0.1 Introduction

This document tests all kinds of features, from simple to advanced, that are not really suited for the reference manual, and are not needed as a regression test.

Here you may also find dirty tricks, or very the very latest features that have not been documented or fully implemented yet.

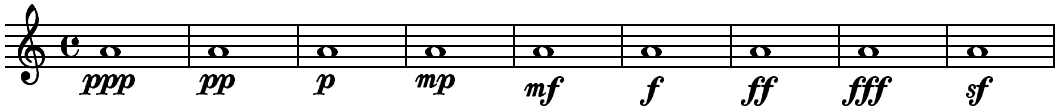
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/+.ly`’:

.

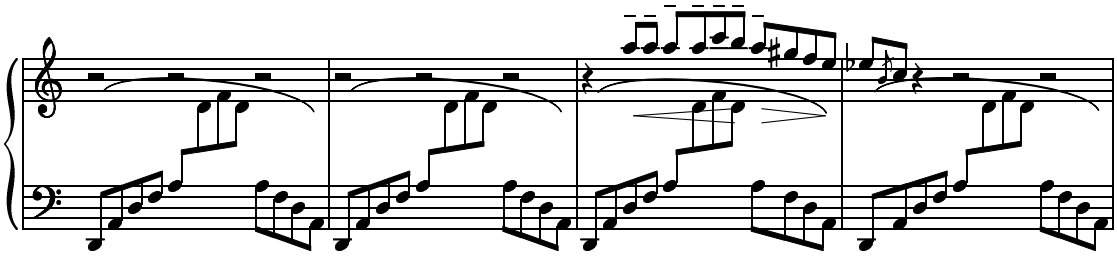
Jazz chord names, but with lower case names for minor chords ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/i`

C columnsC columnsC7

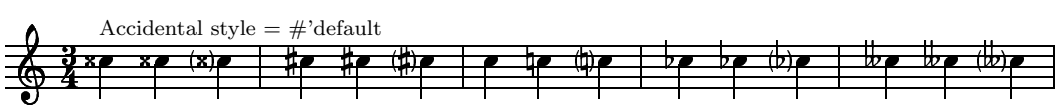
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/absolute-volume.ly`’:

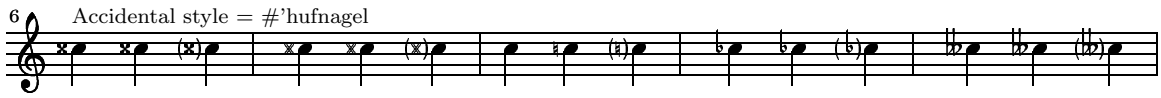


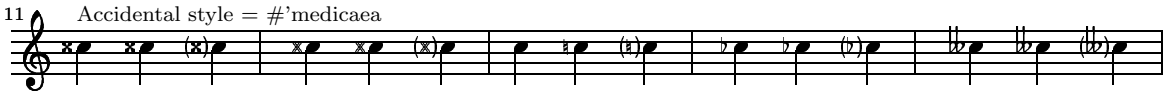
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ac-extra-voice.ly`’:

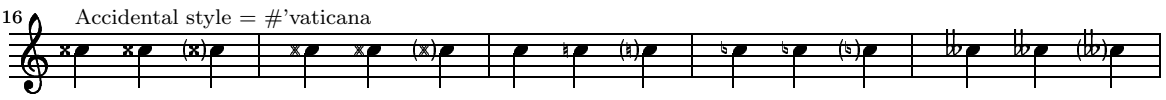


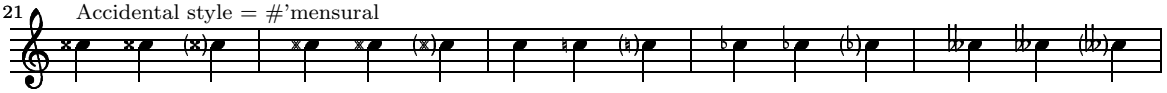
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/accid.ly`’:



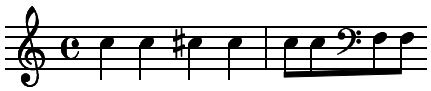




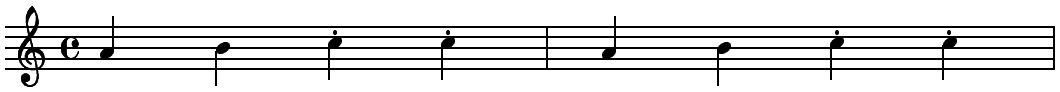




'/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/accidental-spacing.ly':

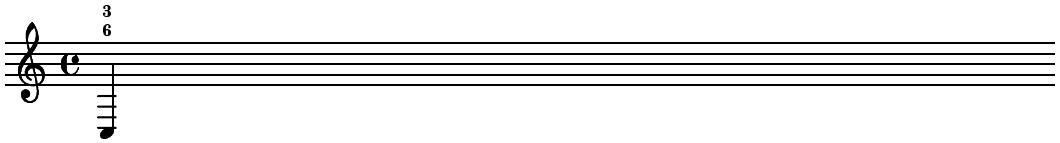


Using make-music, you can add various stuff to notes. Here is an example how to add staccato dots. Note: for this simple case one would not use scm constructs. See separate-staccato.ly first. '/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/add-staccato.ly':

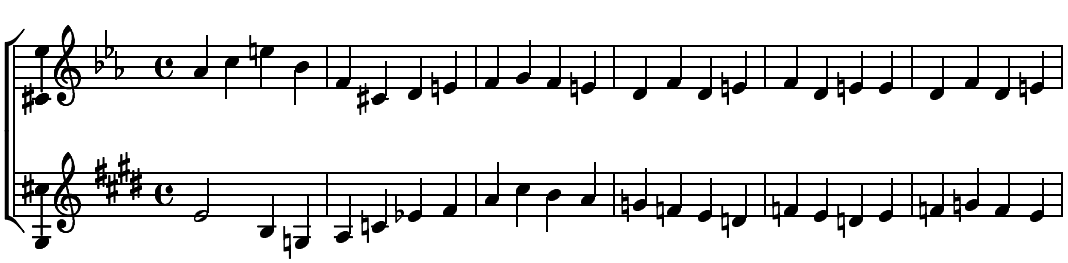


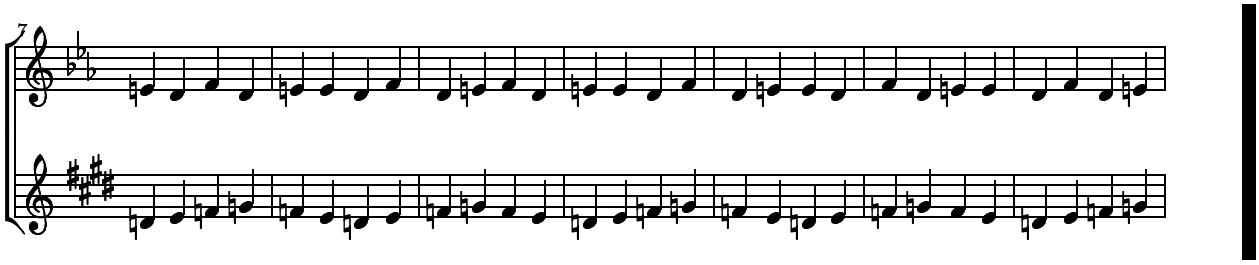
Using make-music, you can add various stuff to notes. Here is an example how to add an extra fingering.

In general, first do a display of the music you want ot create, then write a function that will build the structure for you. '/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/add-text-script.l

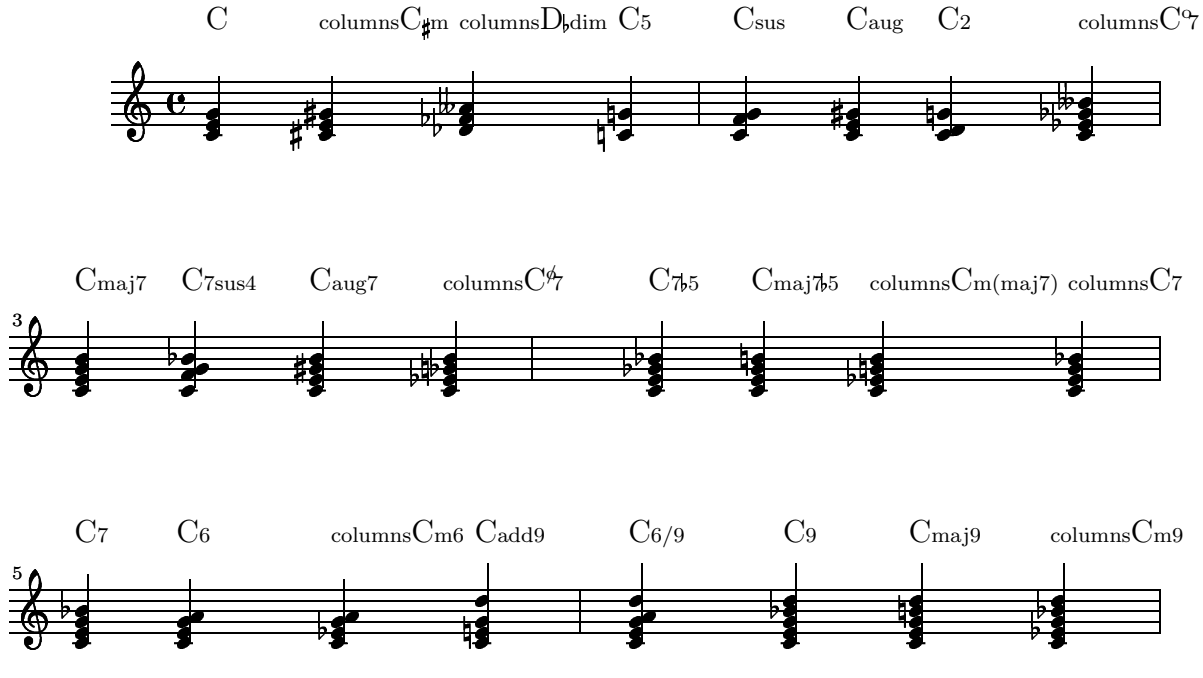


'/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ambitus.ly':

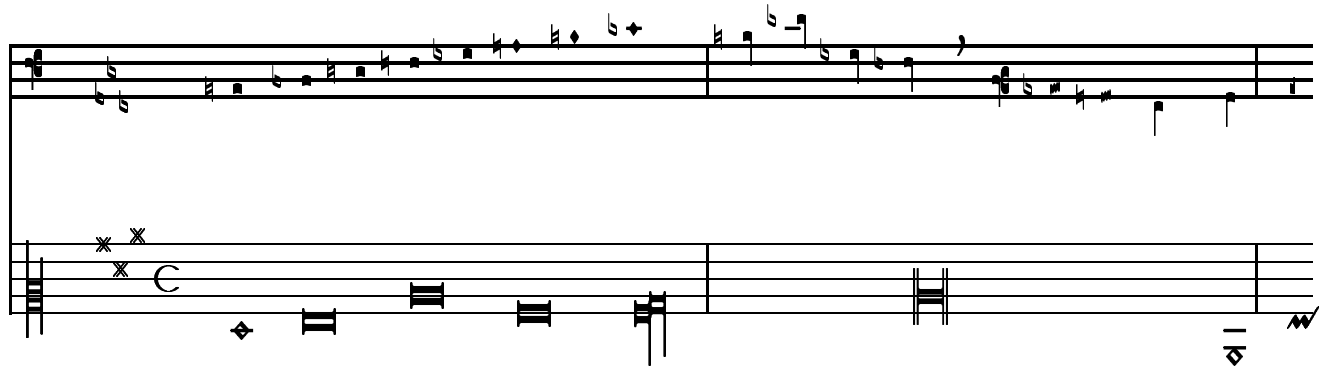


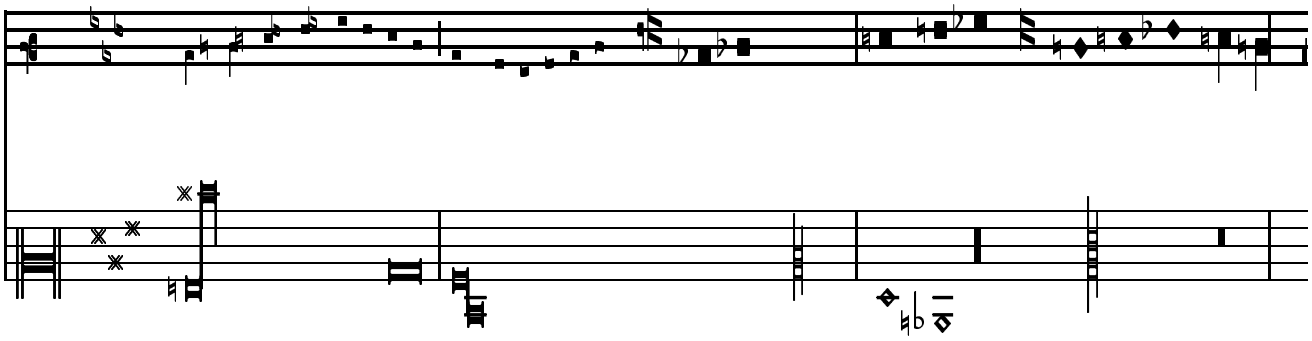
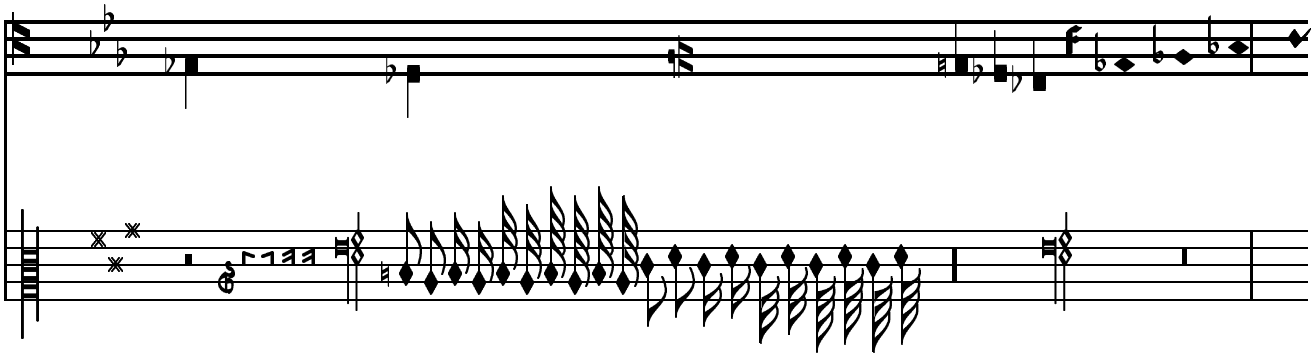
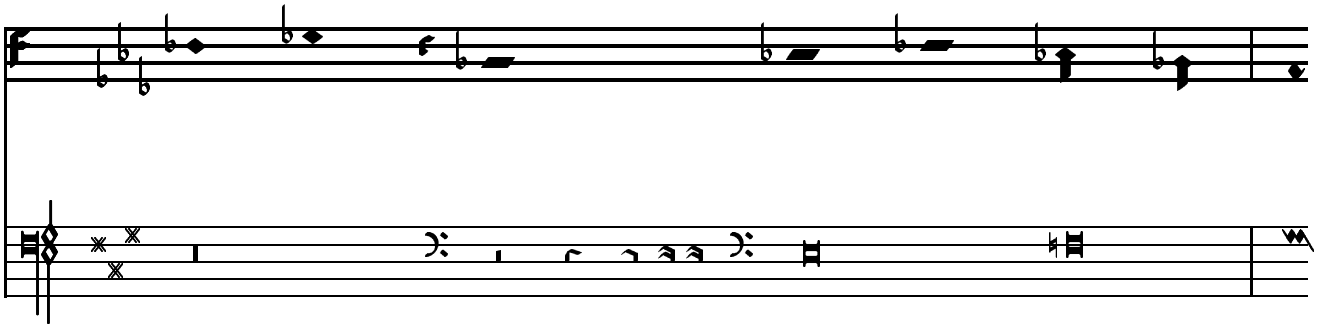


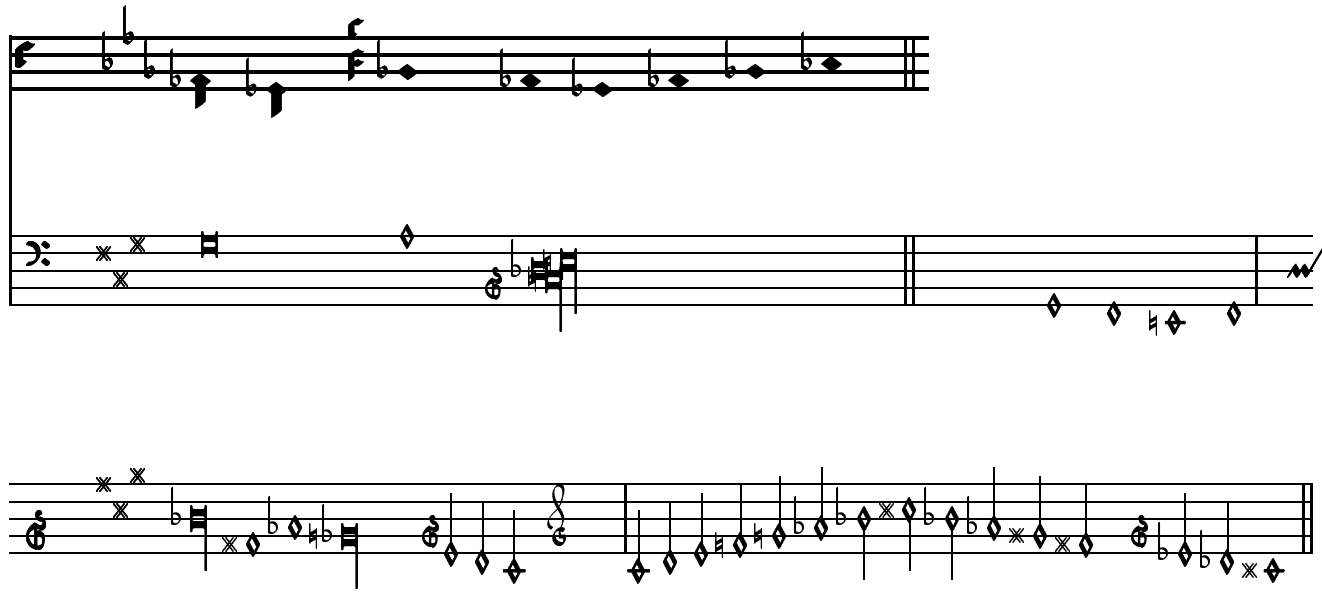
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/american-chords.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ancient-font.ly’:

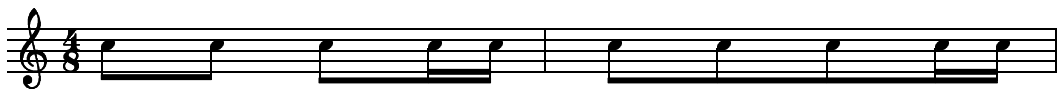


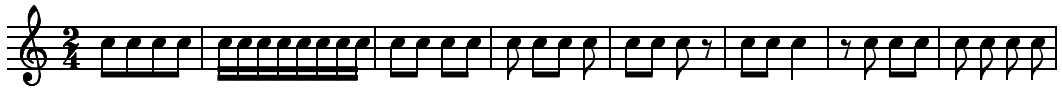


Should use old style. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ancient-time.ly’:

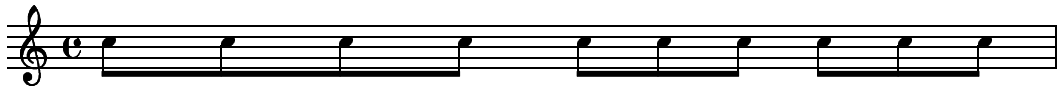
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/auto-beam-4-8.ly’:



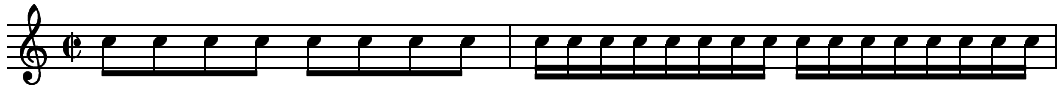
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/auto-beam-override.ly’:

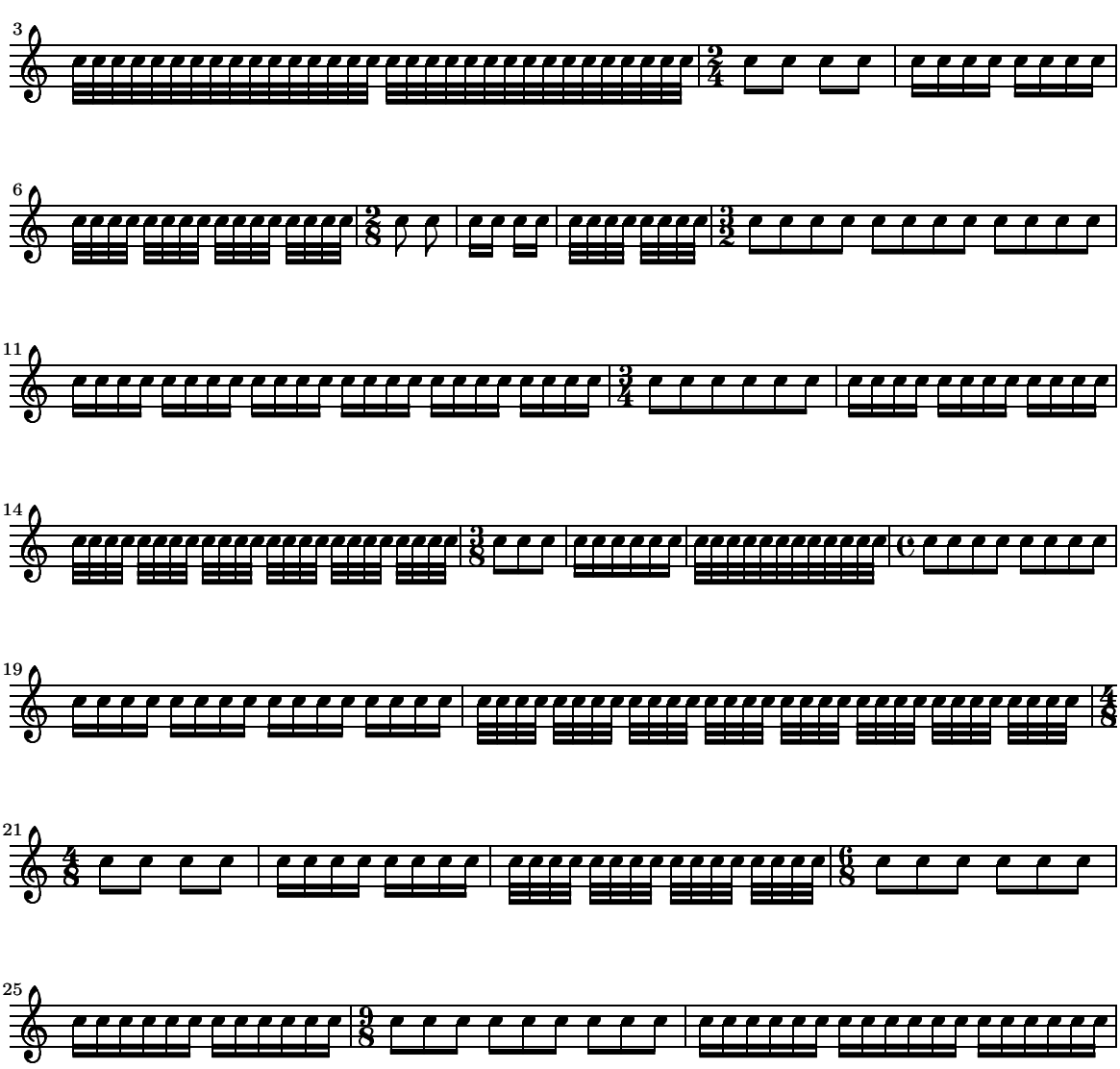


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/auto-beam-triplet.ly’:

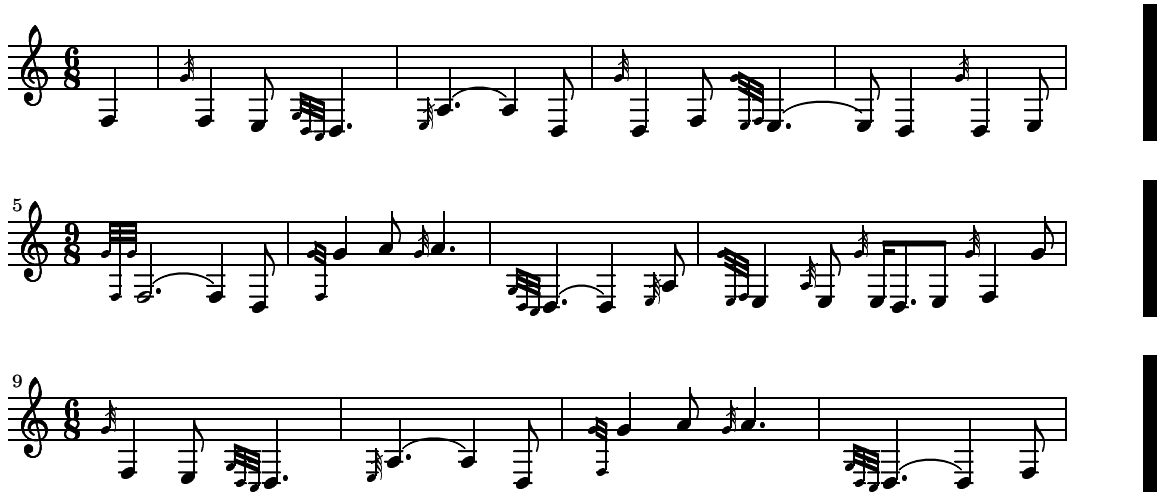


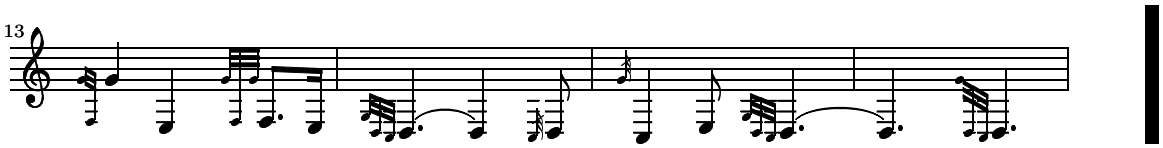
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/auto-beam.ly’:



