempo is lowered by Clicking in the tempo dial and then Click-Dragging around. It its increased by Click-Dragging to the left from the tempo dial; its increased by Click-Dragging to the right.

Tempo changes can't be written into a track. If saving a track as an midi file, the whole track will have whatever tempo is selected at the time of saving.

3 Midi files and sounds

3.1 Midi files

Currently only single track Midi files can be exported. When the `Midi File->Save` is selected a file browser appears into which the name of the file is entered. Its not crucial but it is a convention to have a `.mid` ending on the filename. When a name has been nominated, the current track will be exported, including midi channel and tempo information i.e. make sure these have the values you want first.

Tempo changes can't be written into a track. If saving a track as an midi file, the whole track will have whatever tempo is selected at the time of saving.

Volume controls are used when saving a midi file: master, accent and volume controls associated to instruments.

3.2 Sound maps

TK-707 *sound maps* are mappings of the 16 instrument keys to descriptions of their long names, short names, abbreviate name and midi note values. The first (longer) names are used in the pattern grid display, while the second (shorter) names are used under the 16 instrument keys and abbreviations are used under the volume controls. The midi note values need to correspond to values recognized by the sound module being used with TK-707 (either external or built into a sound card). Appropriate values will need to be gleaned from your hardware documentation, however the built in values should work on any module conforming to the General Midi standard.

The built-in sound map is supplied as an example in the `707.map` file. A second file, `727.map`, is supplied and can be loaded to play with the sounds of Roland's TR-727, the latin percussion counterpart of the TR-707. Note that the 727 mapping won't work on a 707 because it doesn't recognize the necessary range of midi note values. It will work on modules (such as Roland D-110) which do recognize the necessary range.

Files with similar layout to the supplied `.map` can be constructed using a text editor and loaded into TK-707 from the `Map->Load Local Sound Map` menu.

Alternatively, the currently loaded sound map can be edited via an editing interface which is accessed from the `Map->Edit Sound Map` menu. Just change the name, short name and midi note values as required and press the `APPLY` button to apply the new settings to TK-707. The Cancel button will revert sound setting to whatever they were before the editing window was started. The `OK` button will apply the currently edited settings and close the editing window.

As new midi note values are entered for an instrument, the sound that they access in the midi sound module can be previewed using the `TEST` buttons. To speed the process of finding suitable sounds, three keyboard/mouse shortcuts are provided:

- `Shift + Mouse button 1` in the "Note" widget increments the value by 1 and test plays the new sound;
- `Ctrl + Mouse button 1` in the "Note" widget decrements the value by 1 and test plays the new sound;

- `Mouse button 2` in the "Note" widget and dragging up or down continuously decrement or increments the note value and test plays. the new sound.

Any editing of sound maps are lost if TK-707 is closed down without saving the sound map. A sound map can be saved using the `Map->Save Sound Map` menu item. A file suffix of `.map` is recommended (but not absolutely required).

4 Ternary feelings and poly-rhythms

4.1 Scale changes

This feature is related to fast figures and 6/8 figures.

Four scales are available on each pattern. Click on the `SCALE` in Pattern Write mode for changing the scale of the current pattern. A small lamp on the left of the musical score score indicates the current scale. For each scale, a vertical bar indicates the duration of a whole note.

4.1.1 Binary 4/4 measures

On the first scale, the whole note duration is decomposed into sixteenth. Each sixteenth is associated to the a step of the pattern. There is 16 sixteenth in the pattern if the last step is set to 16, and then the total duration of the pattern is those of a whole note.

This is the default scale when creating pattern associated to a 4/4 measure.

4.1.2 Binary 2/4 measures

For the second scale, the whole duration is decomposed into eighthes. Each eighthes is associated to a step of the pattern. When the last step is set to 16, there is 16 eighthes in the pattern, and the total duration of the pattern is those of a half whole. When playing such pattern, lamps associated to steps are turning twice faster ! The pattern is then a 2/4 measure. The whole duration should be recovered in a song by inserting two of such patterns.

This scale is designated for fast schemes on binary songs.

4.1.3 Triplets and composed 12/4 measures

On the third scale, the whole duration is decomposed into four third of a quarter. Each third of a quarter is associated to a step of the pattern.

When the last step is set to 12, the total duration of the pattern is those of a whole note see Section 4.2 [Last step changes], page 12. The pattern is then a 12/4 measure.

Alternatively, the last could be set to 9 or 15 and then the pattern is a 9/4 or a 15/4 measure, respectively.

This pattern is designated for ternary-based measures or for inserting ternary feelings (triplets) into a binary song.

4.1.4 Sextuplets and composed 12/8 measures

Finally, the fourth scale decomposes the whole duration into two sextuplets (12 times 1/6 of a quarter sextuplets). Each 1/6 of a quarter is associated to a step of the pattern. When the last step is set to 12. See Section 4.2 [Last step changes], page 12, the total duration of the pattern is those of a half whole note. The whole duration should be recovered in a song by inserting two of such patterns. The pattern is then a 12/8 measure.

Since each quarter is represented by 6 steps on the pattern, we are able to superpose binary and ternary schemes, i.e. poly-rhythmic effects.

The `son-montuno.dat` demonstrates such effects.

This pattern is designated for fast schemes on ternary based measures or for inserting ternary feelings (triplets) into a binary song.