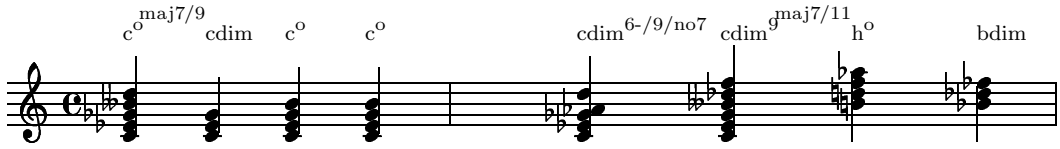


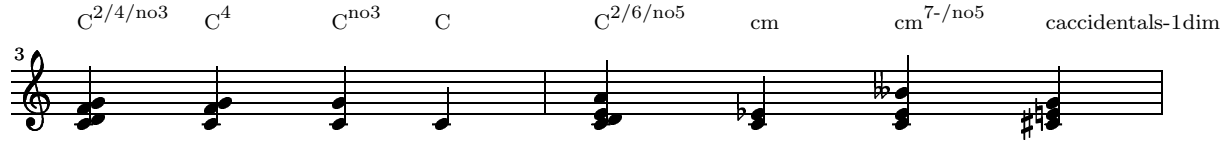
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/bagpipe.ly’:



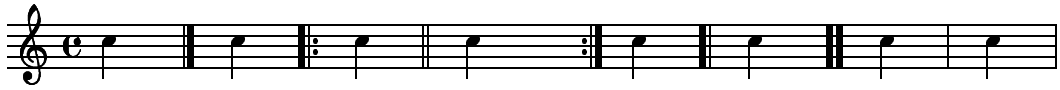


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/banter-chords.ly’:

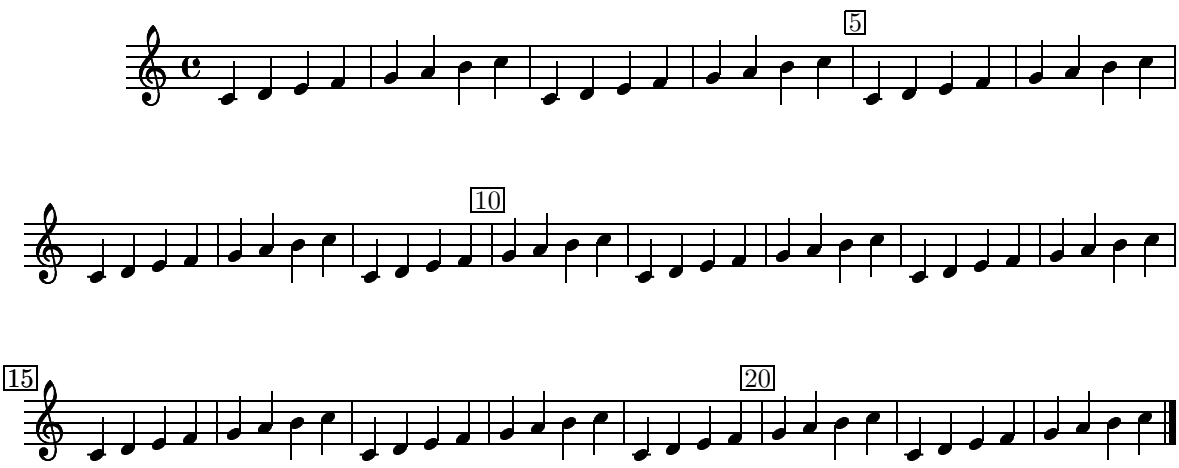




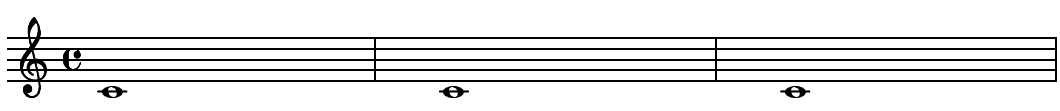
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/bar-lines.ly’:

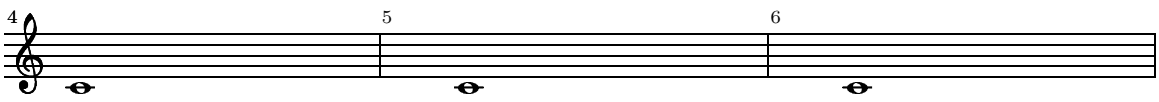


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/bar-number-every-fifth.ly’:

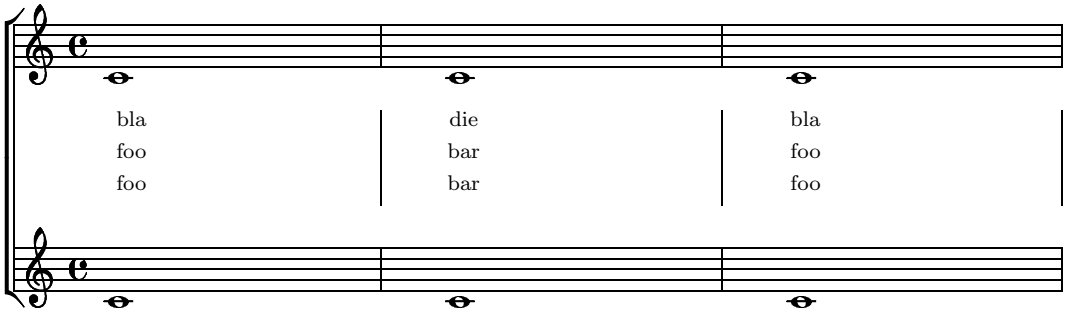


Second line has bar-numbers on start of every measure. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/

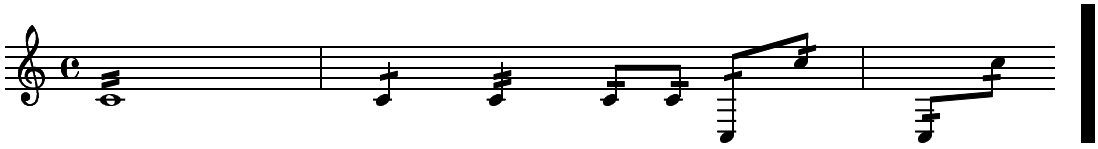




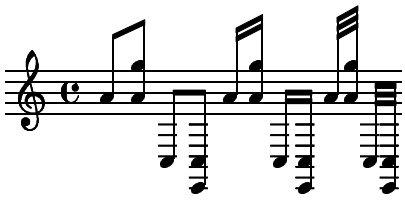
You can move around Bar_engraver and Span_bar_engraver if you want bar lines on lyrics.
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/barline-lyric-only.ly’:



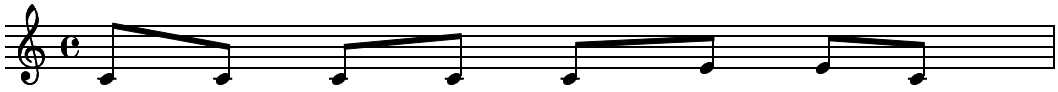
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-abbrev.ly’:



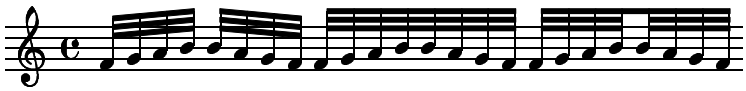
Hmm. what’s this supposed to test? ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-chord



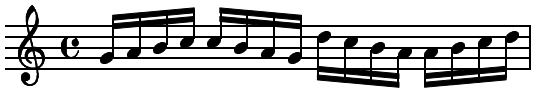
Controlling beam positions. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-control.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-count.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-damp.ly’:



There are several ways to calculate the direction of a beam

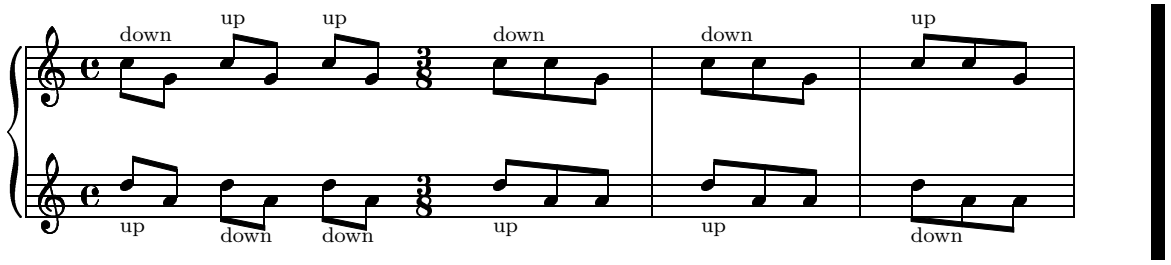
‘`majority`’
: number count of up or down notes

‘`mean`’ mean centre distance of all notes

‘`median`’ mean centre distance weighted per note

These beam direction functions are defined in ‘`scm/beam.scm`’. If your favourite algorithm isn’t one of these, you can hook up your own.

Of course, this depends on the neutral-direction for the middle line, down by default. We set that to 1 (up) in the lower staff. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-dir-funct`



There are several ways to calculate the direction of a beam.

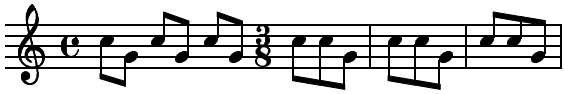
`majority` number count of up or down notes

`mean` mean center distance of all notes

`median` mean centre distance weighted per note

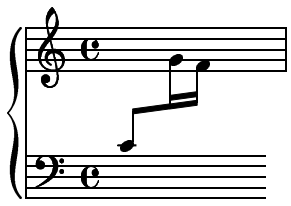
You can spot the differences of these settings from these simple examples:

These beam direction functions are defined in ‘`scm/beam.scm`’. If your favourite algorithm isn’t one of these, you can hook up your own. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/b`

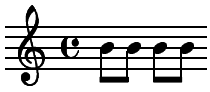


‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-dir.ly`’:

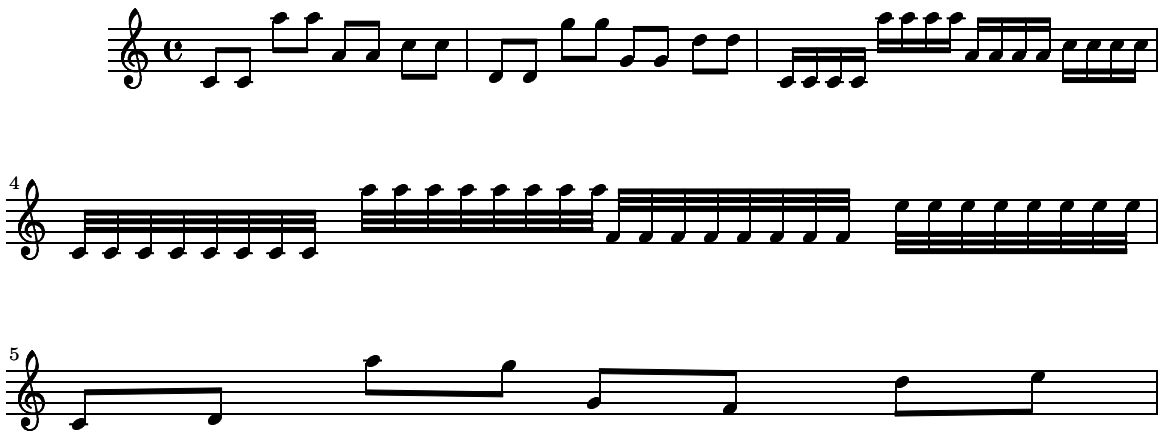
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-isknee.ly`’:



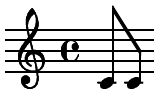
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-neutral-direction.ly’:



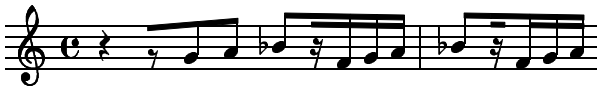
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-pos.ly’:



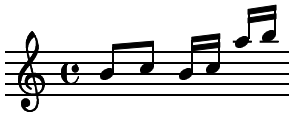
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-position.ly’:



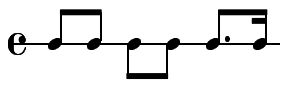
Beams over rests. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-rest.ly’:



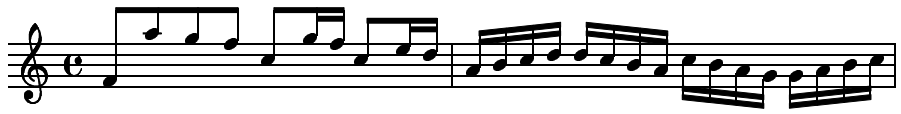
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-second.ly’:



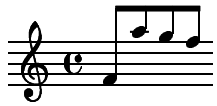
Beam-stem attachment test. Helper file for ps/dvips problems. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/



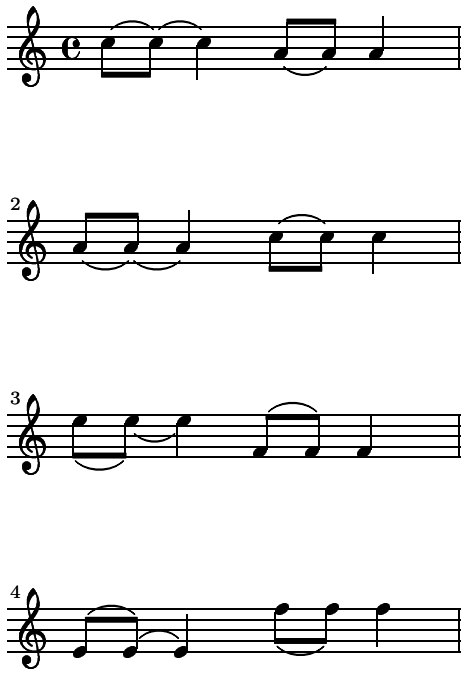
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-suspect.ly’:



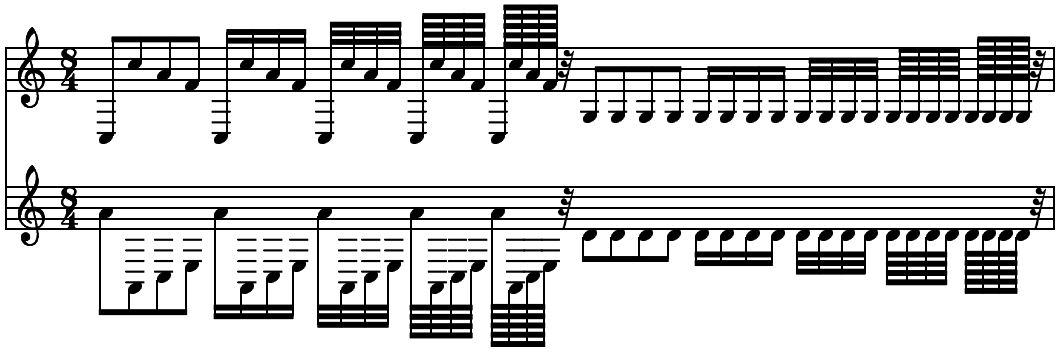
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beam-trend.ly’:

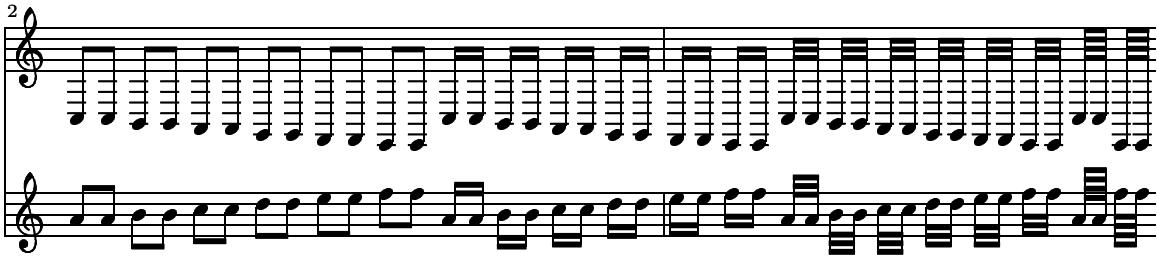


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beamed-slur-endings.ly’:

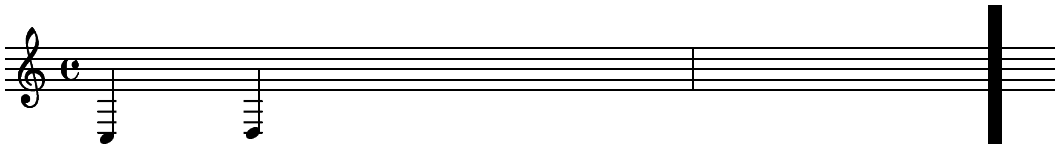


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/beams.ly’:



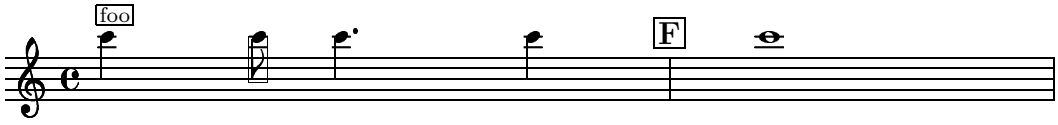


print lesson sheets that contain blank lines and just portions of blank lines. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/blank-notes.ly`’:

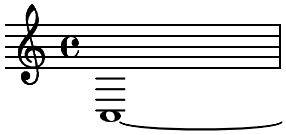


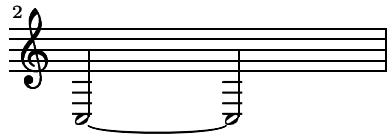
Blank music paper with clefs ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/blank.ly`’:

overriding the molecule callback can also be used to draw a box around arbitrary grobs.
TODO: circled molecules.
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/boxed-molecule.ly’:

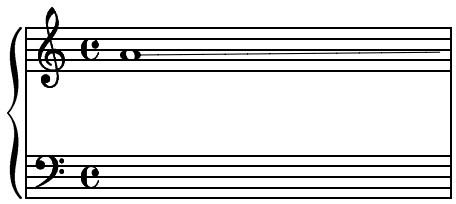


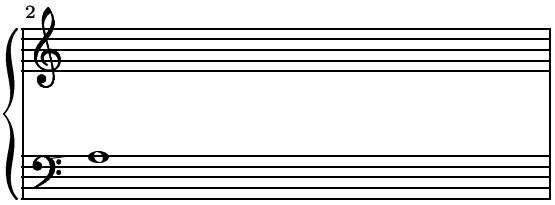
Broken spanners can be adjusted individually, but this requires complicated scheme code.
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/broken-spanner-adjustment.ly’:



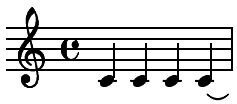


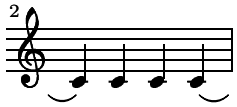
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/broken-thread-line.ly’:

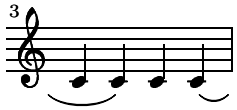


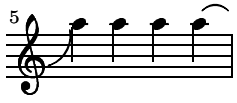


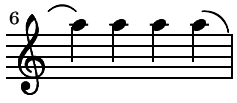
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/broken.ly’:

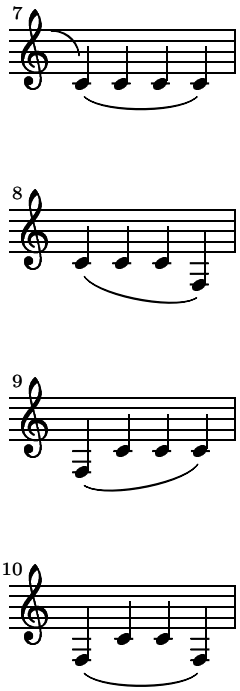






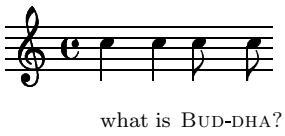




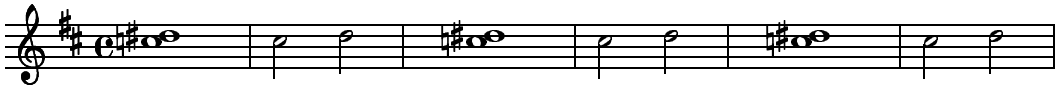


Small caps is available as font-shape caps.

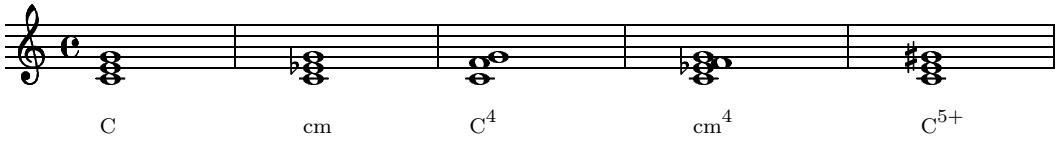
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/caps.ly’:

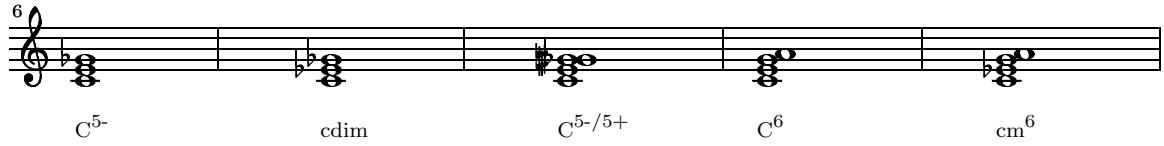


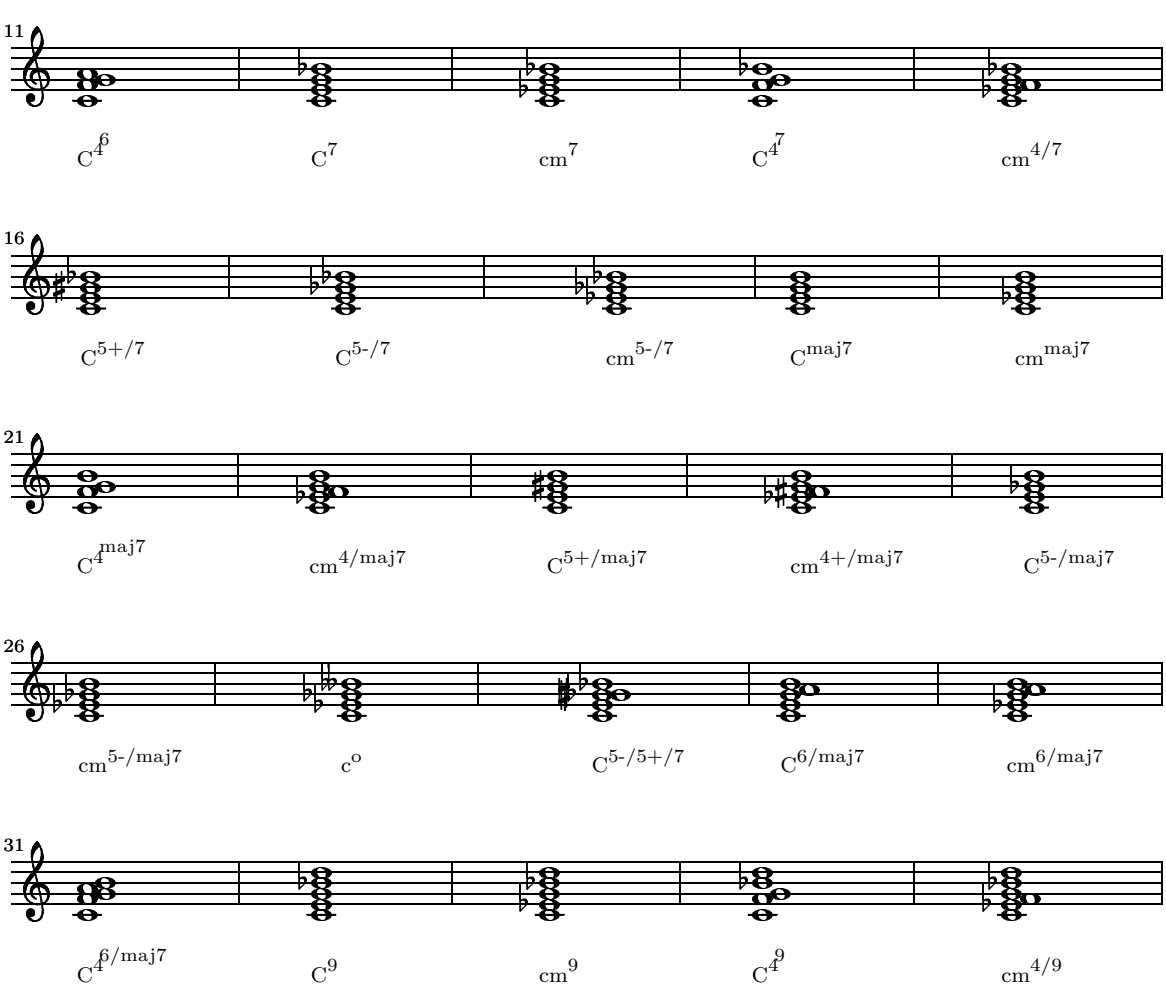
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/cautionaries.ly’:



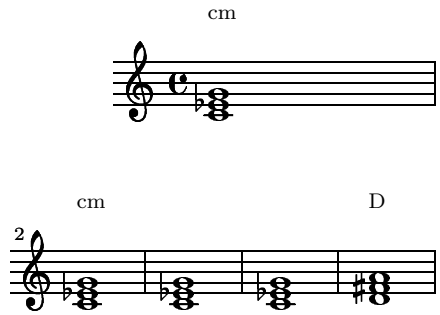
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/chord-banter.ly’:



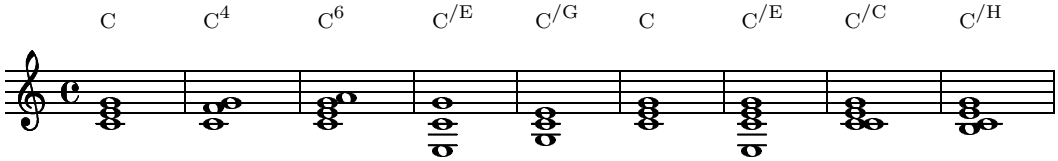




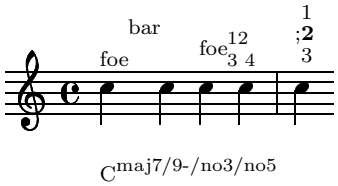
property chordChanges: only display chord names when there's a change in the chords scheme, but always display the chord name after a line break. '`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/chord-changes.ly`':



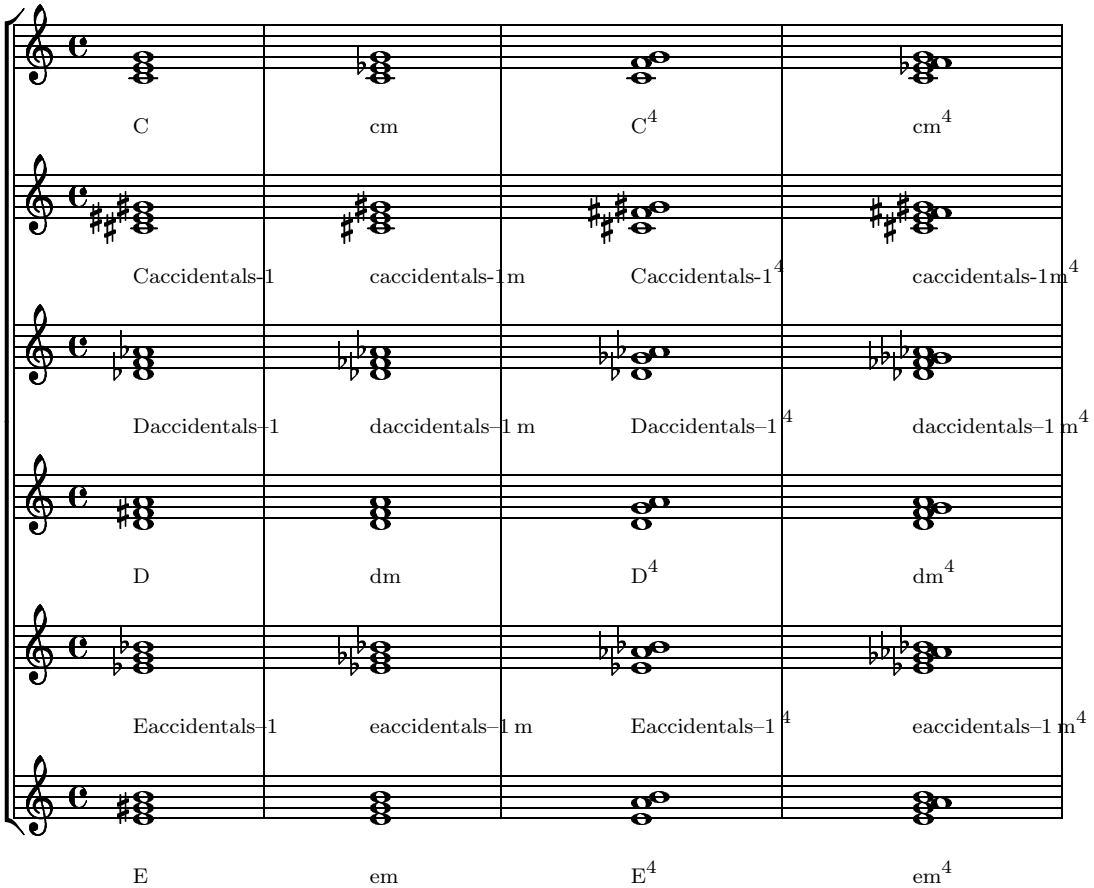
'`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/chord-inversion.ly`':

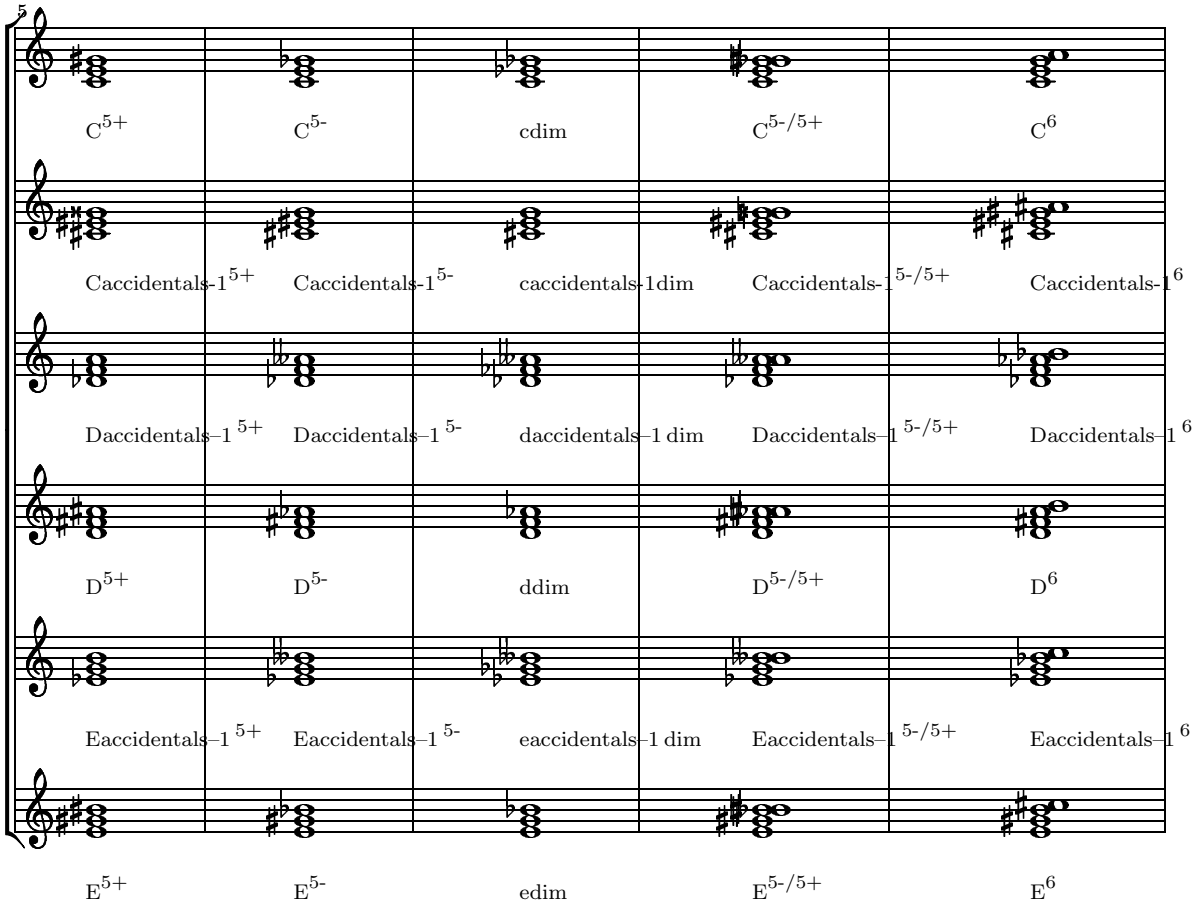


chord/markup test ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/chord-markup.ly’:

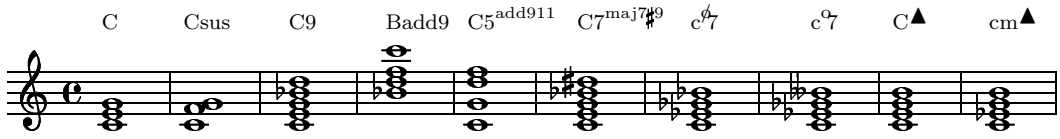


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/chord-table.ly’:

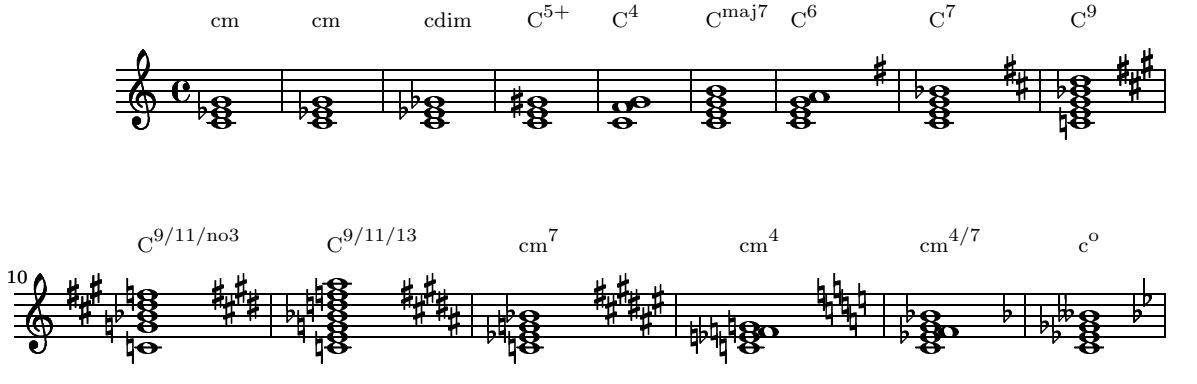


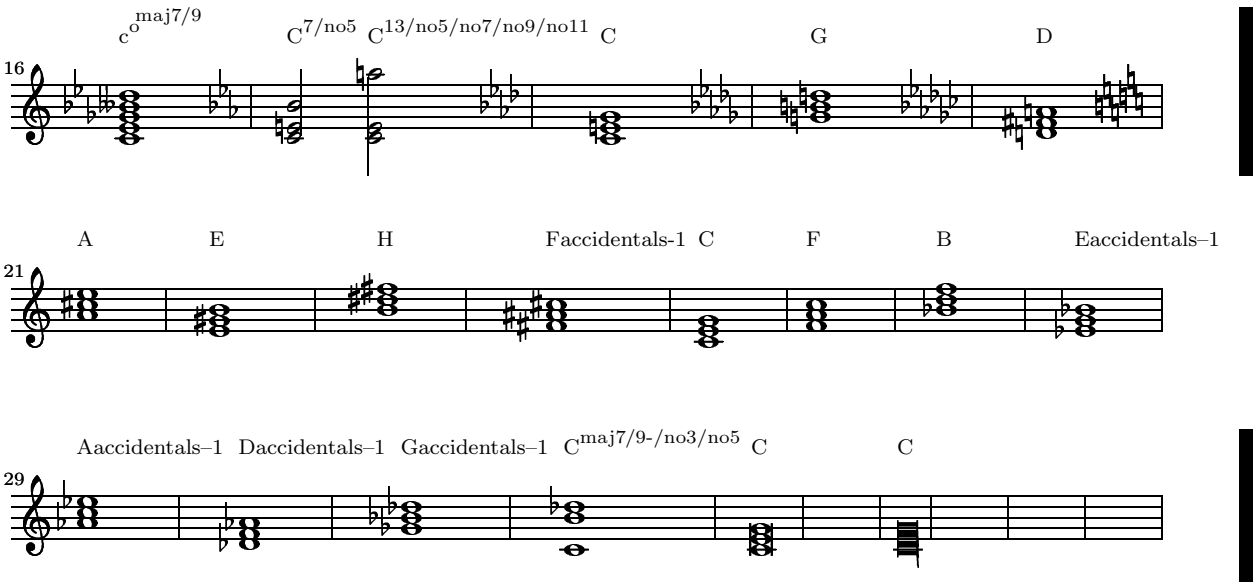


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/chord-taxonomy.ly’:

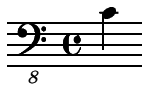


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/chords.ly’:

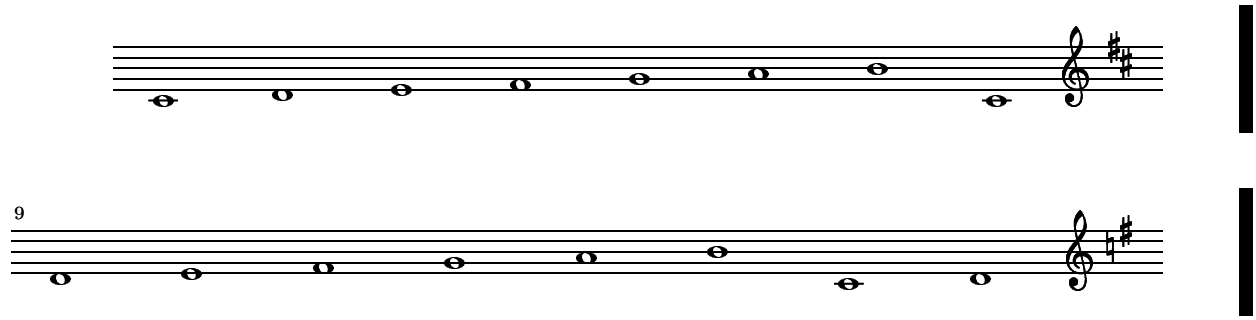




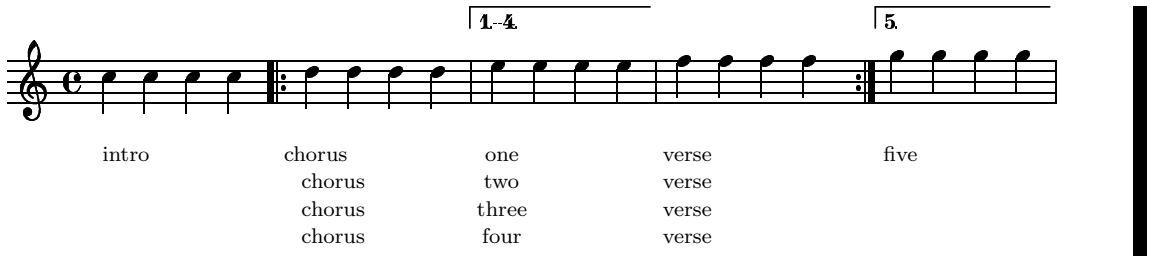
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/clef-8-syntax.ly’:



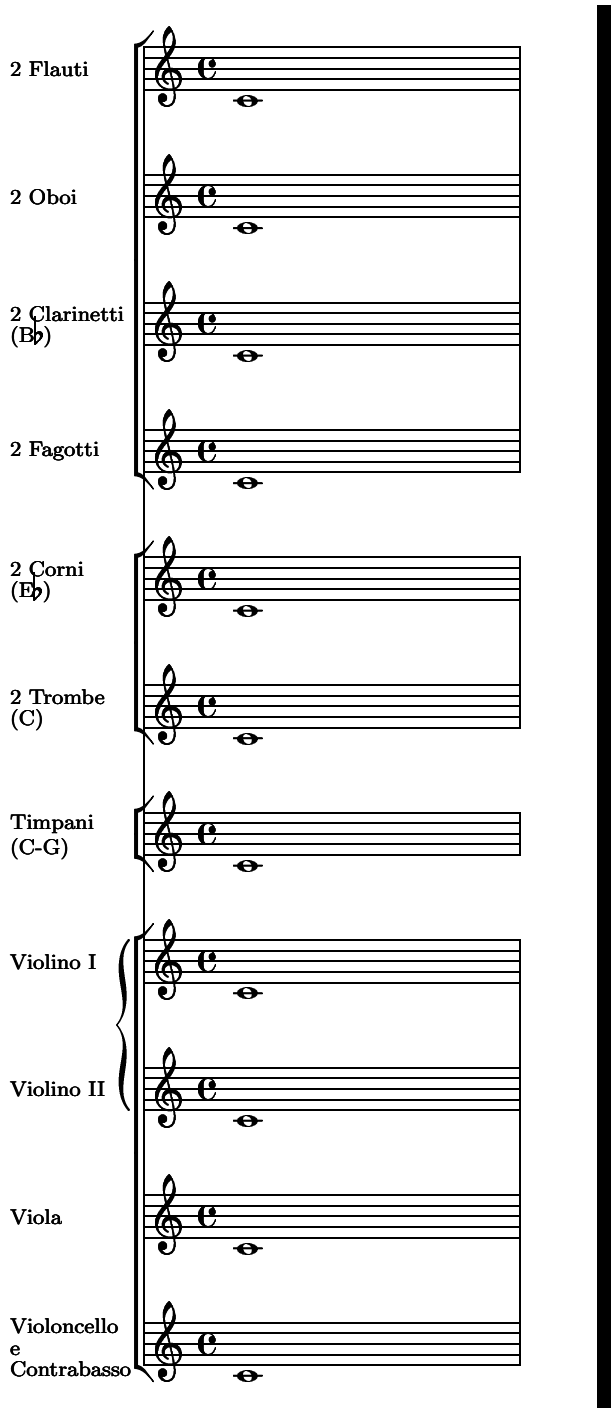
Scales, but with clef and key signature at the end of the line. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/in

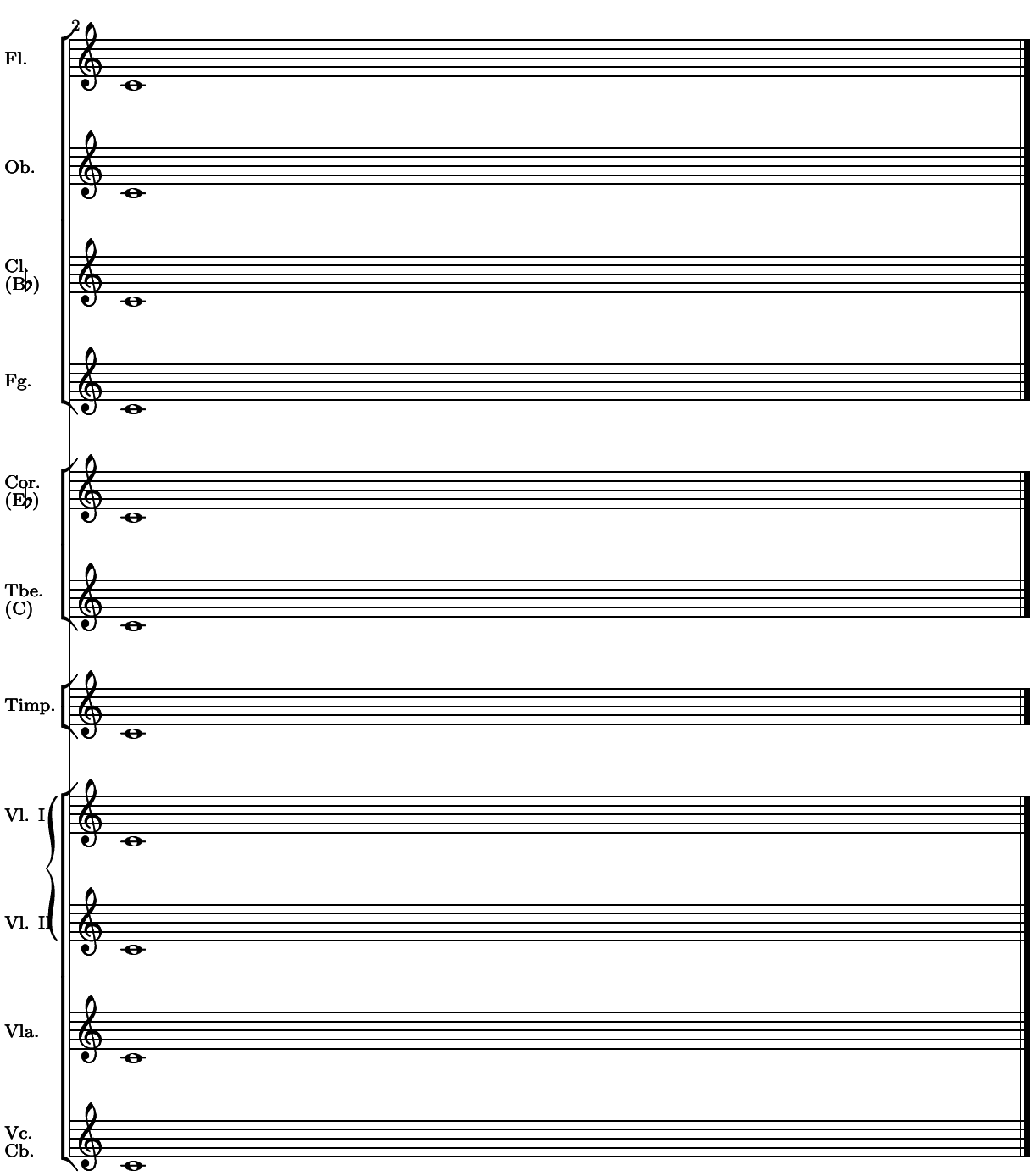


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/coda-kludge.ly’:

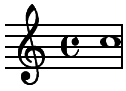


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/coriolan-margin.ly’:

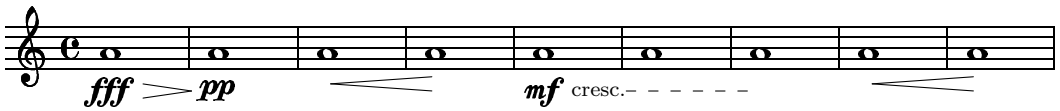

2 Flauti
2 Oboi
2 Clarinetti
(B♭)
2 Fagotti
2 Corni
(E♭)
2 Trombe
(C)
Timpani
(C-G)
Violino I
Violino II
Viola
Violoncello
e
Contrabasso



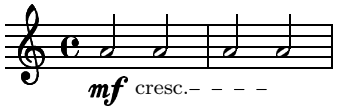
Display the number of systems, or the system number of a Grob. This can be most useful to assertain that a piece uses a specified number of lines. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/count-systems.ly`’:



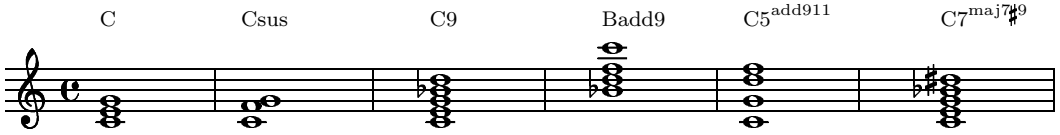
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/crescendi.ly’:



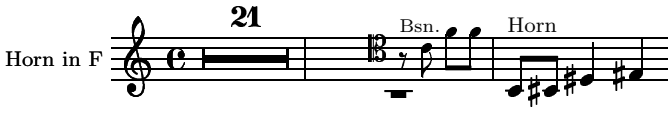
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/crescendo-text.ly’:



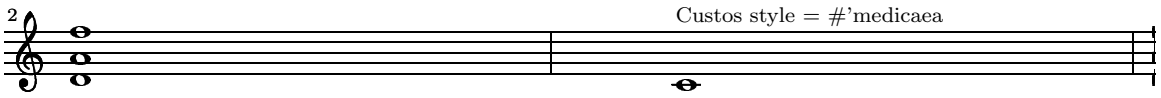
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ct-jazz.ly’:

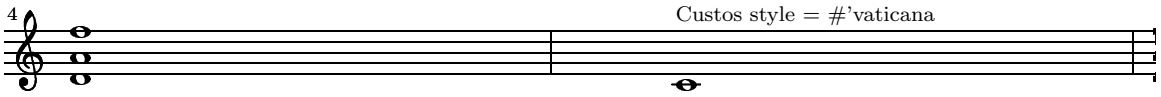


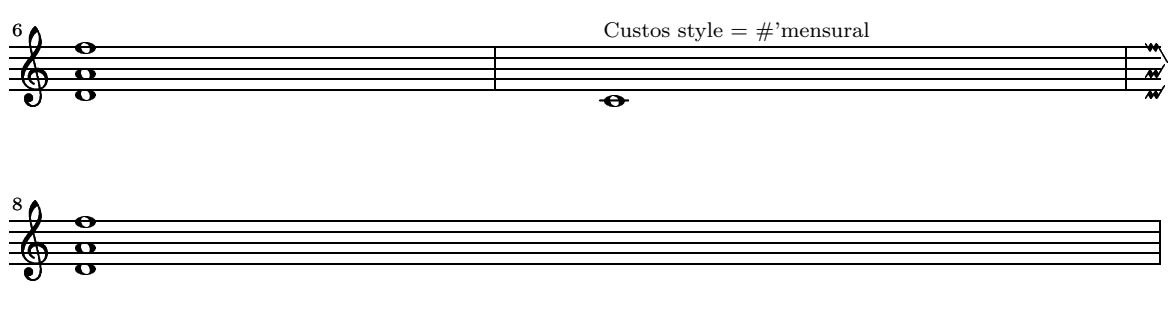
Cue notes should be set in smaller type. Cue clefs are usually not restored explicitly. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/cue-notes.ly’:



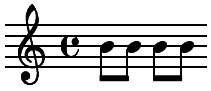
custodes in various styles. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/custos.ly’:



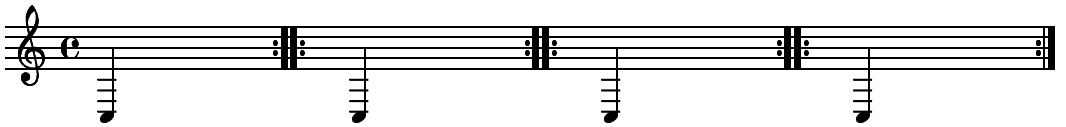




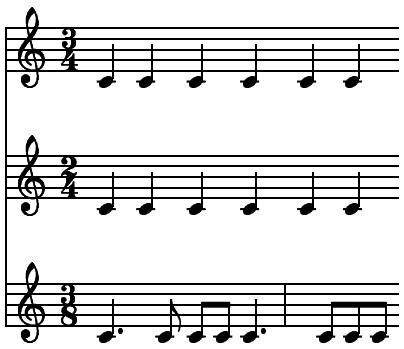
The direction of a perfectly centred beams can be controlled through `Voice.Beam`'s grob-property directly`neutral-direction` '`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/default-neut`



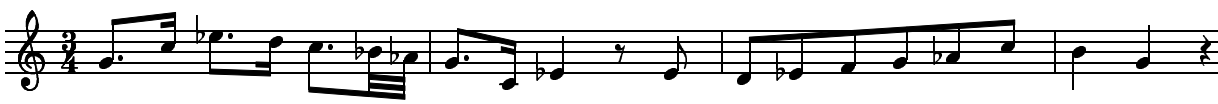
By setting barAlways and defaultBarType, you can automatically insert barlines everywhere. '`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/defaultbars.ly`':



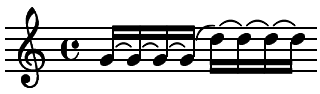
'`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/different-time-signatures.ly`':



'`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/diverse-spacing.ly`':

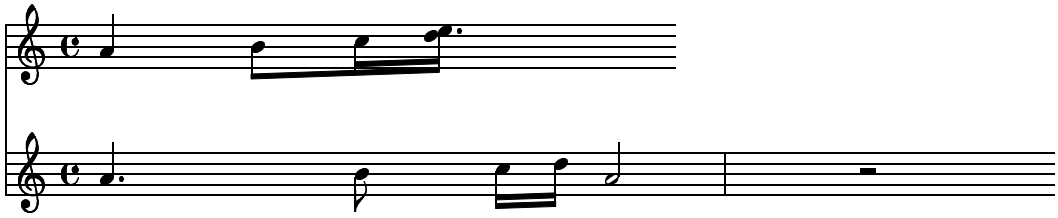


Slurs can be forced to always attach to note heads. '`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test`



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/drarn.ly’:

‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/duration-check.ly’:

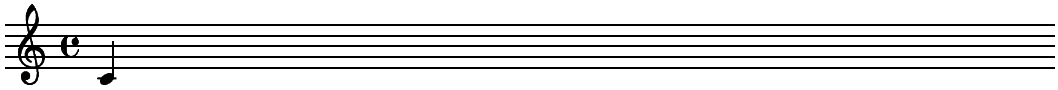


By inserting the TEX command \embeddedps, you can insert postscript directly into the output.