4.2 Last step changes

Remarks that the last step is not automatically changed from 16 to 12 when changing the scale of a pattern. It could be chosen to 15 for the scale 3, for instance.

Click on the `LAST STEP` in Pattern Write mode for changing the scale. A new window will appear for last scale selection of the last step associated to the current pattern.

5 Accents and flam

5.1 Accents

5.1.1 Inserting accentuated notes

There is two available accents: a weak and a strong one. Use the weak accent by clicking on the `ACCENT` button in Pattern Step Write or Pattern Tap Write modes. The "accent" text at the bottom of the button becomes orange. Enter then notes as usual: notes are *weakly accentuated* and the circle marker is orange, instead of the usual grey.

Use the strong accent by clicking a second times on the `ACCENT` button. Now, the "accent" text becomes red. Then, enter notes: they are *strongly accentuated* and the circle marker is red too.

Then, click a third times on the `ACCENT` button. The "accent" text at the bottom of the button comes back to the initial foreground color. When you insert notes, they will be no more accentuated.

In Pattern Step Write mode, you could also insert notes by using the `1`,..,`16` step touchs. Conversely, in Pattern Tap Write, you could also use the instrument keys, e.g. `Cowbell`.

5.1.2 Controlling the accent intensity

The weak and strong accent intensities could be modulated by using the volume control labeled "AC". Moves the label control up and down. When up, accent effects are maximum while when down, there is no more perceptible accent effects. The volume control of accents is global.

5.2 Flam

5.2.1 Inserting flams

Flam is a kick note written ahead of the main beat, and takes on the effect of a *grace note*. Control-double-click (`Control-Button-1` two times) near an instrument/step intersection to insert a flam note for this instrument at that step. Instead of the circle marker, a star marker represents the flam note at the selected intersection see Section 2.3 [Creating patterns], page 5.

There is another way for inserting flams: In Pattern Step Write mode, click the `START` button; Select an instrument by shift-clicking on an instrument/step key, e.g. `Shift-Snare`. Now, as usual, as you click on the instrument/step keys, the selected snare will be added at that step 4. Instead of this, if you control-click on the instrument/step keys, e.g. `Ctrl-4`, then a snare note with a flam is inserted at step 4.

Conversely, in Pattern Tap Write mode, enter a flam by using the `Ctrl-Cowbell`.

Flam could be combined with accents, and the associated star marker becomes orange or red, depending on the weak or strong accent.

5.2.2 Controlling the flam interval

The time between the grace note and its following main note can be varied in the range 0:4. The default value is 2. A zero value means that flam is off. Each pattern has an associated flam interval value.

Click on the `FLAM/SHUFFLE` button in Pattern Write mode for changing the flam interval of the current pattern. A new window will appear for the flam interval selection.

Appendix A Command line options

A.1 Synopsis

```
tk707 options...
```

A.2 Options

`-l` List the available sound ports

`-p` *port* Use this port

`-small`

`-medium`

`-normalsize`
Set the size of the window, since some screens are too small for the whole window.

`-display` *screen id*
Display to use.

`-use` *window id.*
Id of window in which to embed application. See `xwininfo`.

`-geometry`
Initial geometry for window

`-colormap`
Colormap for main window

`-name` Name to use for application

`-sync` Use synchronous mode for display server

`-visual` Visual for main window.

`-h`

`-help` Print short help.

Appendix B Converting MIDI to audio WAV and MP3::

This small annex explain how to convert your output midi file to formatted audio file (ex. RIFF WAVE) and compressed MP3 audio files. We assume your have converted your file `salsa.dat` into MIDI format `salsa.mid` in the `tk707` environment see Section 3.1 [Midi files], page 9.

Then enter the commands:

```
timidity bolero.mid -Ow bolero.wav
lame -b128 bolero.wav bolero.mp3
```

and compare the file sizes:

```
ls -al
-rw-r--r--   1 maria martinez     1051 Mar  9 09:20 son-montuno.mid
-rw-r--r--   1 maria martinez  3937324 Mar  9 09:22 son-montuno.wav
-rw-r--r--   1 maria martinez   493056 Mar  9 09:22 son-montuno.mp3
```

The MIDI file does not contains the sounds. It contains only the score of your music and a MIDI render is required to heat it. The audio WAV and MP3 contains the complete sound and are suitable exports, such as CD-ROM writers. Moreover, the MP3 file is compressed by rougthly a factor ten. The MP3 audio files can be played back by popular mp3 players such as mpg123.

The `timidity` tool is a midi-to-wav converter available at http://www.goice.co.jp/member/mo/timidity.

The `lame` tool is a wav-to-mp3 convarter available at http://www.sulaco.org/mp3.

Appendix C GNU General Public License

Version 2, June 1991

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

   Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

   These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
   a. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
   b. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
   c. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

   The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

   If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

   If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

   It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

   This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

   Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.
10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software

which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
one line to give the program's name and an idea of what it does.
Copyright (C) 19yy  name of author

This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details
type ‘show w’.  This is free software, and you are welcome
to redistribute it under certain conditions; type ‘show c’
for details.
```

The hypothetical commands ‘`show w`’ and ‘`show c`’ should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than ‘`show w`’ and ‘`show c`’; they could even be mouse-clicks or menu items—whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright
interest in the program ‘Gnomovision’
(which makes passes at compilers) written
by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Index

Short Contents

Table of Contents