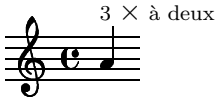
TODO: make molecule-callback to do this.

‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/embedded-postscript.ly’:

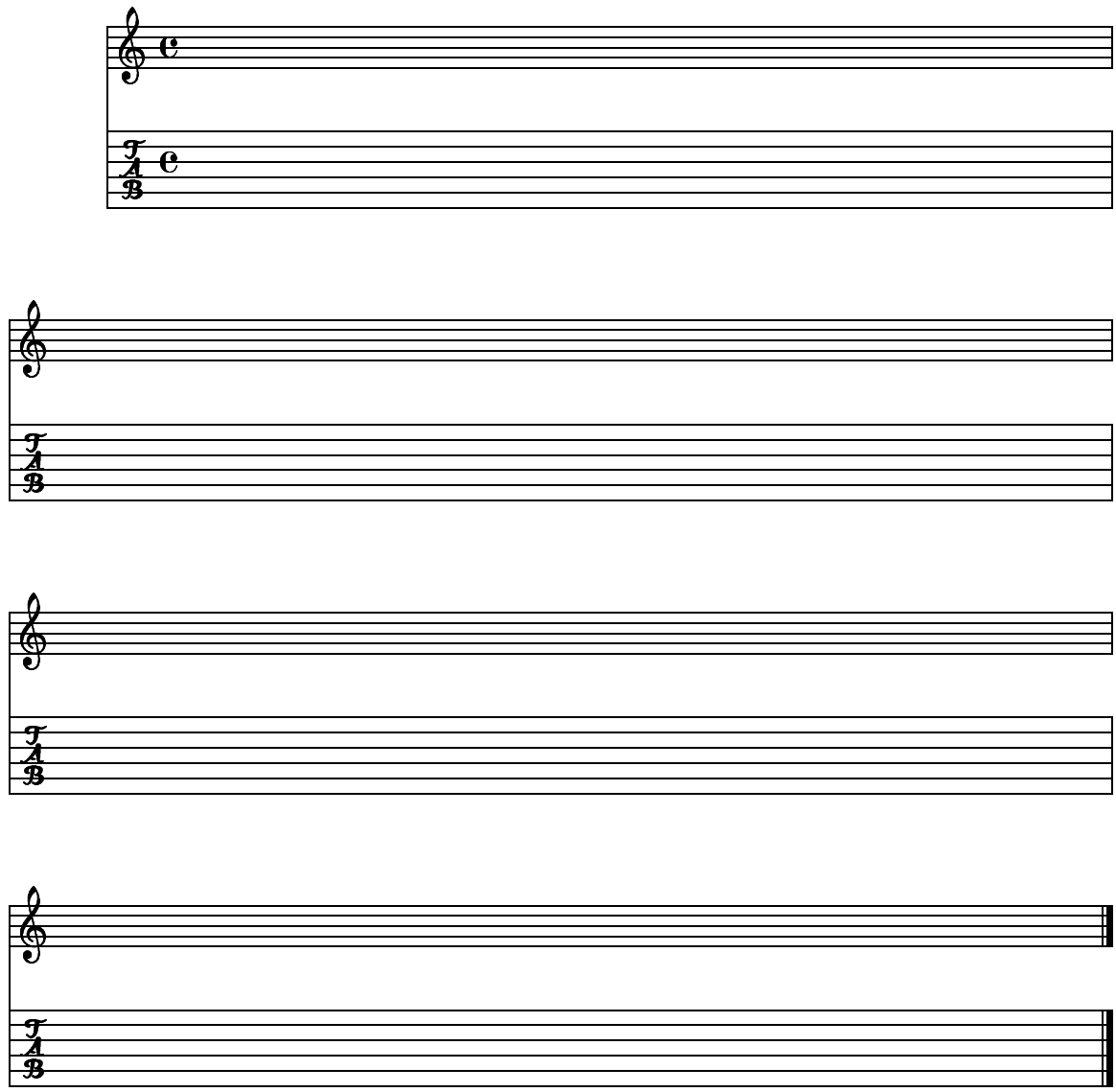
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/embedded-scm.ly’:



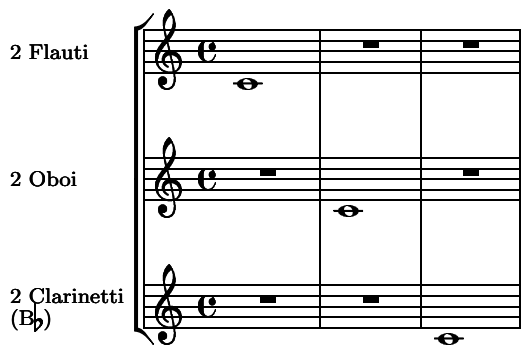
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/embedded-tex.ly’:

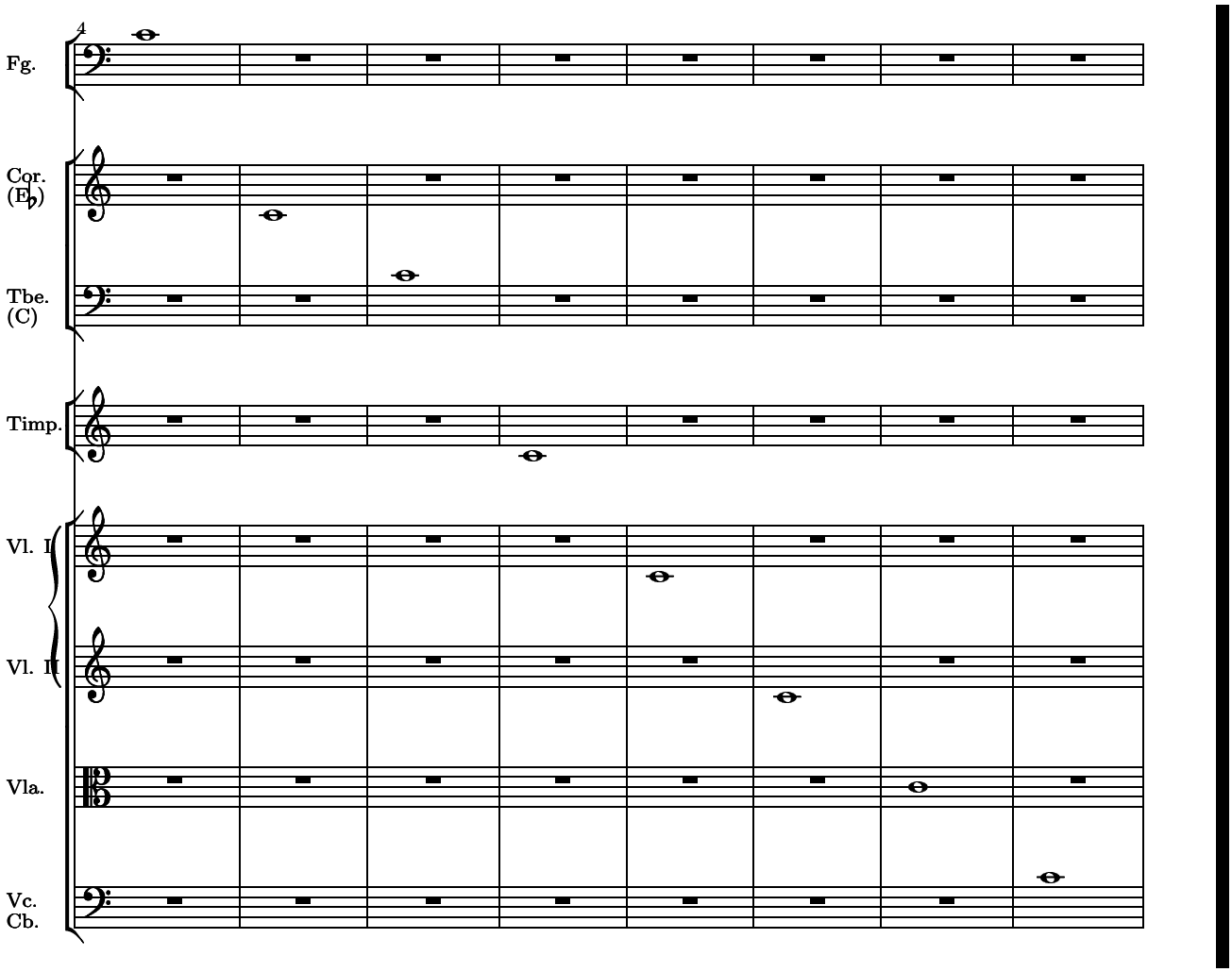


Blank music paper, another example: empty staffs and a tablature staff. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/empty-staffs.ly’:

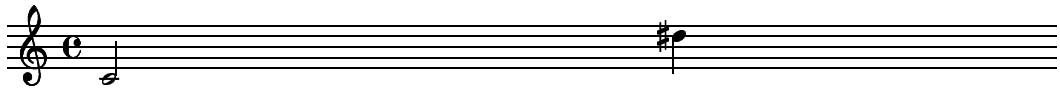


The full orchestra plays a notes, where groups stop one after another. USe this to tune equalizer settings. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/equaliser.ly`’:

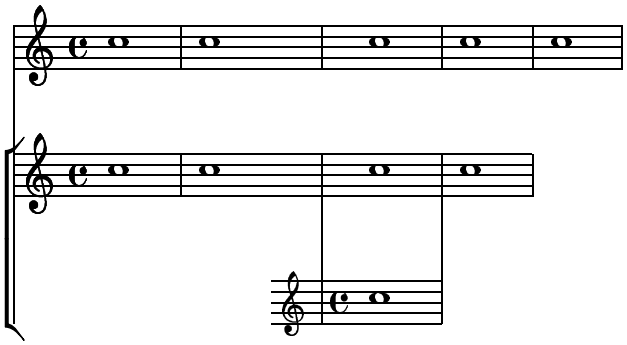




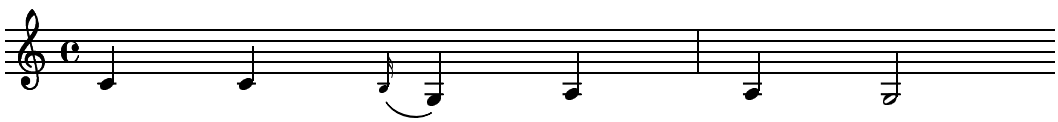
Explicit pitches and durations. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/explicit.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/extra-staff.ly’:



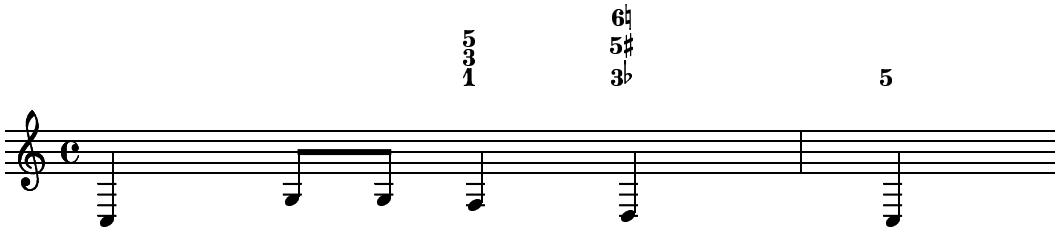
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/fake-grace.ly’:



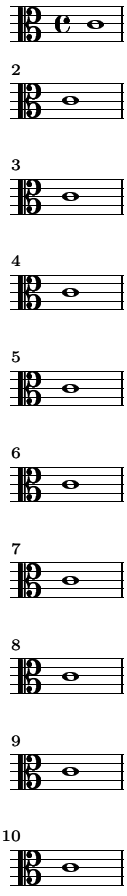
Test figured bass.

Figured bass is created by the FiguredBass context which eats note-requests and rest-requests. You can enter these either using standard `< >` notation, or using the special `\figures { }` mode, which allows you to type numbers, like `<4 6+>`.

‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/figured-bass.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/fill-a4.ly’:

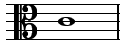


11

12

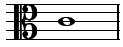
13

14



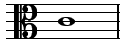
15

16

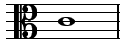


17

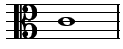
18



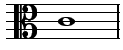
19



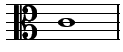
20



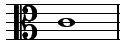
21



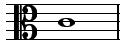
22



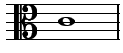
23



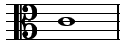
24



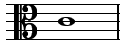
25



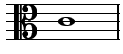
26

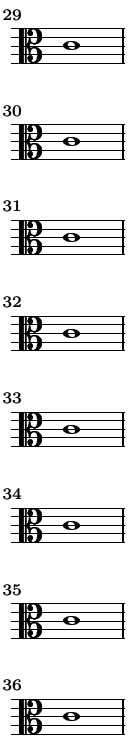


27

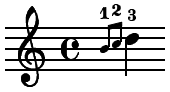


28

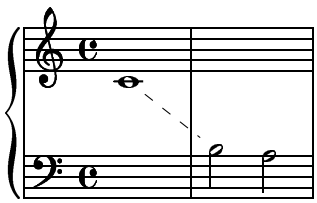




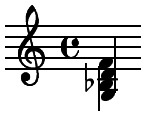
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/fingering.ly’:



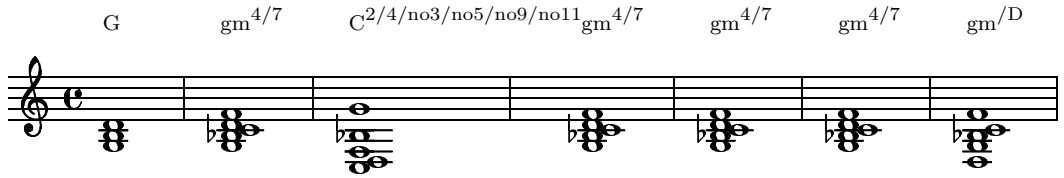
Theads can be traced automagically when they switch staves by setting property `followVoice`. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/follow-thread.ly’:



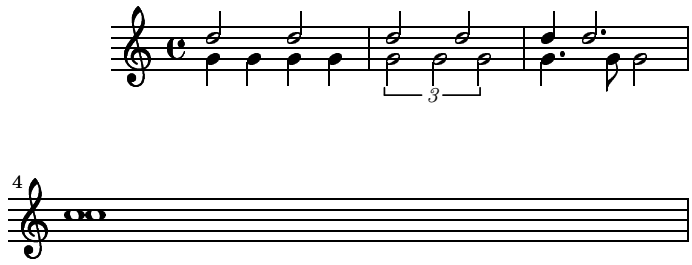
Force hshift to override collisions ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/force-hshift.



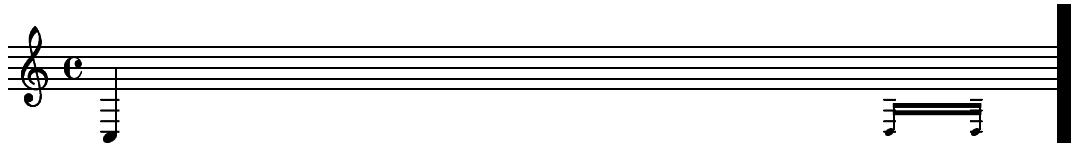
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/gmsusd.ly’:



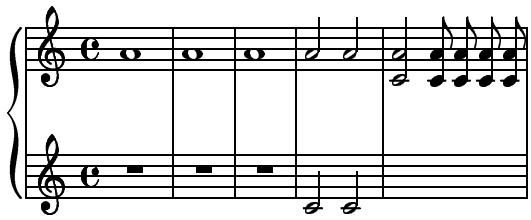
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/gourlay.ly’:



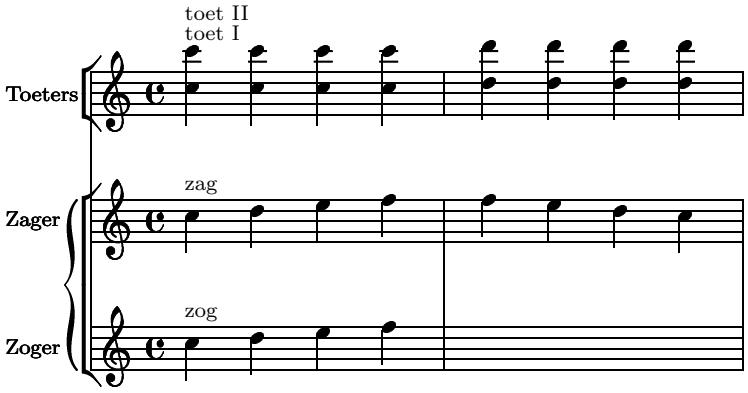
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/grace-end.ly’:

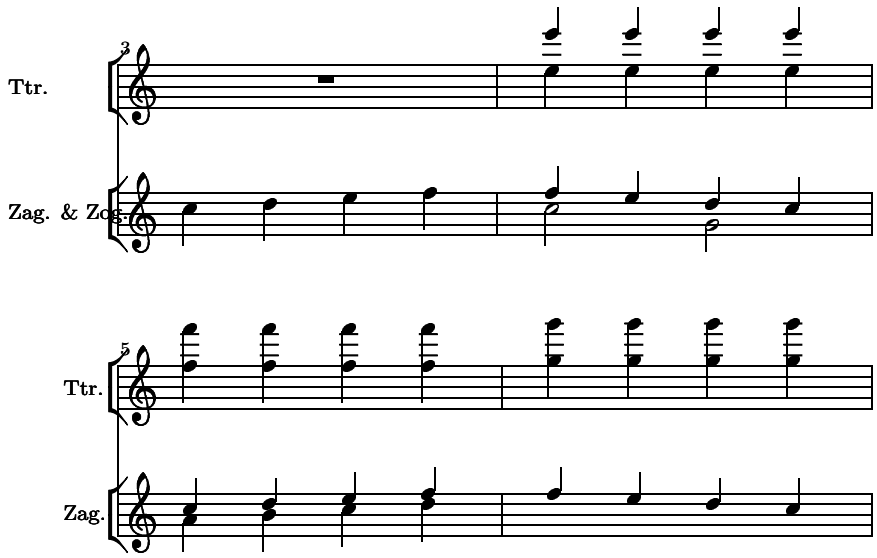


some instruments (notably: cello and double bass) are alternatively written on one and two staves. This is the way to do it: ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/hara-kiri-swit

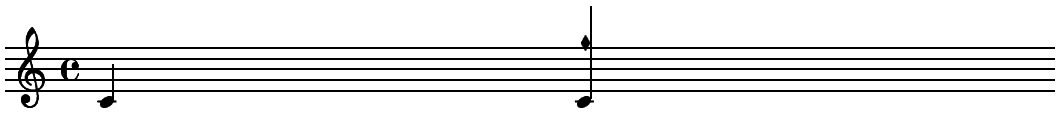


Hara kiri staves disappear when empty ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/hara-kir





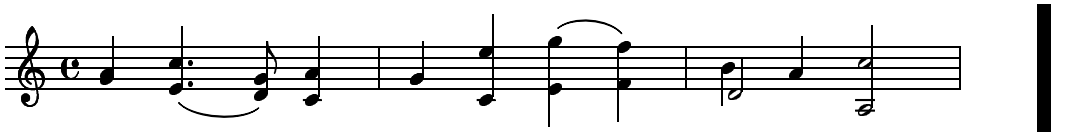
Harmonic notes: a different style on the same stem. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/tes



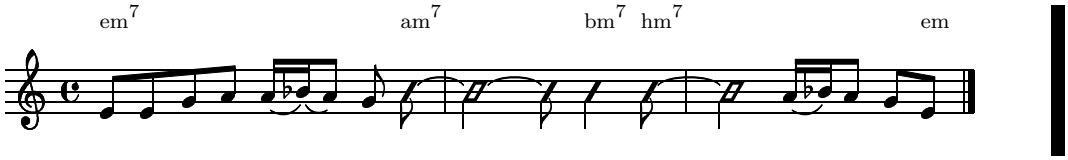
High level functionality can be accomplished with GUILE. Semantics aren’t nice though. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/header-ifelse.ly’:

‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/hshift.ly’:

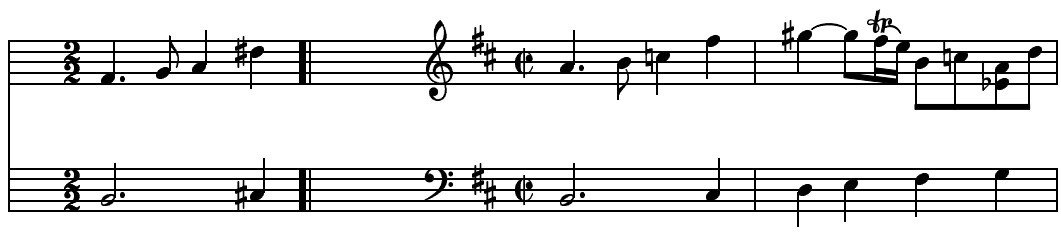
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/hymn.ly’:



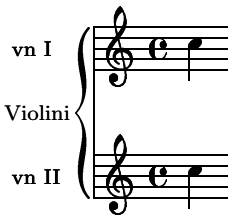
note heads for improvisation have a different shape. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/tes



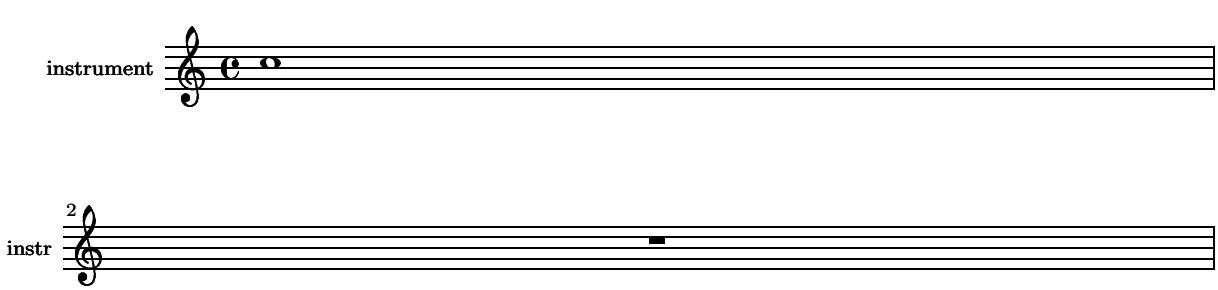
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/incipit.ly’:



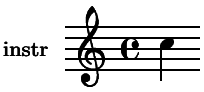
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/instrument-name-grandstaff.ly’:



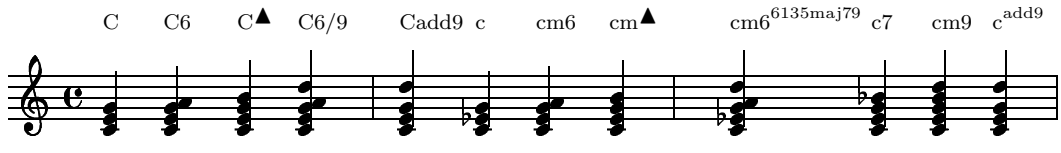
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/instrument-name-mmrest.ly’:

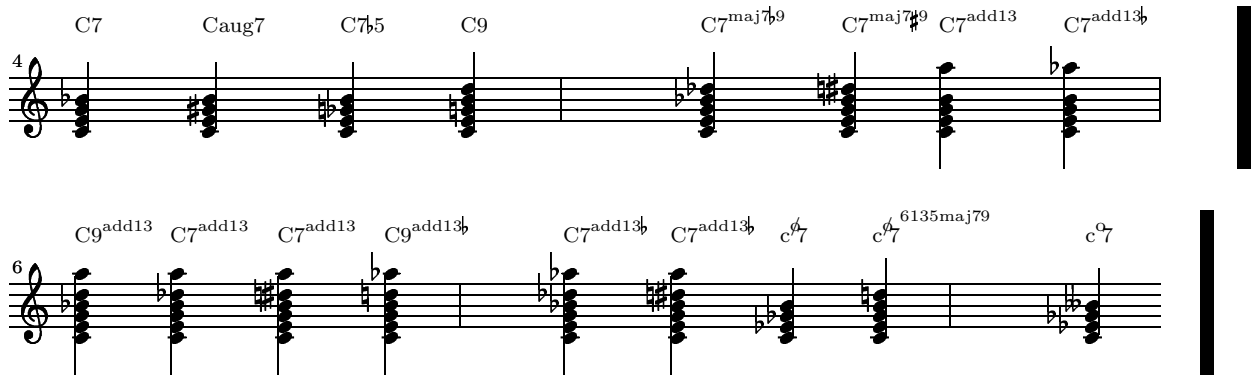


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/instrument-name.ly’:

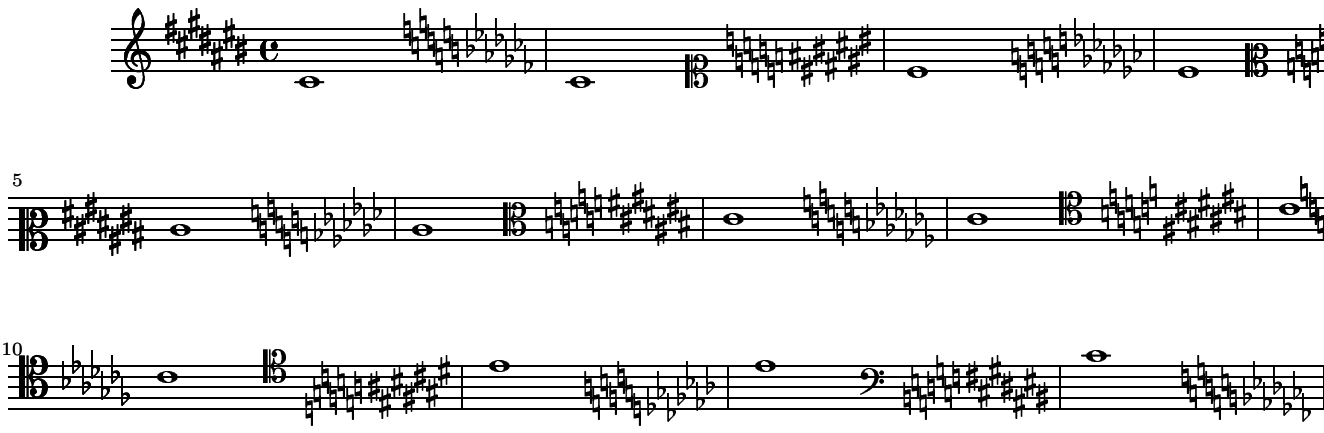


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/jazz-chords.ly’:

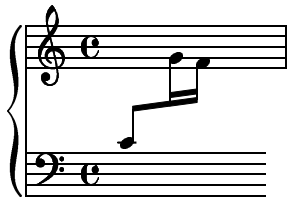




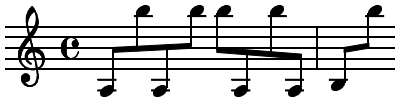
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/key-clefs.ly’:



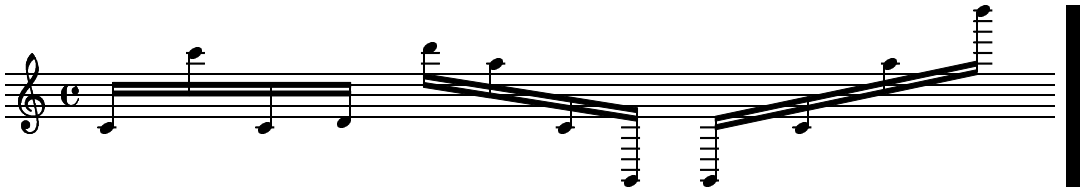
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/knee-mult.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/knee-sym.ly’:

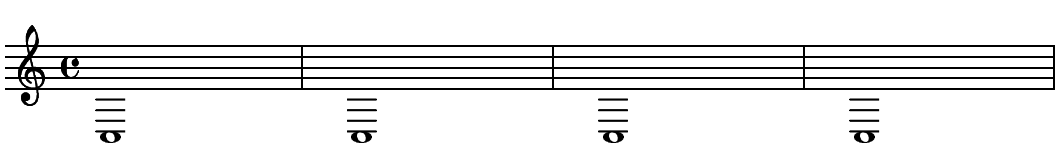


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/knee.ly’:

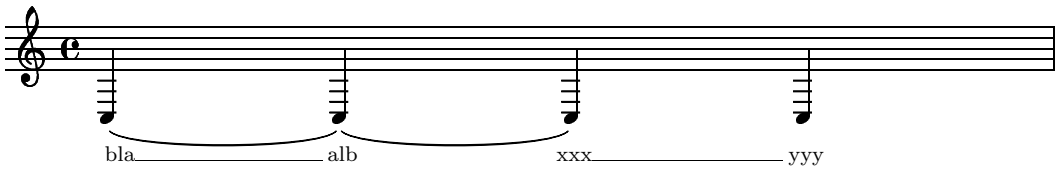


Draw abritrary lines. This brings LilyPond on par with Mup. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/i

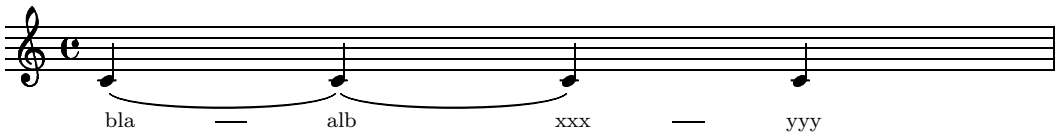
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ly2dvi-testpage.ly’:



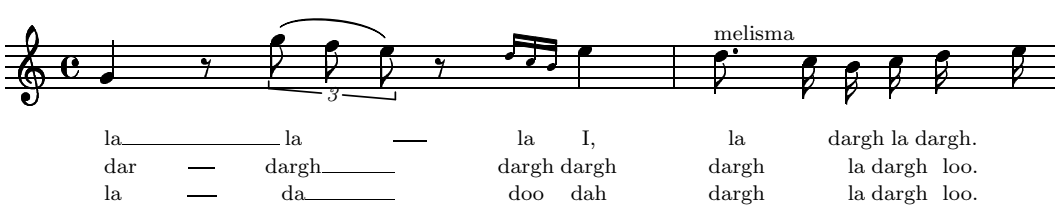
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/lyric-extender.ly’:



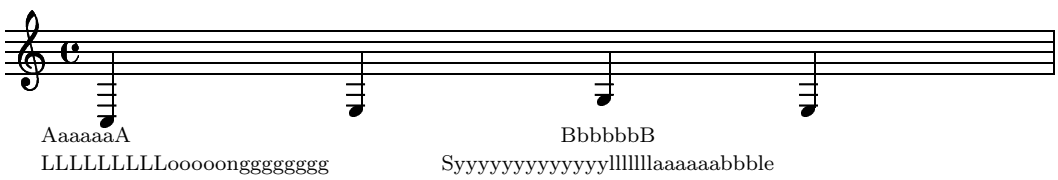
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/lyric-hyphen.ly’:

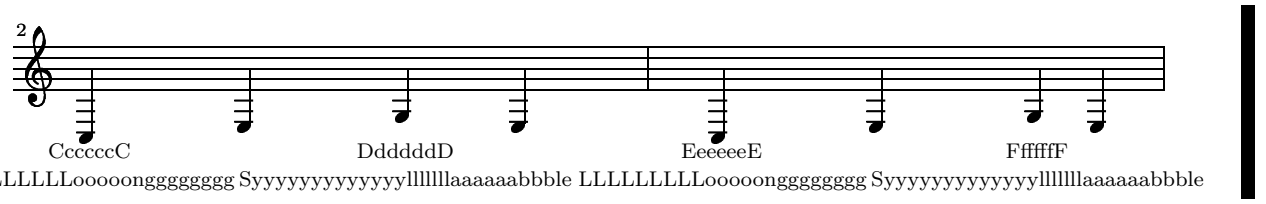


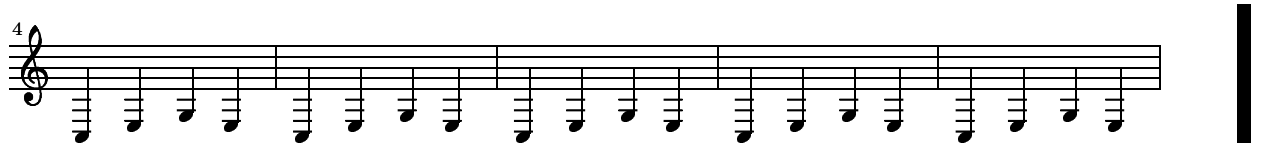
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/lyric-phrasing.ly’:



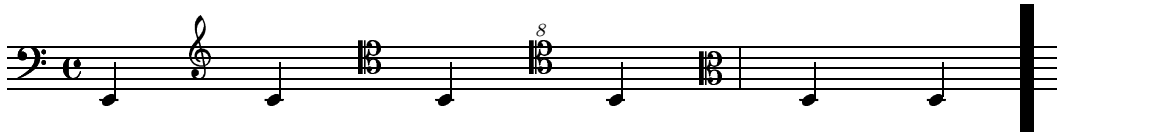
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/lyrics.ly’:



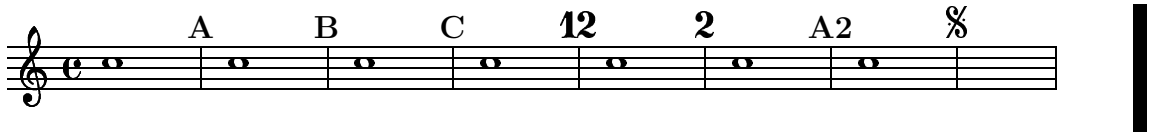




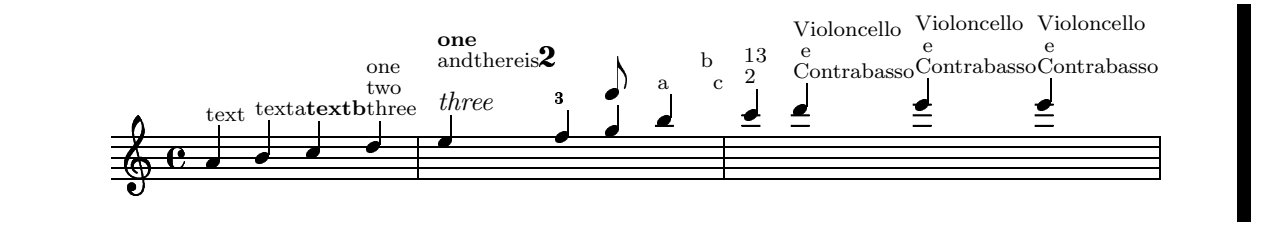
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/manual-clef.ly’:

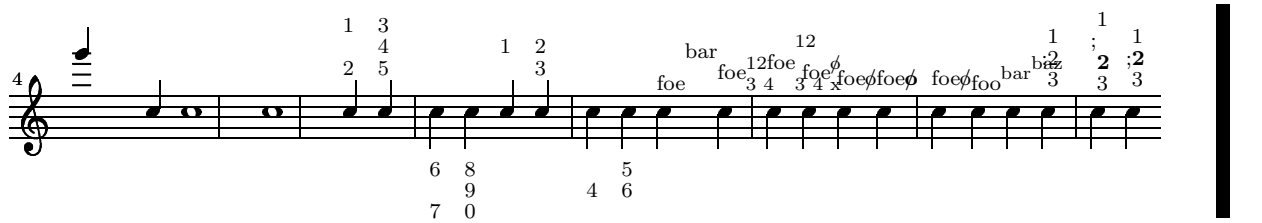


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/mark.ly’:

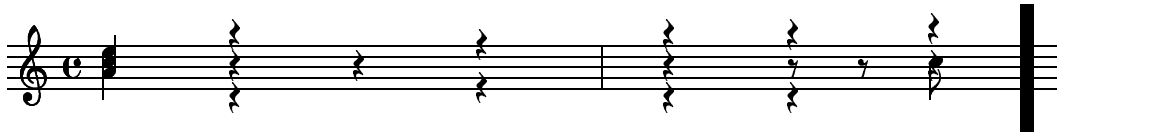


Test font selection and scm text markup ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/markup.l

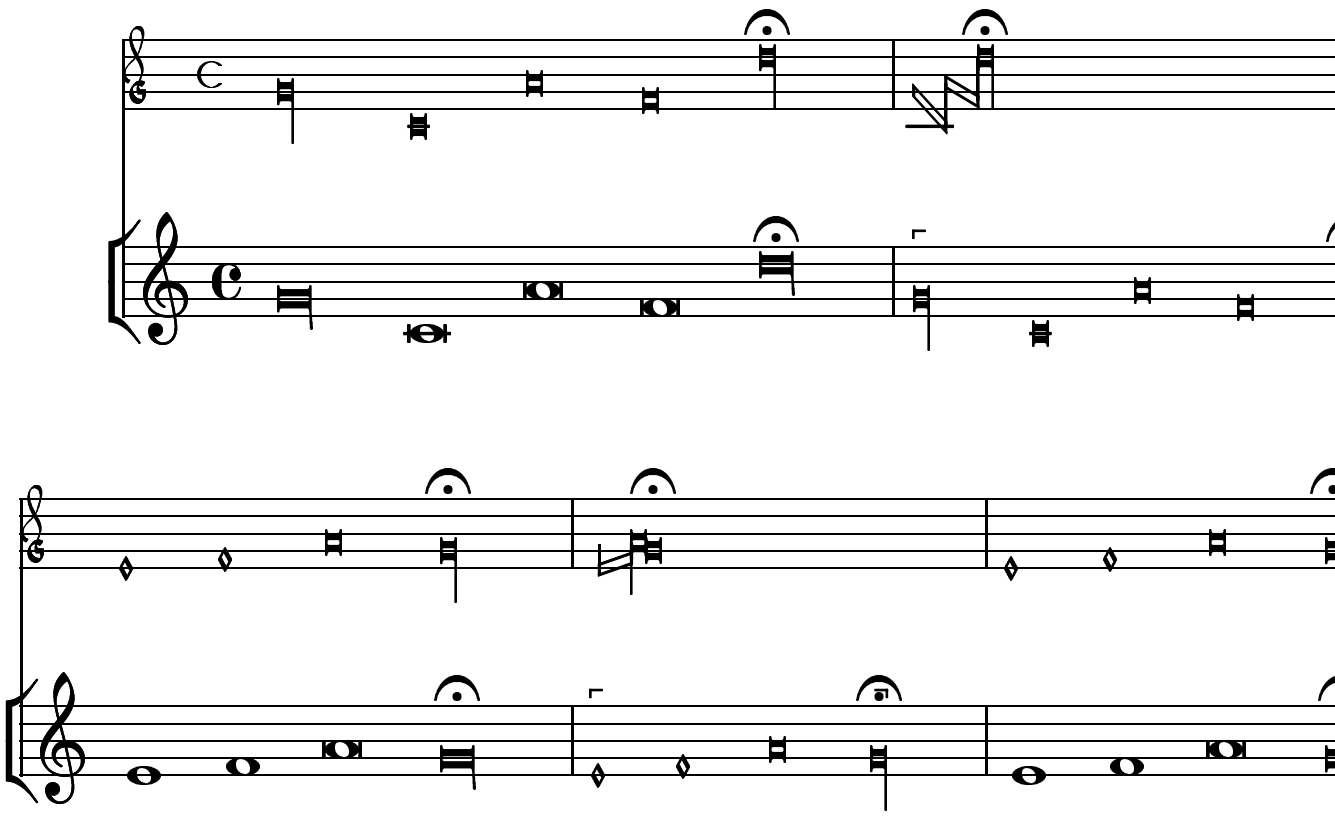




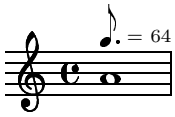
Control the number of rests in a collision with maximum-rest-count. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/maximum-rest-count.ly’:



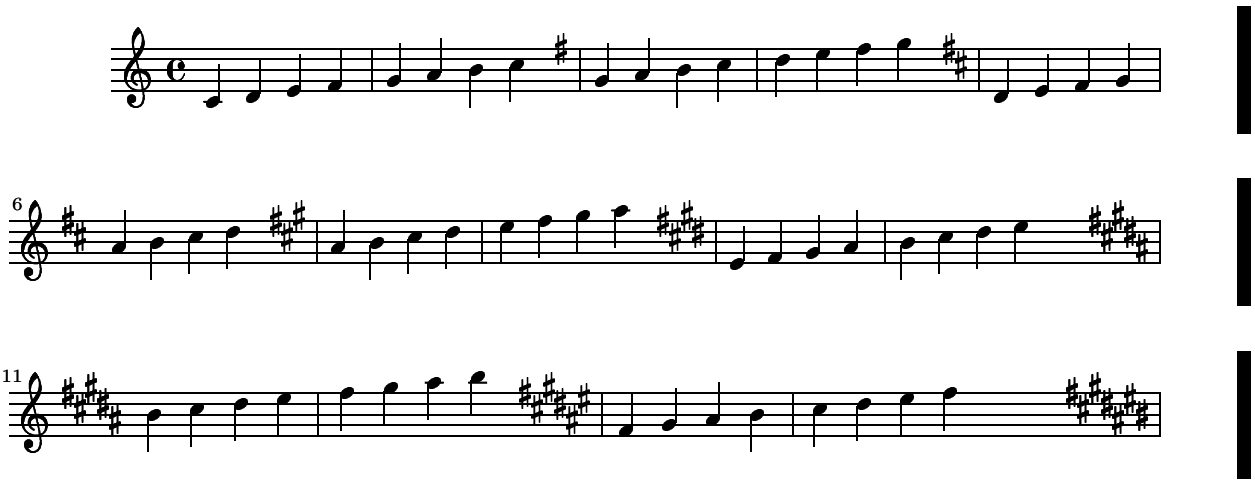
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/mensural-ligatures.ly’:

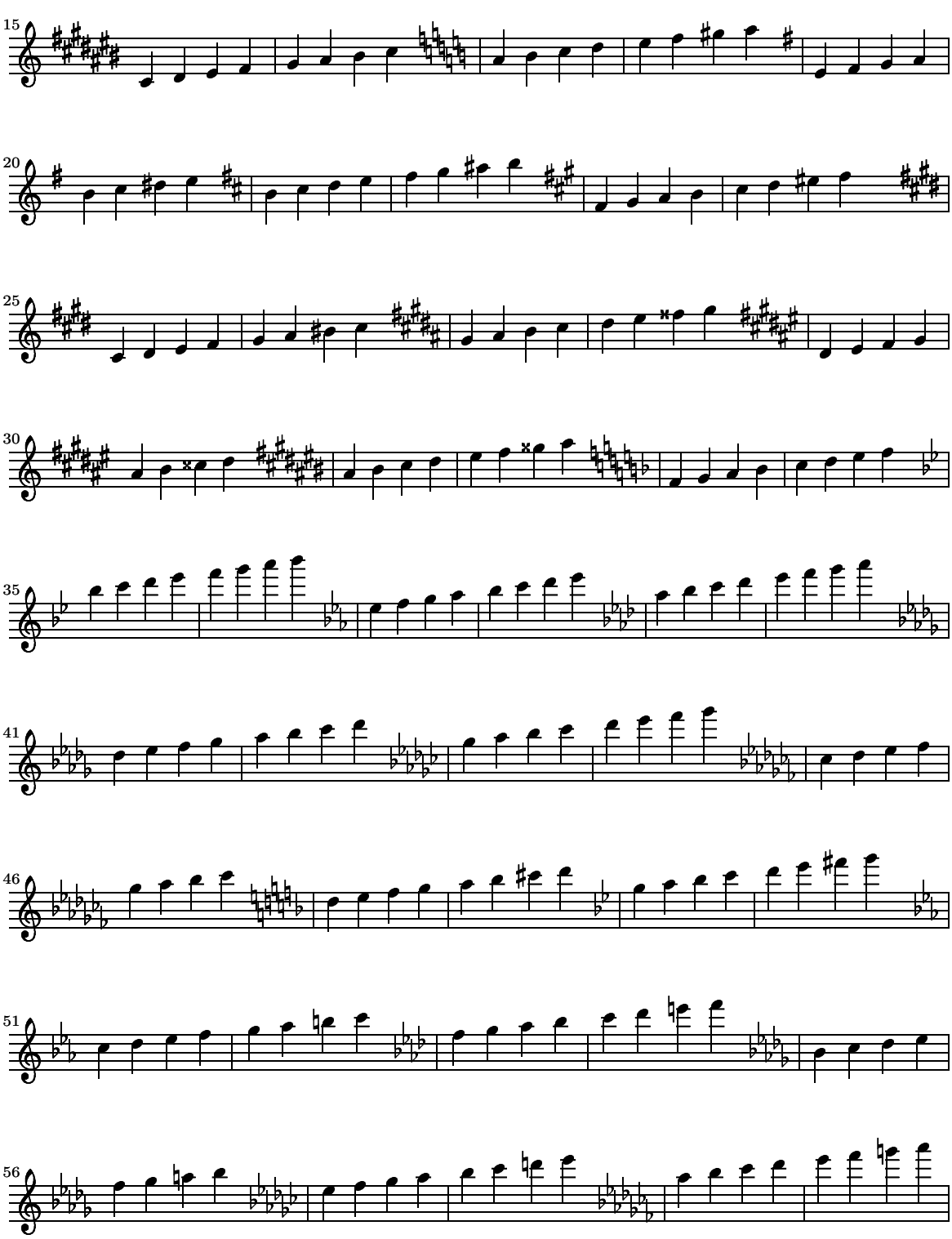


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/metronome.ly’:



MIDI and midi2ly test file. Diff between this and midi2ly.py (experimental, 1.5.17) should be minimal ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/midi-scales.ly’:



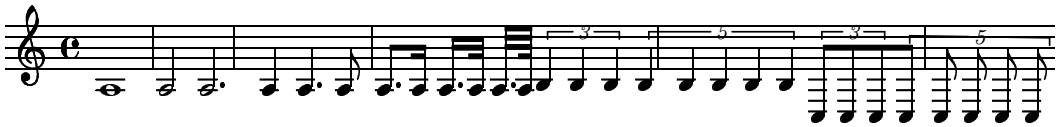


Midi2ly tuplet test.

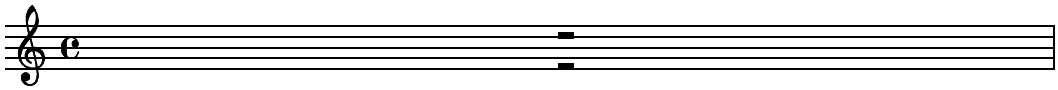
```
python scripts/midi2ly.py --duration-quant=32 \
    --allow-tuplet=4*2/3 \
    --allow-tuplet=8*2/3 \
```

```
        --allow-tuplet=4*3/5 \
        --allow-tuplet=8*3/5 \
        tu.midi
```

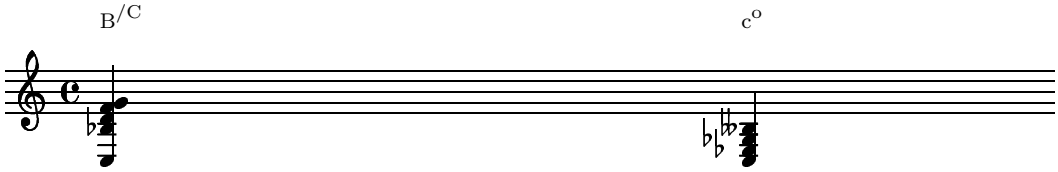
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/midi-tuplets.ly’:



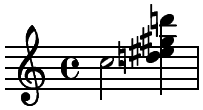
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/mmrest-collision.ly’:



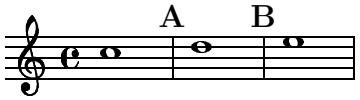
FIXME ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/monstrous.ly’:



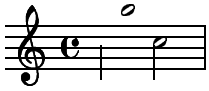
Positions of accidentals may be manually set. This involves some scheme code. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/move-accidentals.ly’:



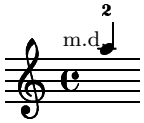
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/move-mark.ly’:



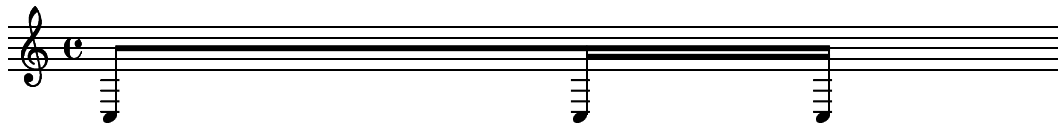
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/move-notehead.ly’:



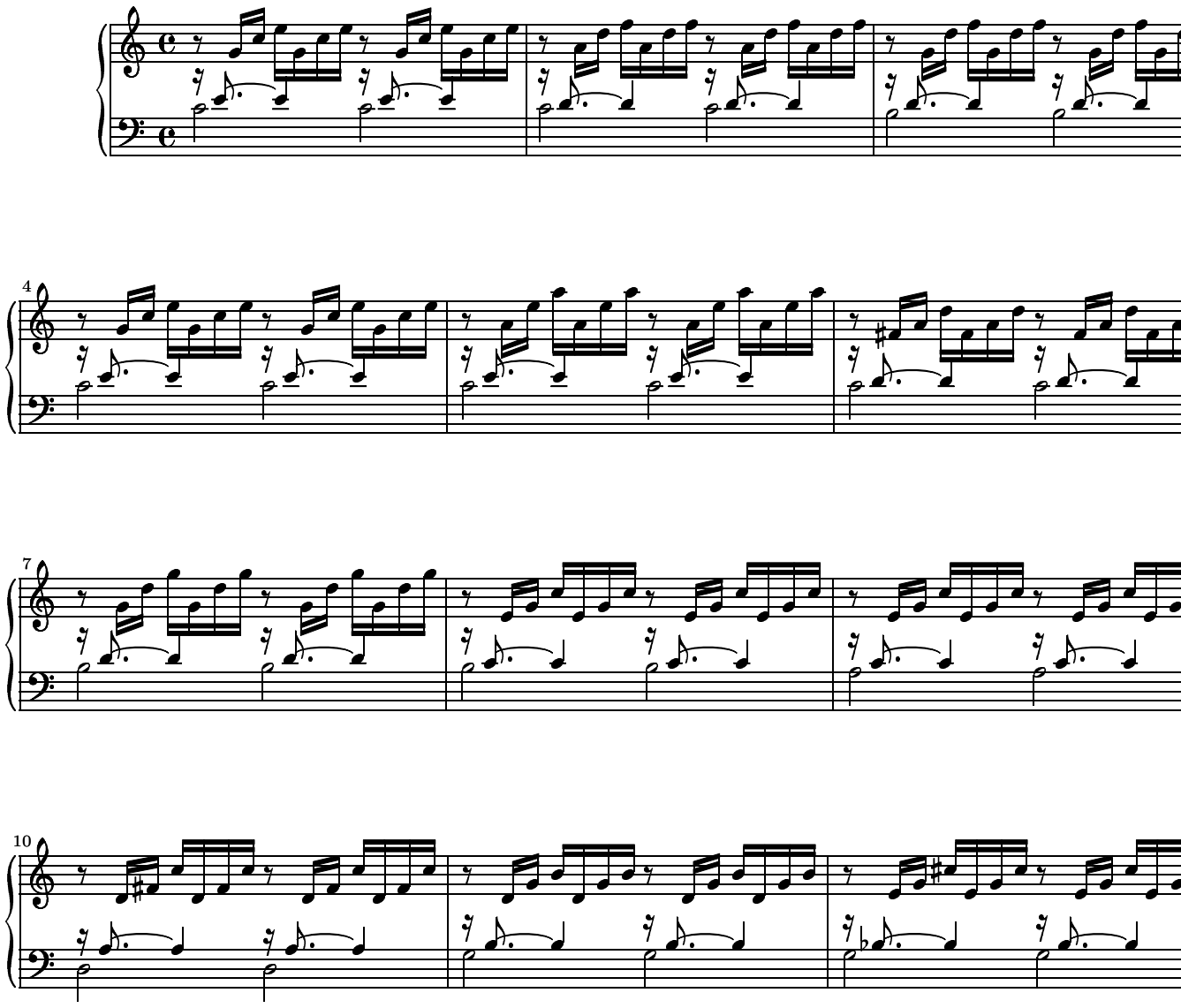
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/move-specific-text.ly’:

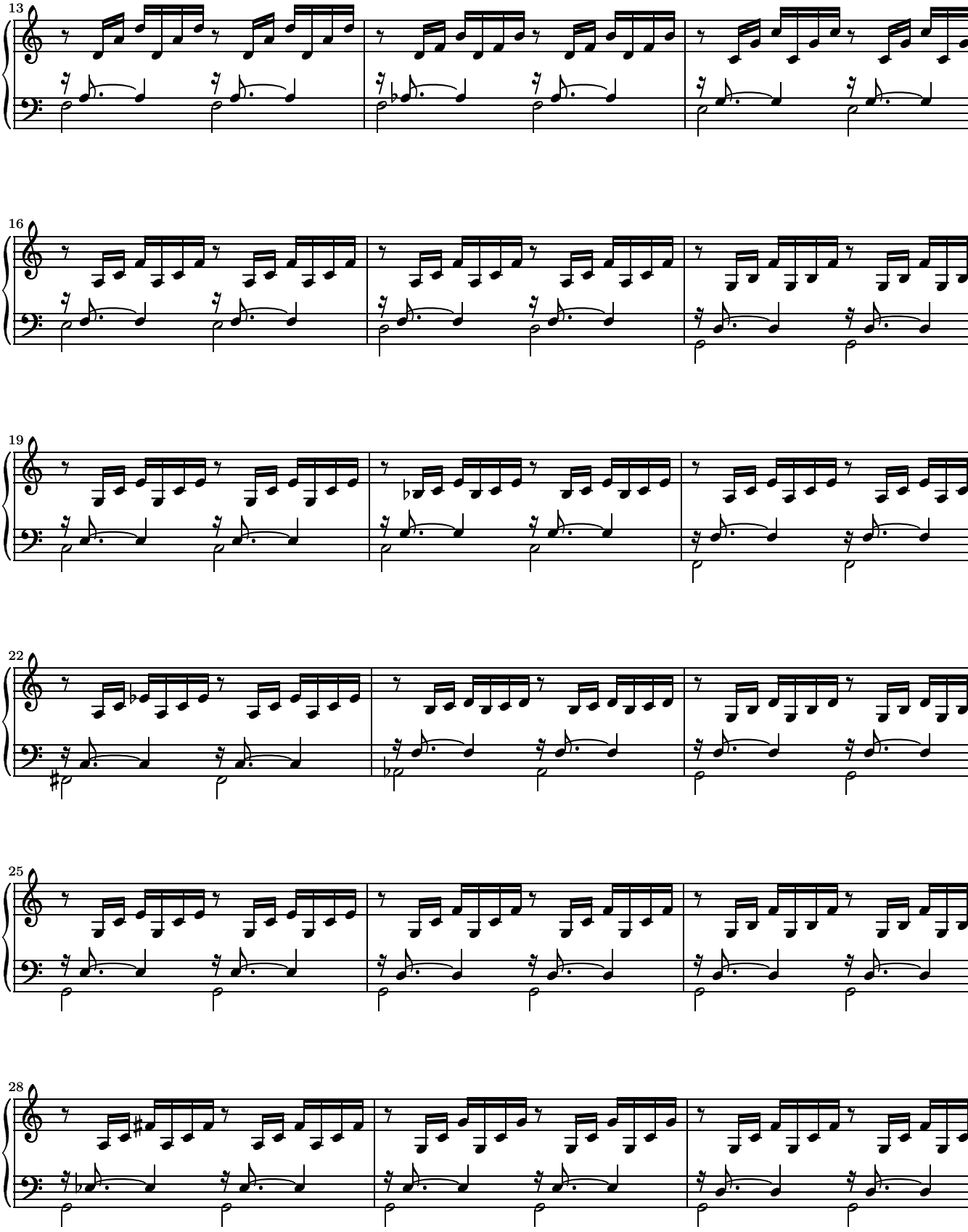


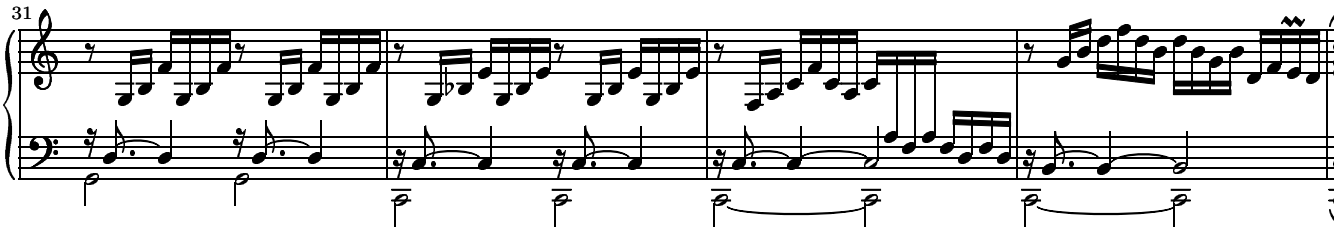
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/multi-slope.ly’:



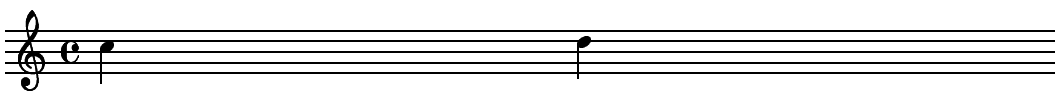
This example shows prelude in C major of WTK1, but coded using Scheme functions to save typing work. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/music-box.ly’:



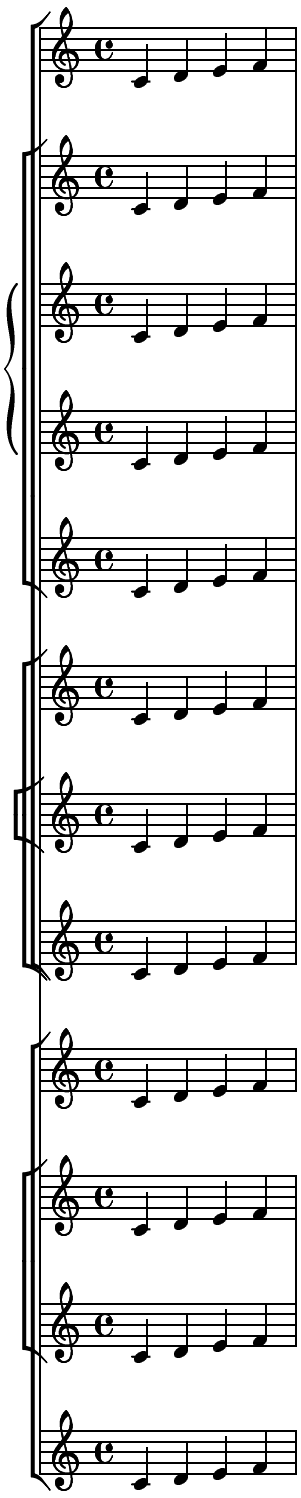

13
16
19
22
25
28



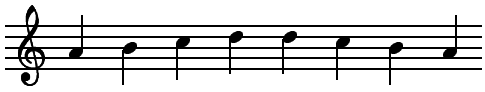
You can create music expressions from Scheme. The mechanism for this is rather clumsy to use, so avoid if possible. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/music-creation.ly`’:



In InnerStaffGroup and InnerChoirStaff, the brackets should be shiftet leftwards. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/nested-groups.ly`’:

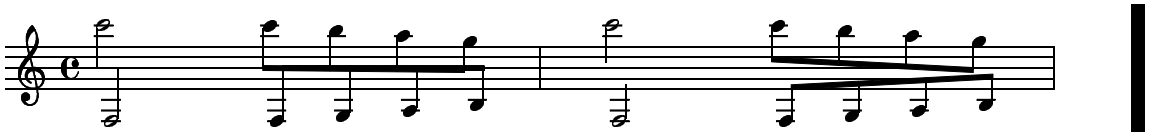


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/no-bar-lines.ly’:

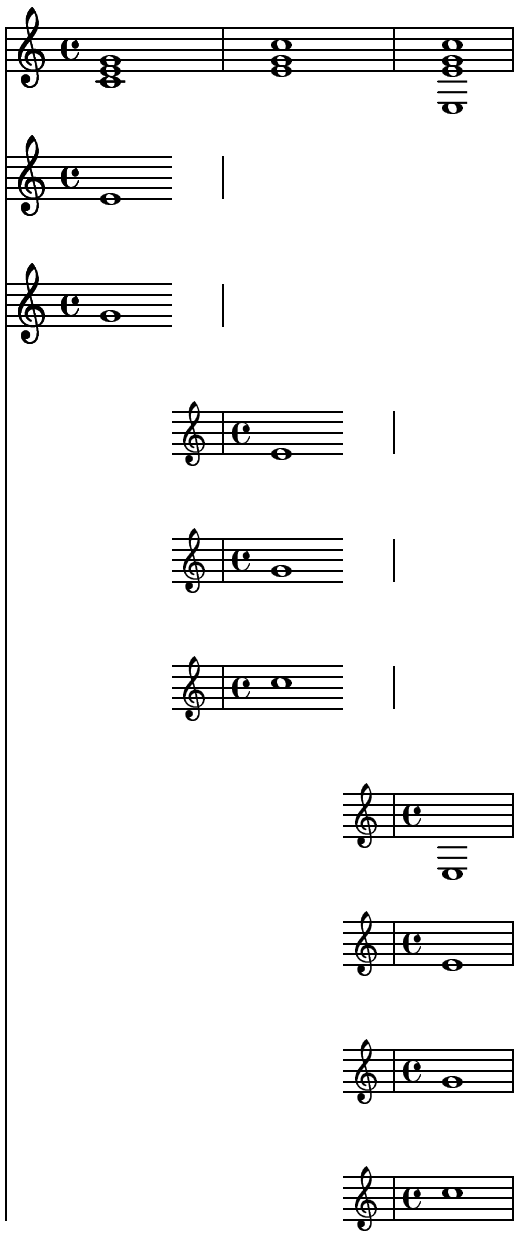


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/no-staff.ly’:

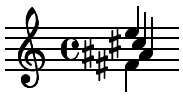
Don’t extend stems to cetner line. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/no-stem-exte



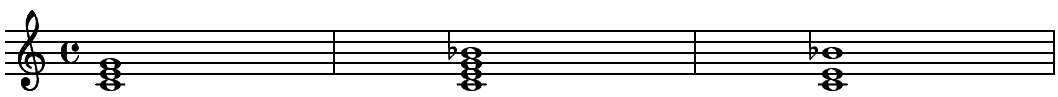
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/note-chords.ly’:



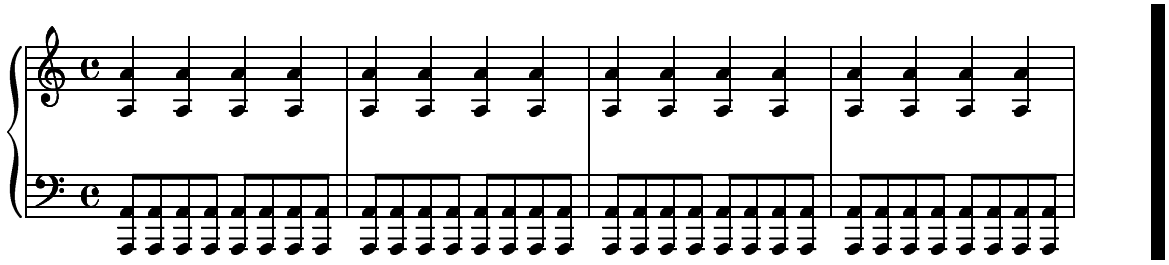
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/note-shift.ly’:



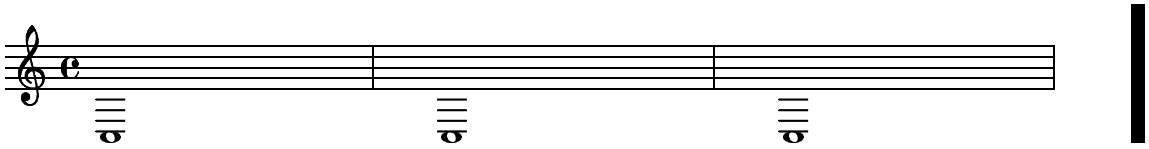
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/notemode-chords.ly’:

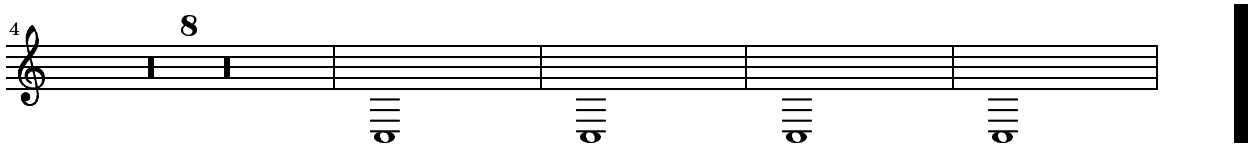


Octave duplicate parts of music ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/octave-duplica

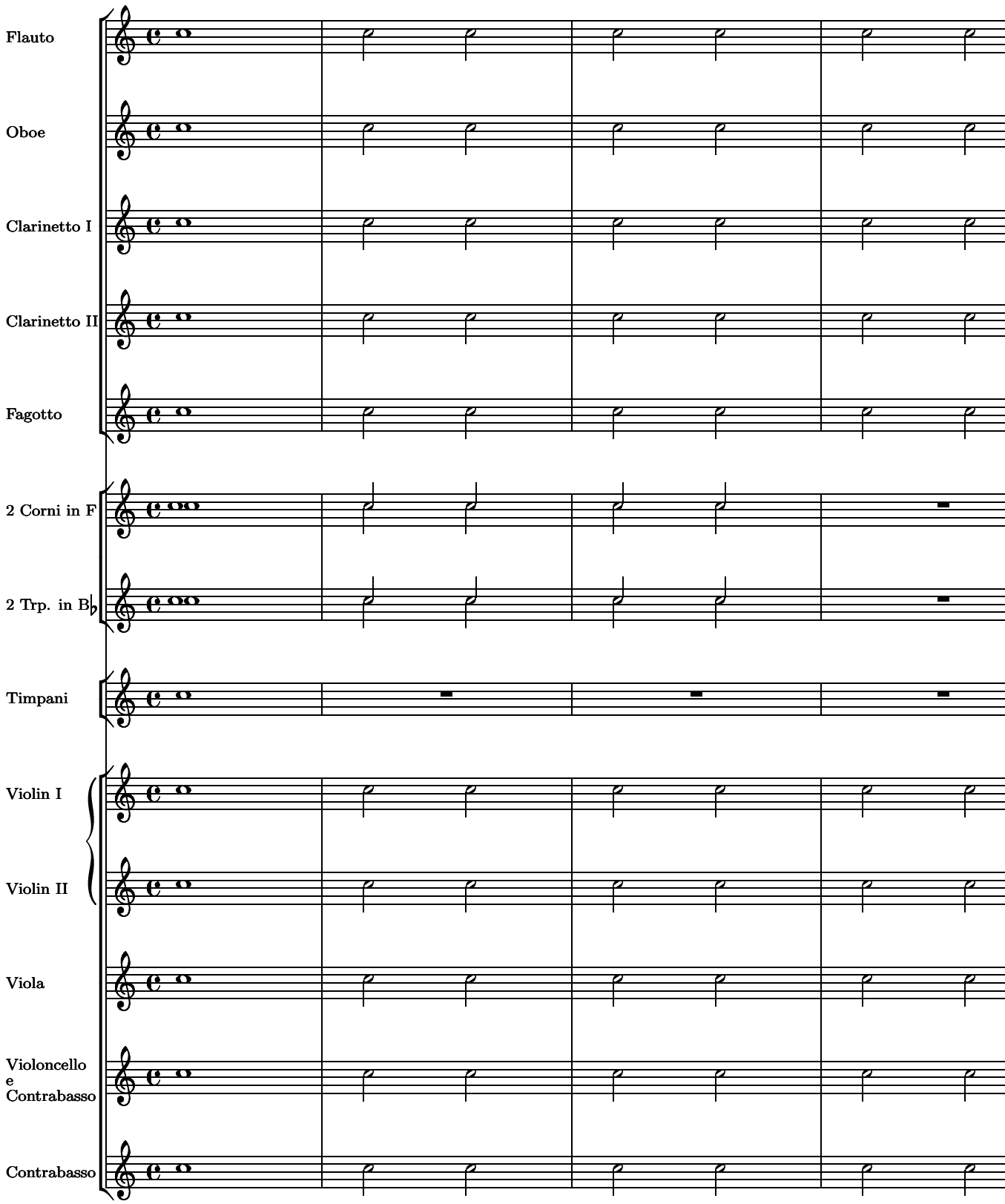


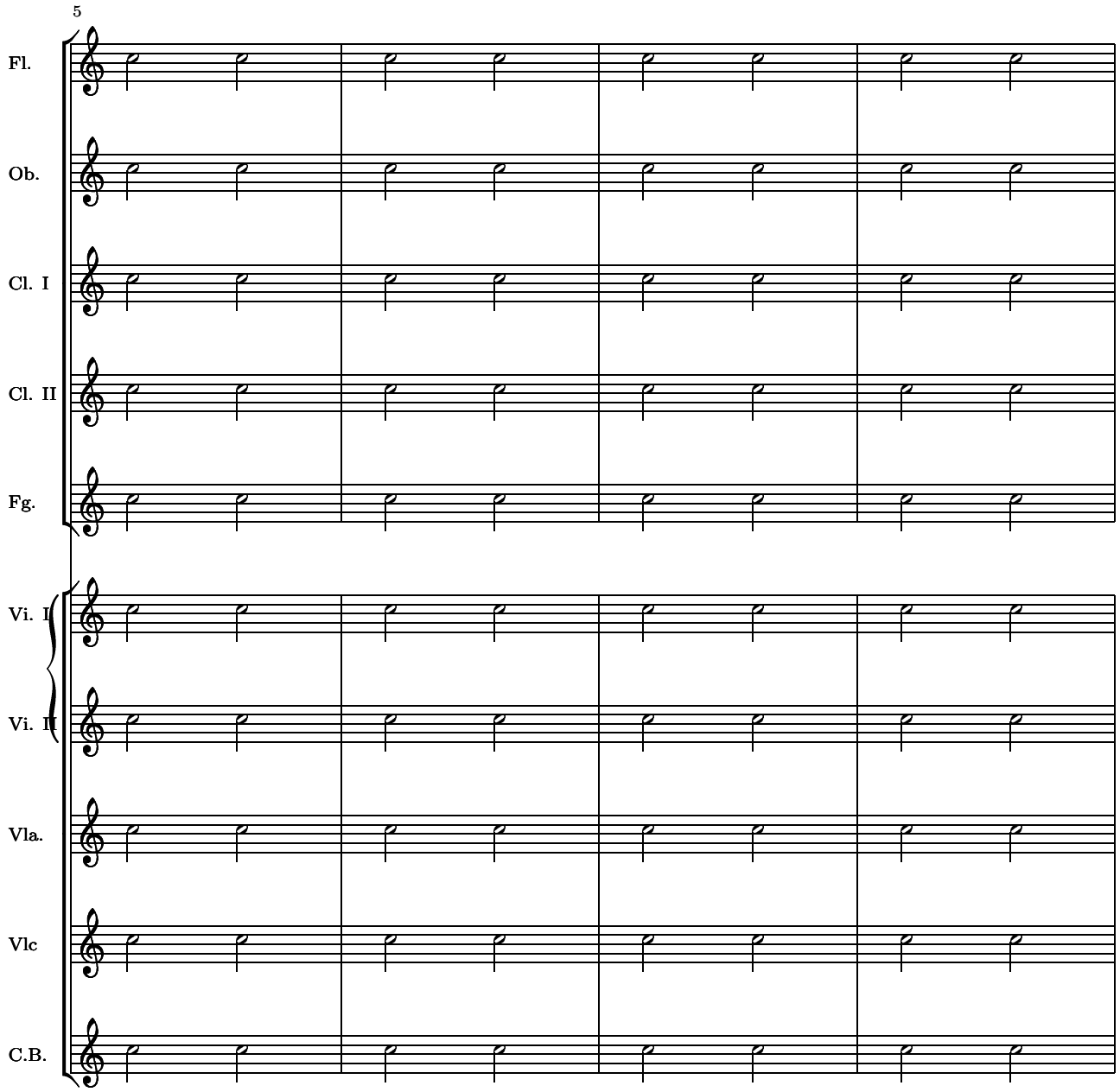
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/orchestalpart.ly’:



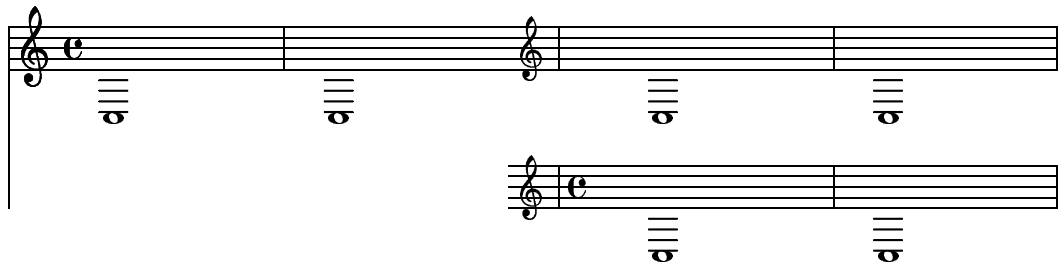


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/orchestscore.ly’:

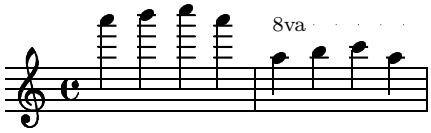

Flauto
Oboe
Clarinetto I
Clarinetto II
Fagotto
2 Corni in F
2 Trp. in B♭
Timpani
Violin I
Violin II
Viola
Violoncello
e
Contrabasso
Contrabasso



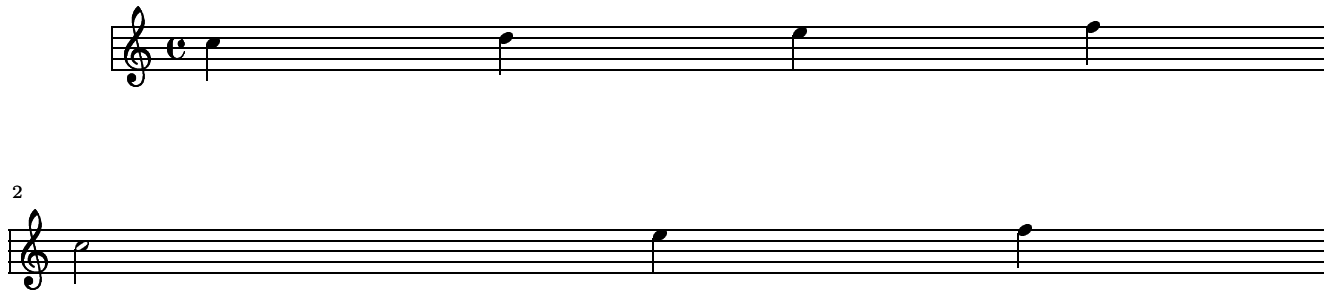
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ossia.ly’:



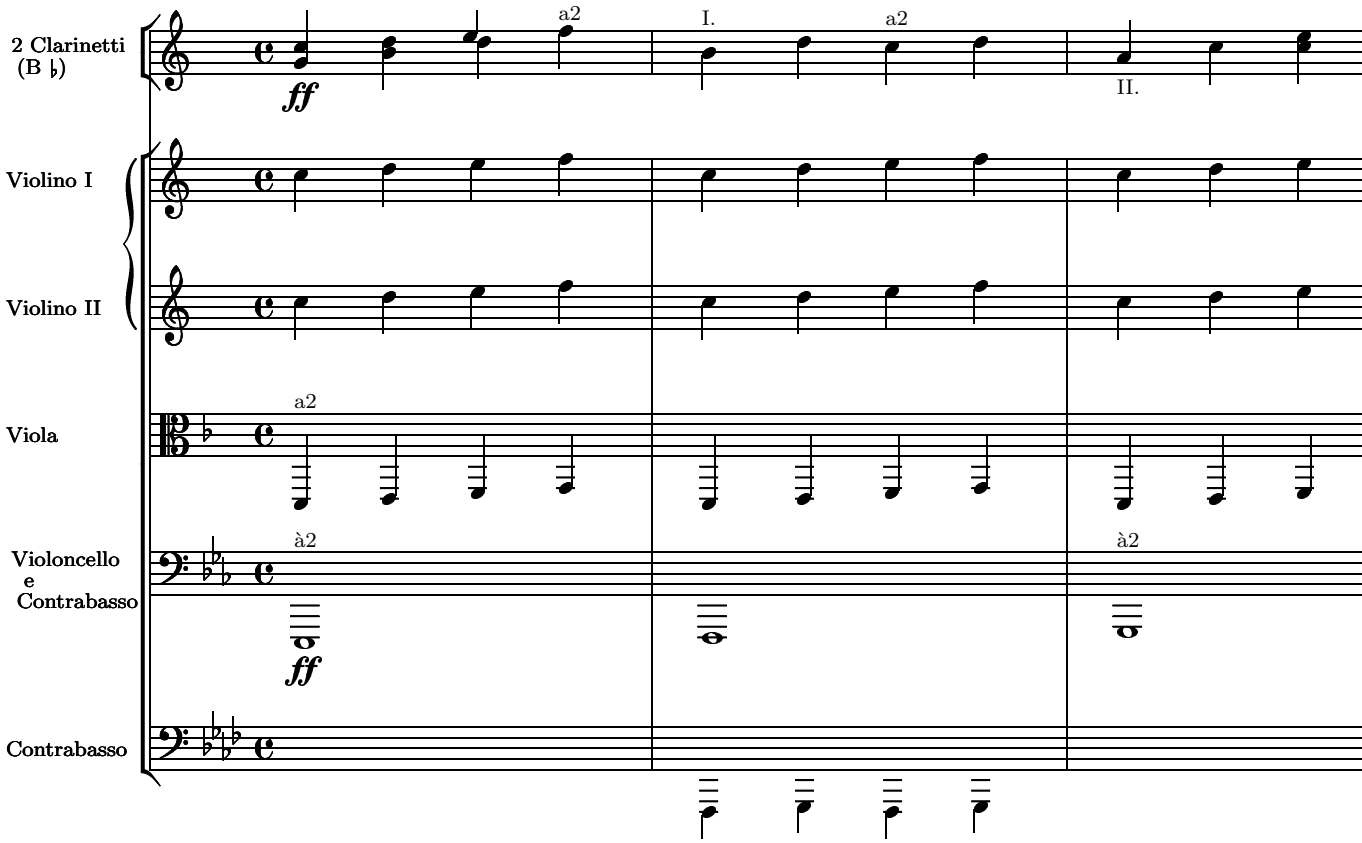
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/ottava.ly’:

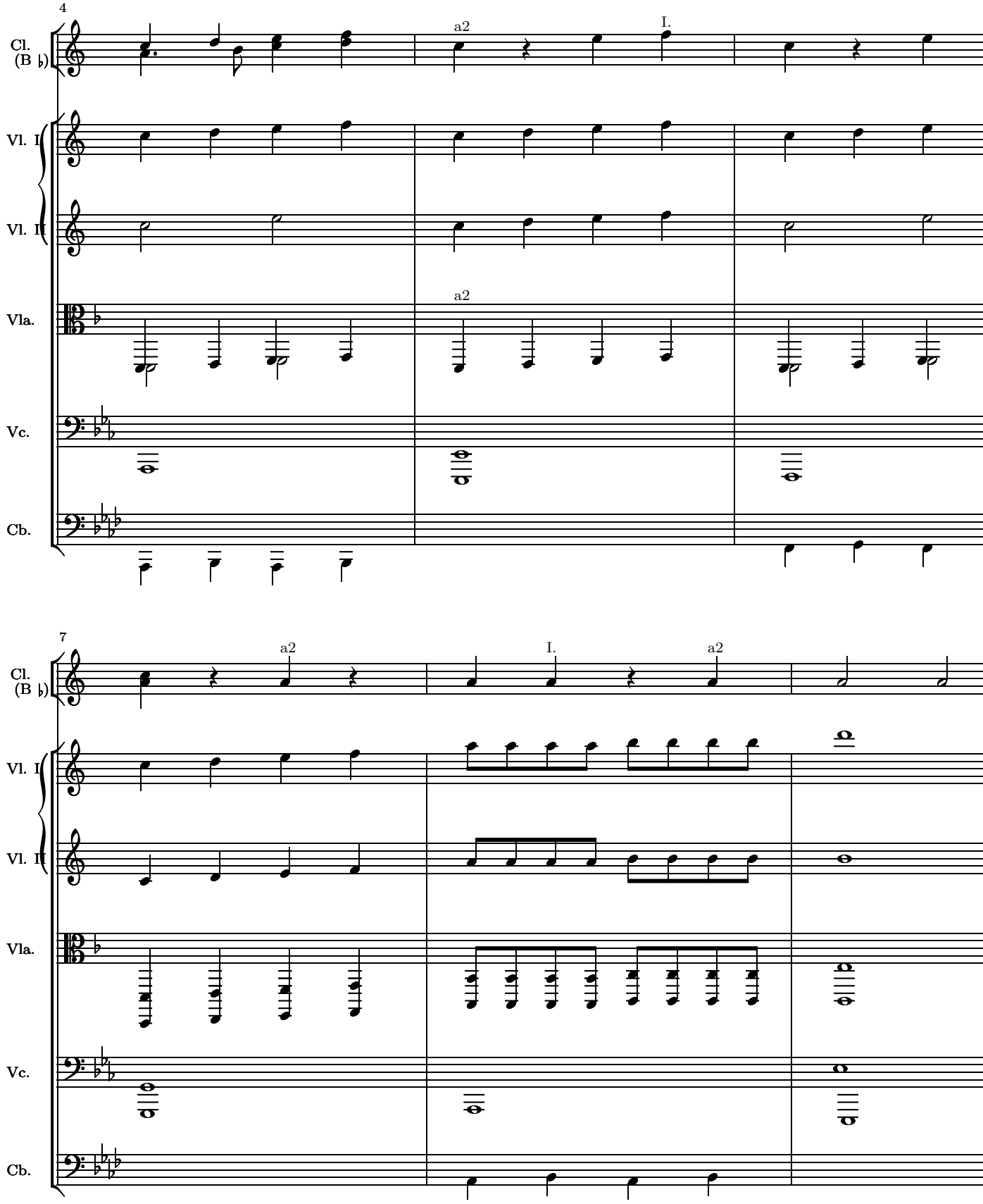


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/part-combine-moments.ly’:

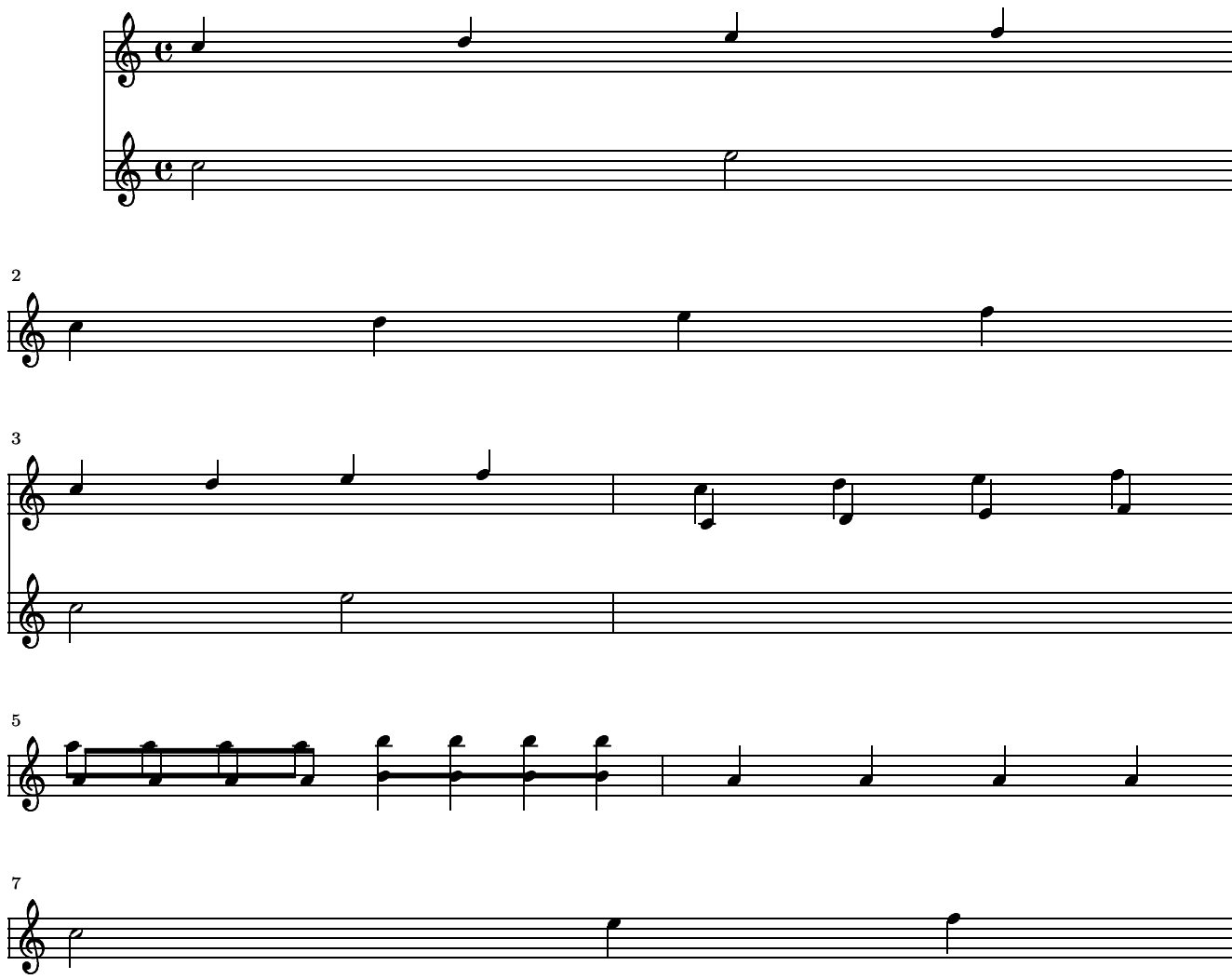


Template for part-combining orchestral scores ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/pa

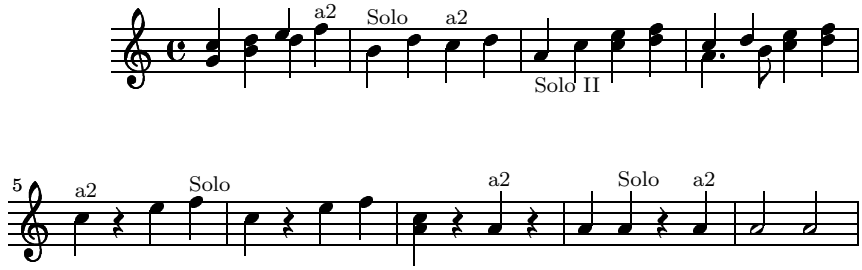



4
Cl.
(B ♭)
Vl. I
Vl. II
Vla.
Vc.
Cb.
a2
I.
a2
7
Cl.
(B ♭)
Vl. I
Vl. II
Vla.
Vc.
Cb.
a2
I.
a2

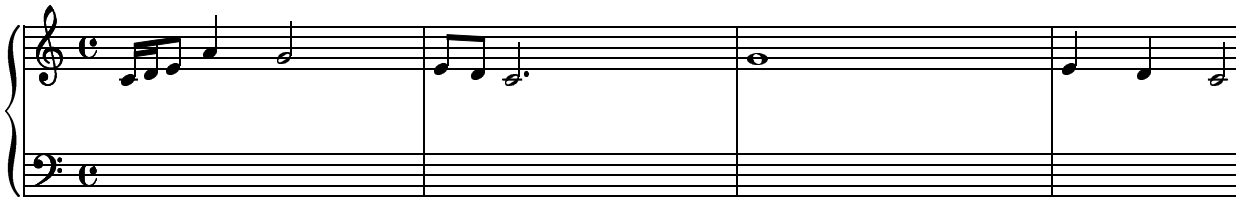
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/part-combine-staff.ly’:



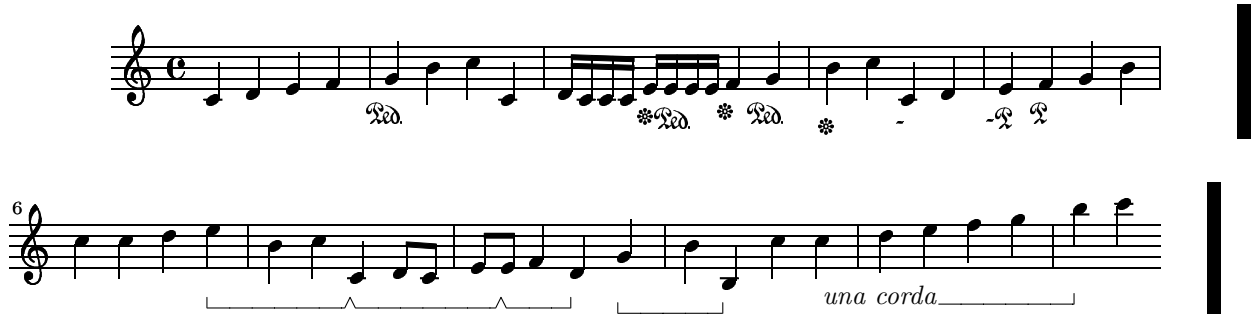
In orchestral scores and hymns, voices are traditionally combined onto one staff. LilyPond has a part combiner, that combines or separates two voices according to actual rhythm and pitch. User-defined texts such as “solo” and “à2” are typeset automagically, as appropriate. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/part-combine.ly’:



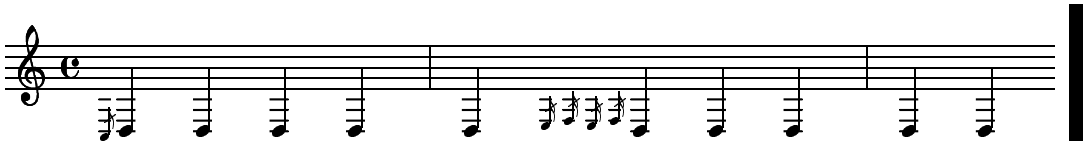
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/partial-blank.ly’:



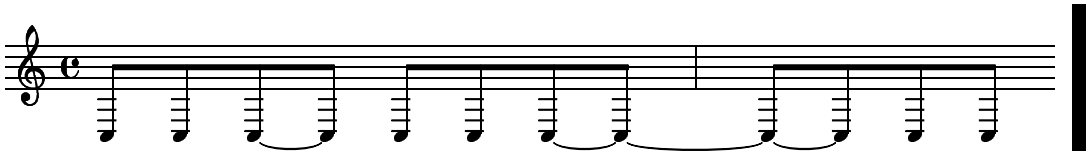
Piano pedal symbols merge stop and start. The strings are configurable. Text style, bracket style, and a mixture of both are supported. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/pedal.ly’:



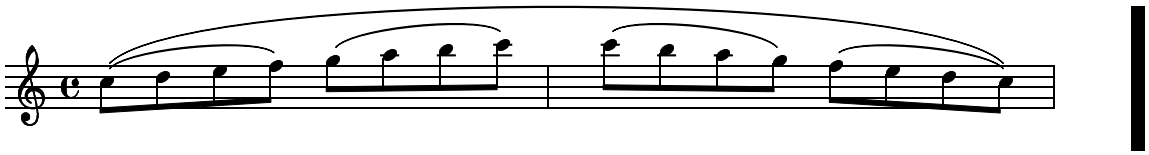
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/perform-grace.ly’:



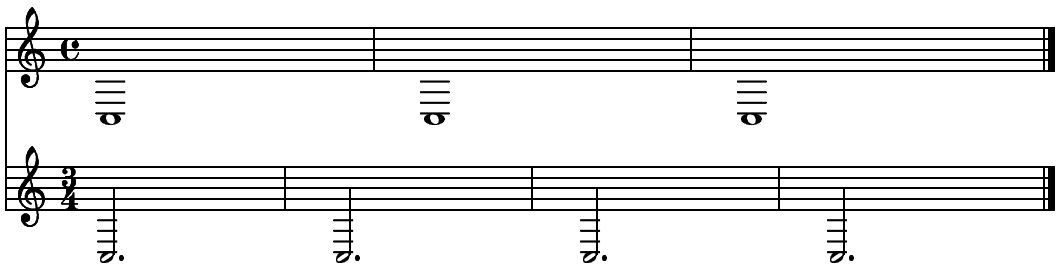
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/perform-tie.ly’:



Make PhrasingSlur higher, to avoid colission from other slurs ‘/home/inger/RPM/BUILD/lilypond-1.6.6/i



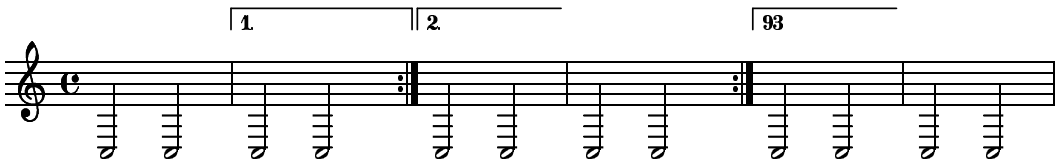
Polymetric music with aligned note values can be written by moving the timing engraver to staff context. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/poly-metric.ly’:



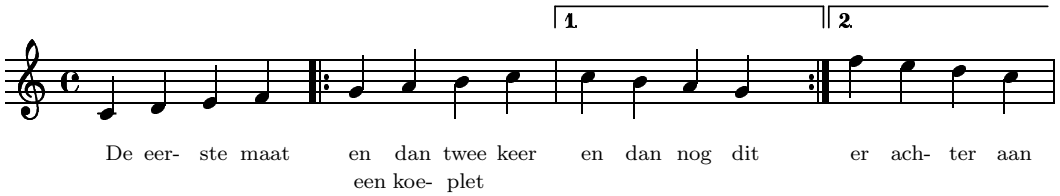
Grob extents may be hard coded using grob properties. This requires Grob::preset_extent () function. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/preset-extent.ly’:

foo bar baz

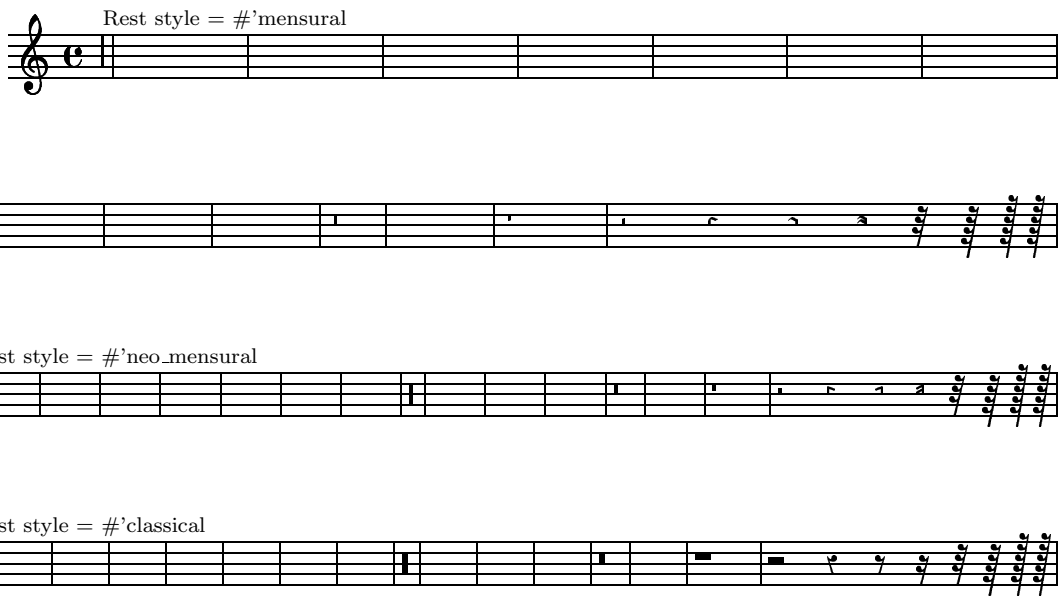
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/repeat-manual.ly’:

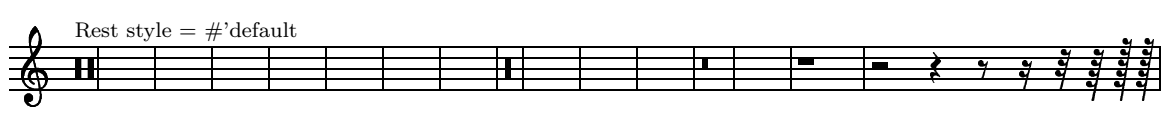


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/repeat.ly’:

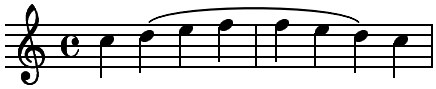


rests in various styles. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/rests.ly’:



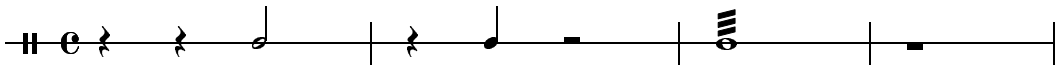


Simple customised music apply. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/reverse-music.`

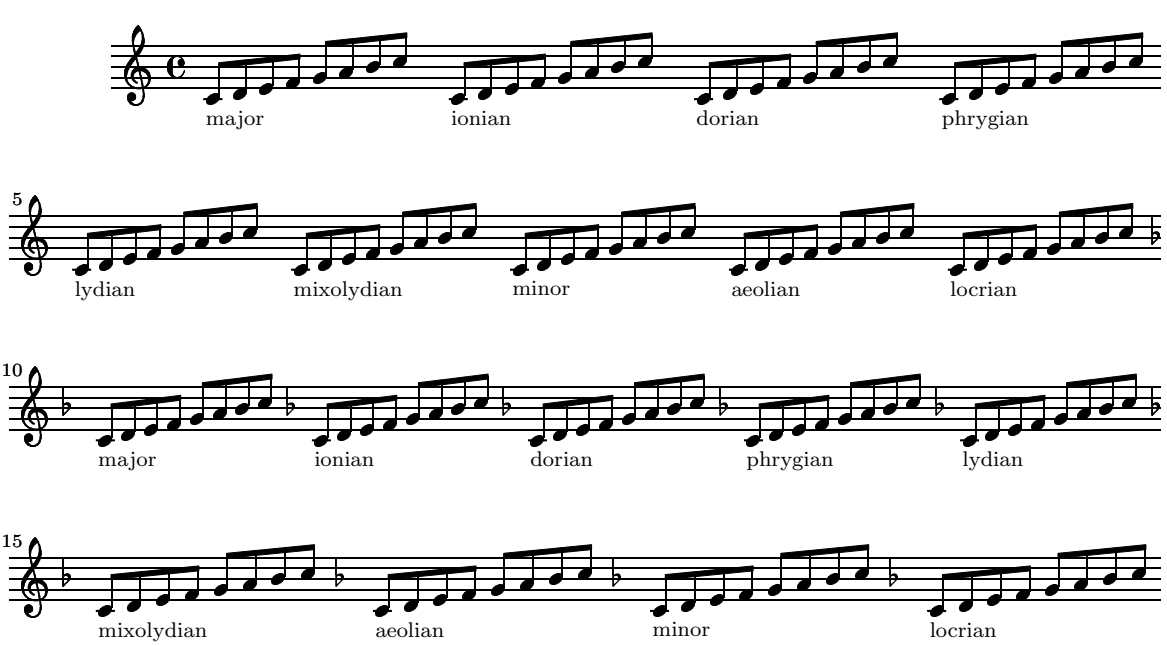


a way to generate rhythm exercises with lilypond (e.g. no staff but retaining the barlines ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/rhythm-excercise.ly`’:

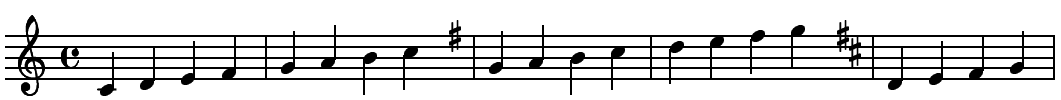
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/rhythmic-staff.ly`’:

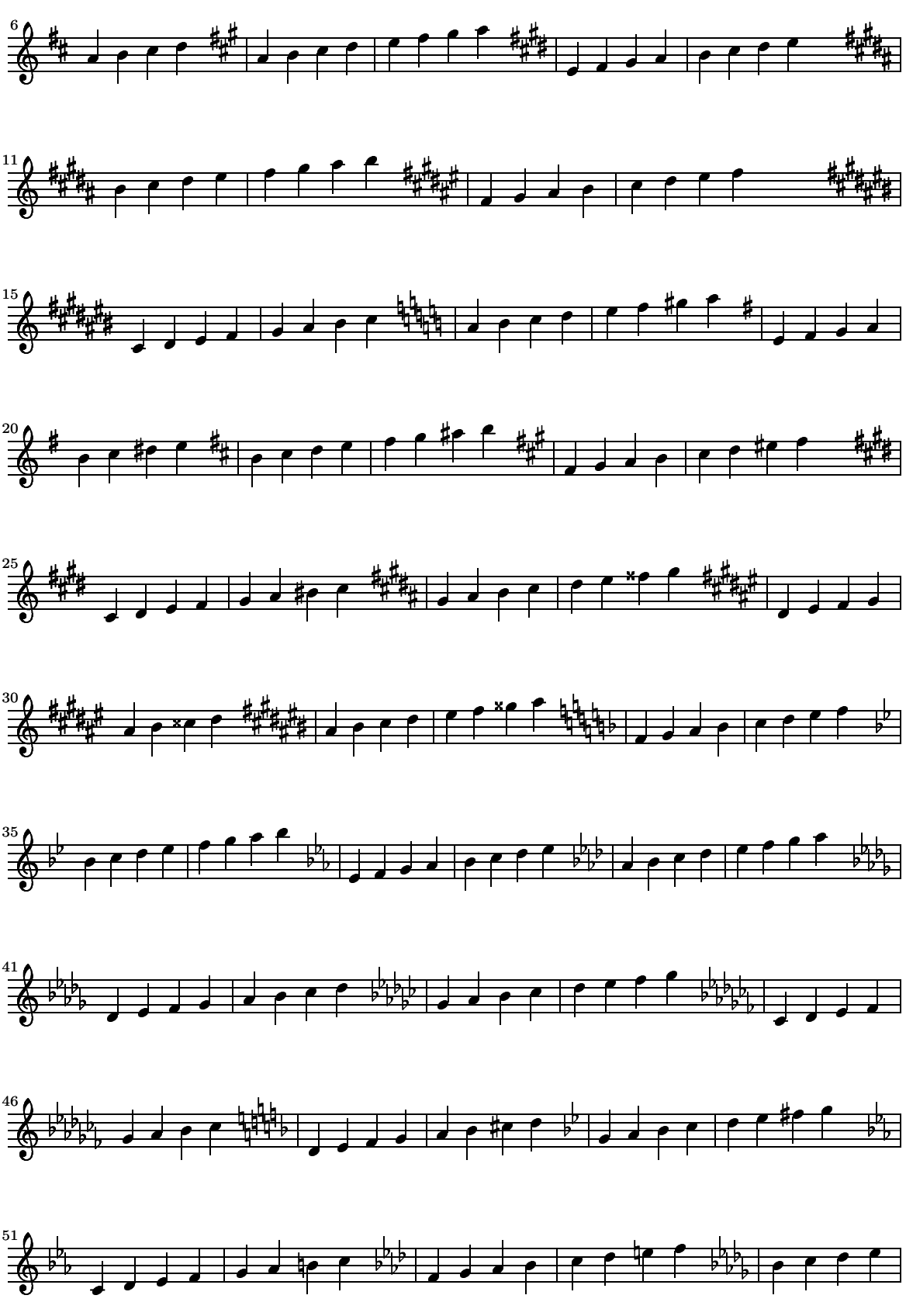


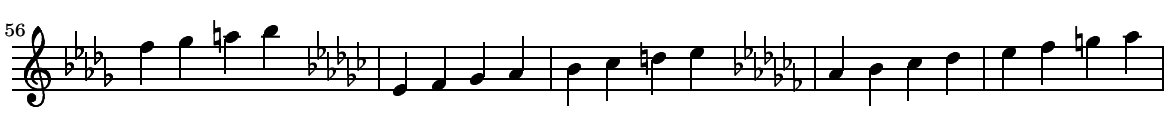
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/scales-greek.ly`’:



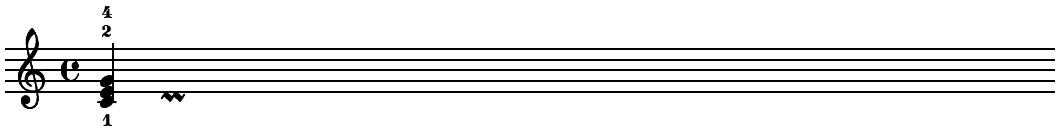
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/scales.ly`’:



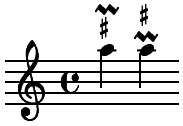




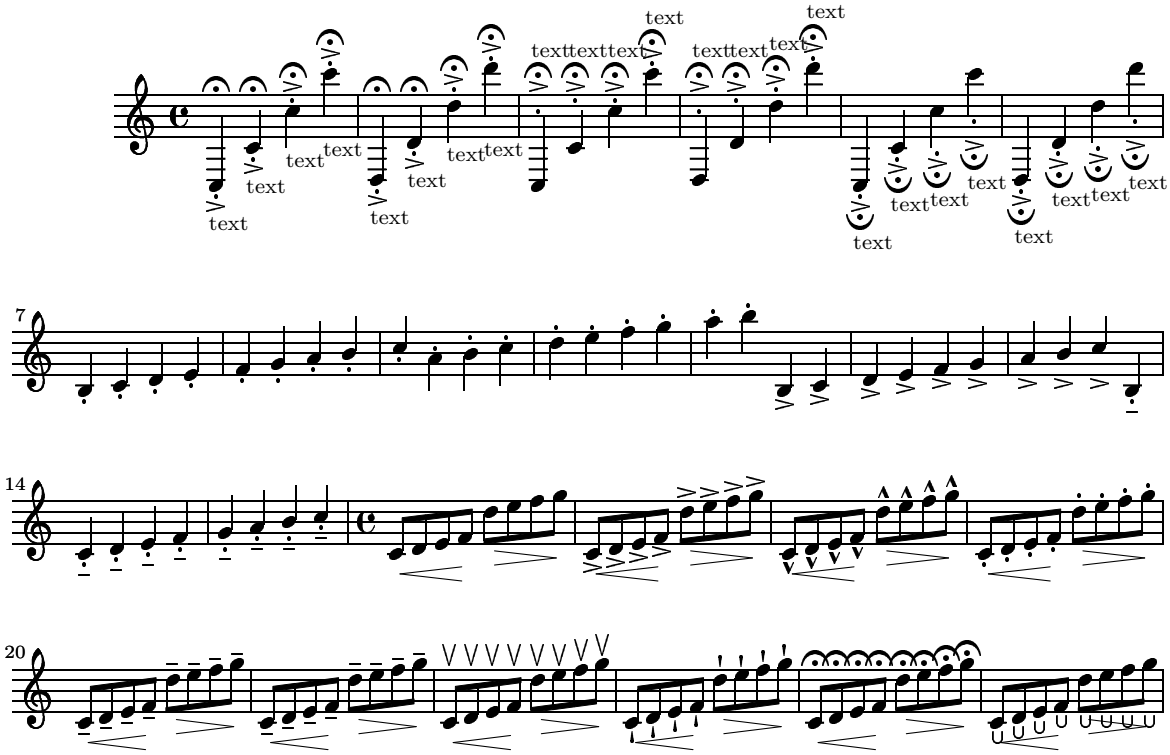
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/script-horizontal.ly’:



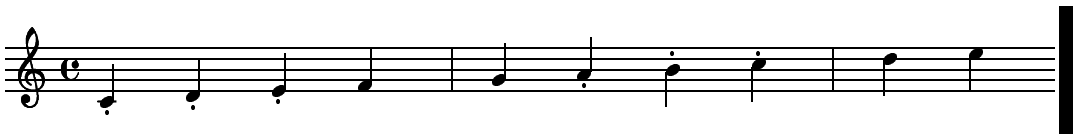
Relative placements of different script types can be controlled by overriding script-priority. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/script-priority.ly’:



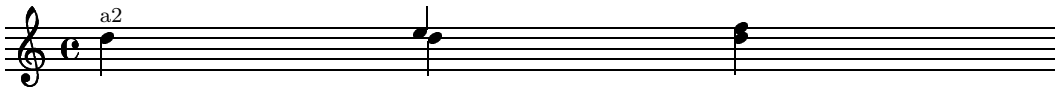
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/scripts.ly’:



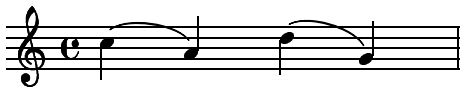
You can enter notes and articulations separately, and merge them into one thread. Here is an example to add repeated staccato dots. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/separate

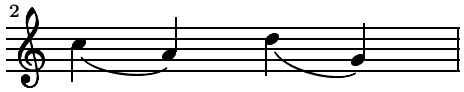


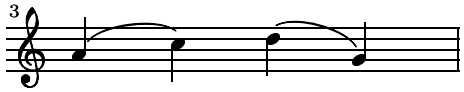
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/short-part-combine.ly’:

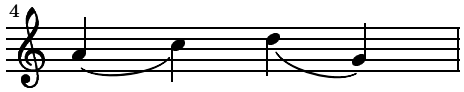


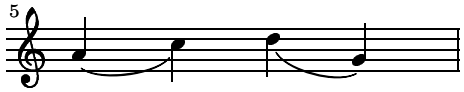
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/simple-slur-endings.ly’:



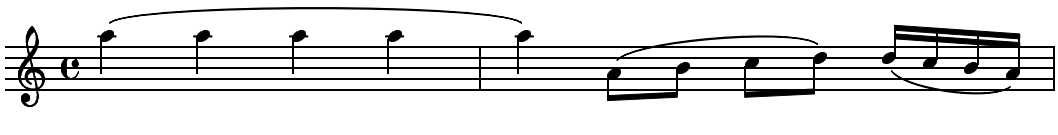




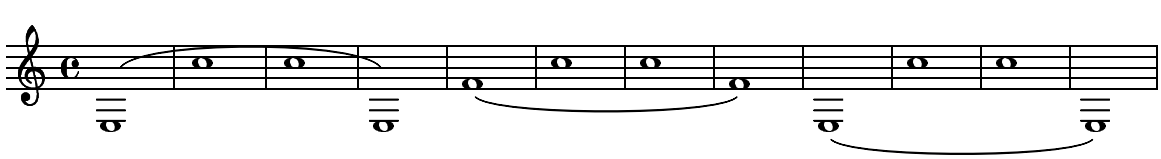


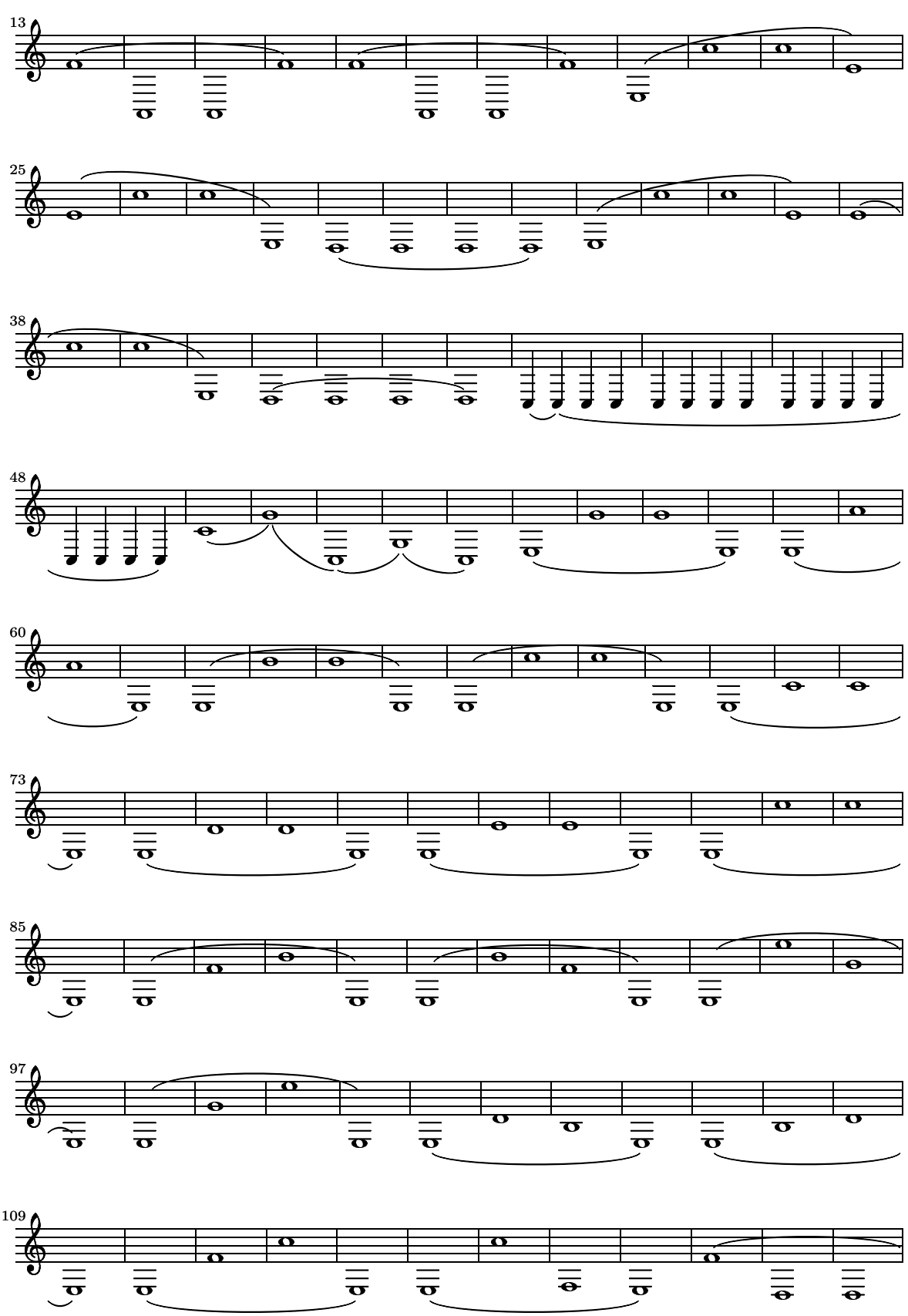


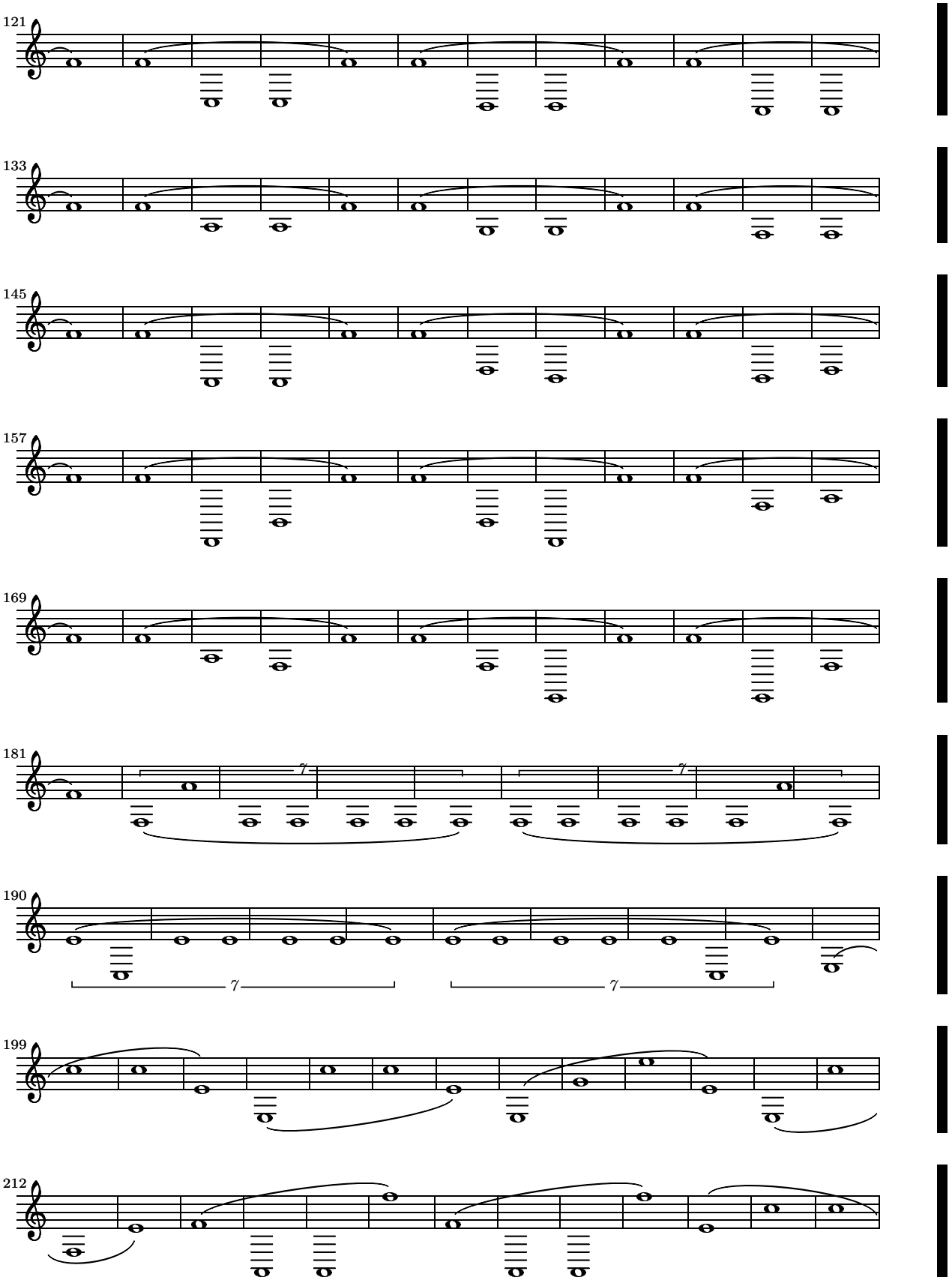
sketch output supported features ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/sketch.ly’:

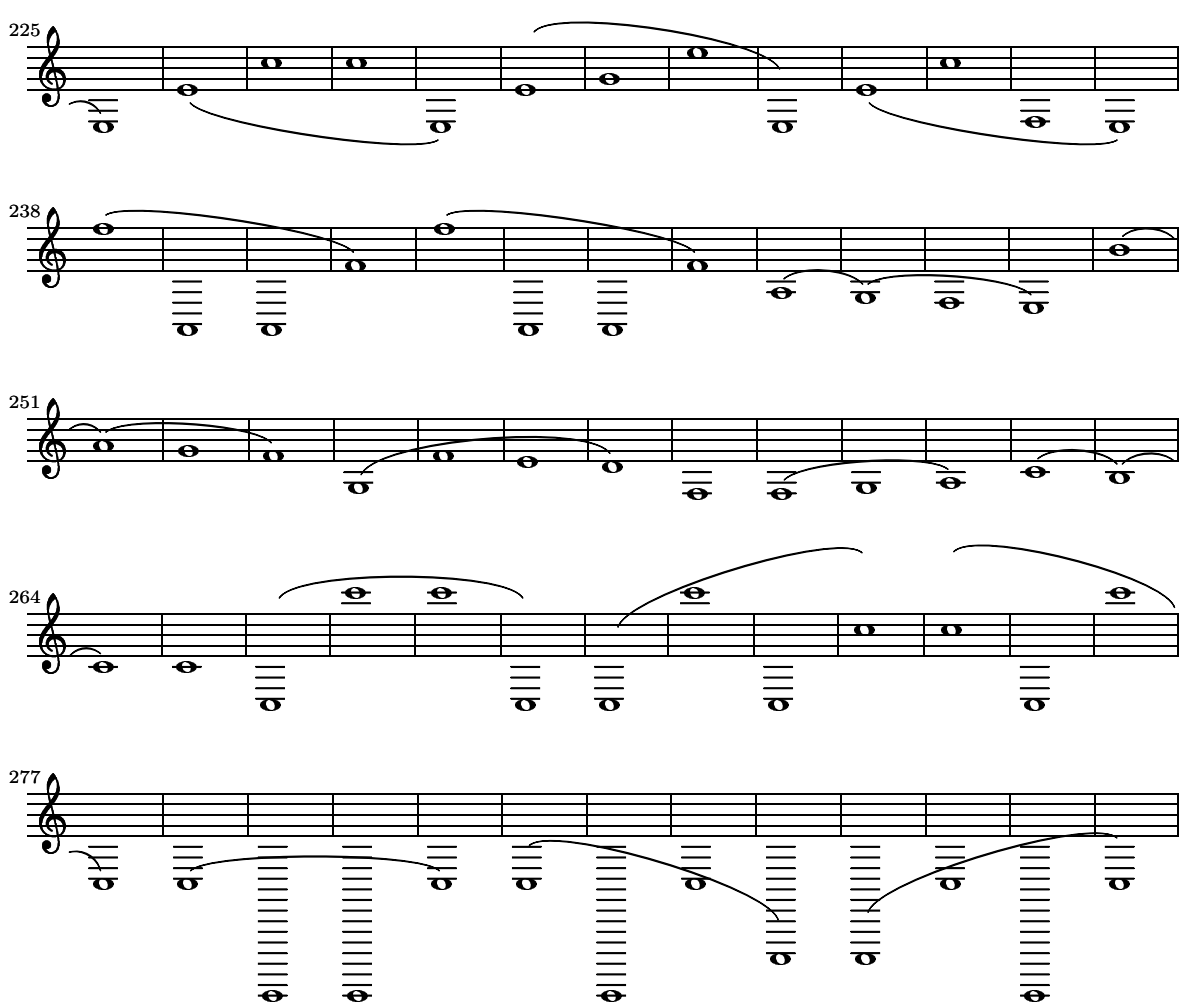


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/sleur.ly’:

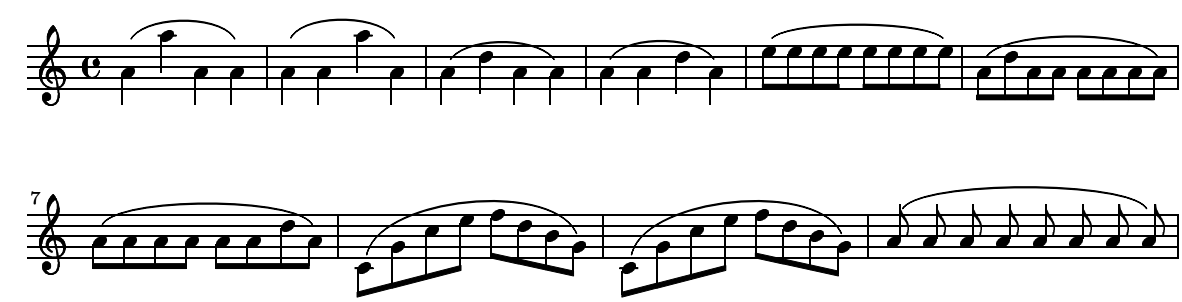



13
25
38
48
60
73
85
97
109


121
133
145
157
169
181
7
7
190
7
7
199
212

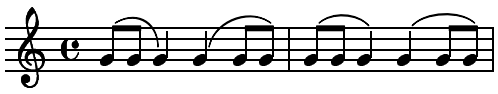


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-area.ly’:

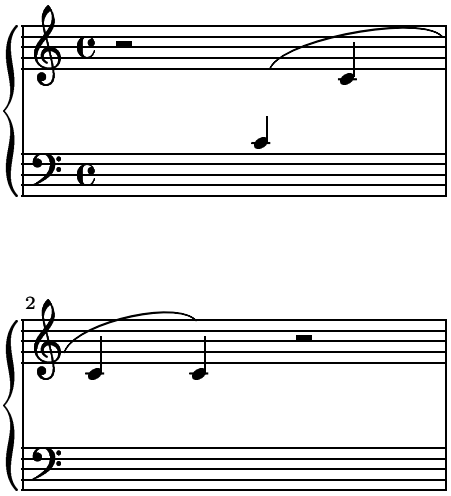


In some cases, you may want to set slur attachments by hand. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/i

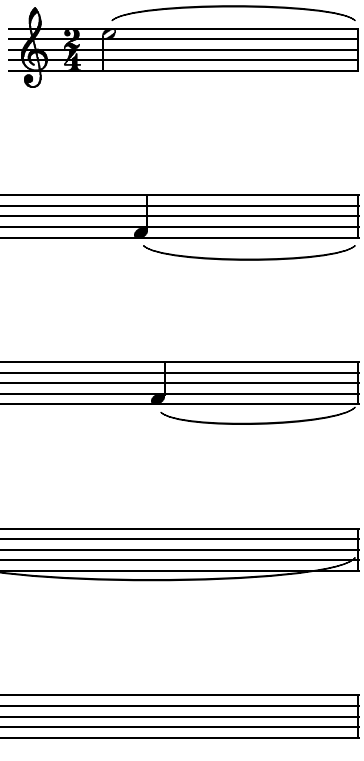
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-attachment.ly’:



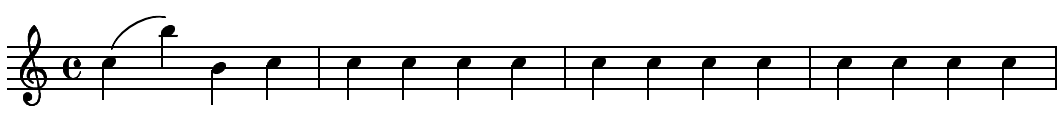
The same goes for slurs. They behave decently when broken across linebreak. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-broken-cross-staff.ly’:



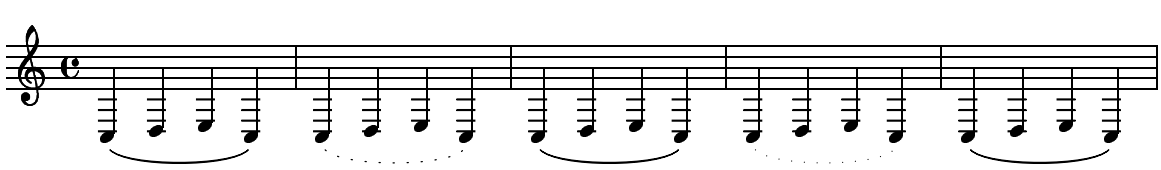
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-broken.ly’:



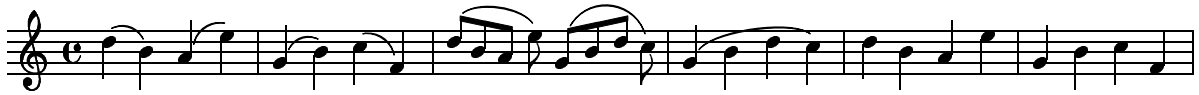
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-damping.ly’:



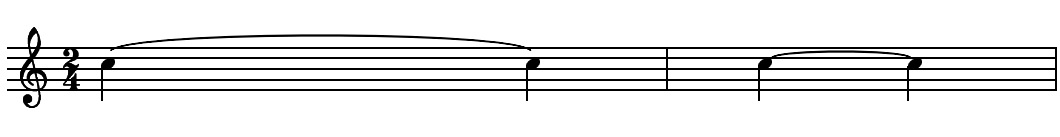
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-dash.ly’:



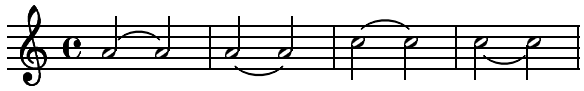
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-follow-music.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-mininum.ly’:

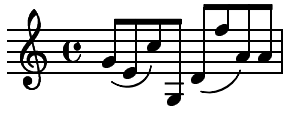


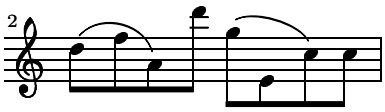
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-positions.ly’:



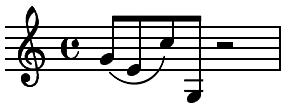
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-proof.ly’:

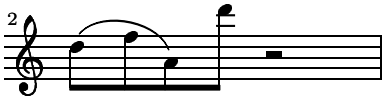
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-symmetry-2.ly’:



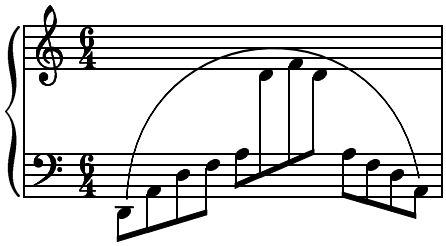


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-symmetry-3.ly’:





You can get ugly slurs, if you want. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/slur-ugly.l



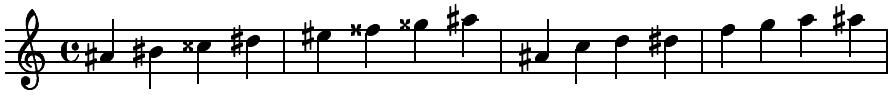
Here’s a copy of my feature request :

> Your task, if you accept it is to implement a \smarttranspose
> command> that would translate such oddities into more natural
> notations. Double accidentals should be removed, as well as E-sharp
> (-> F), bC (-> B), bF (-> E), B-sharp (-> C).

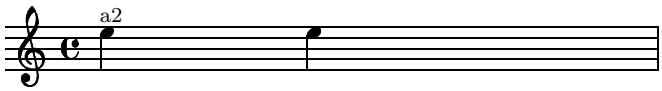
You mean like this. (Sorry ’bout the nuked indentation.)

Modified to use the standard transpose mechanism. The question is
how useful these enharmonic modifications are. Mats B.

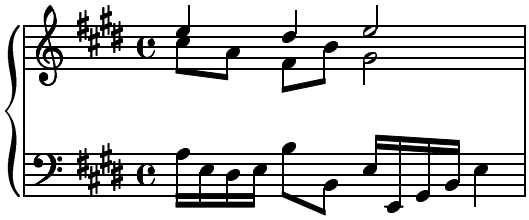
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/smart-transpose.ly’:



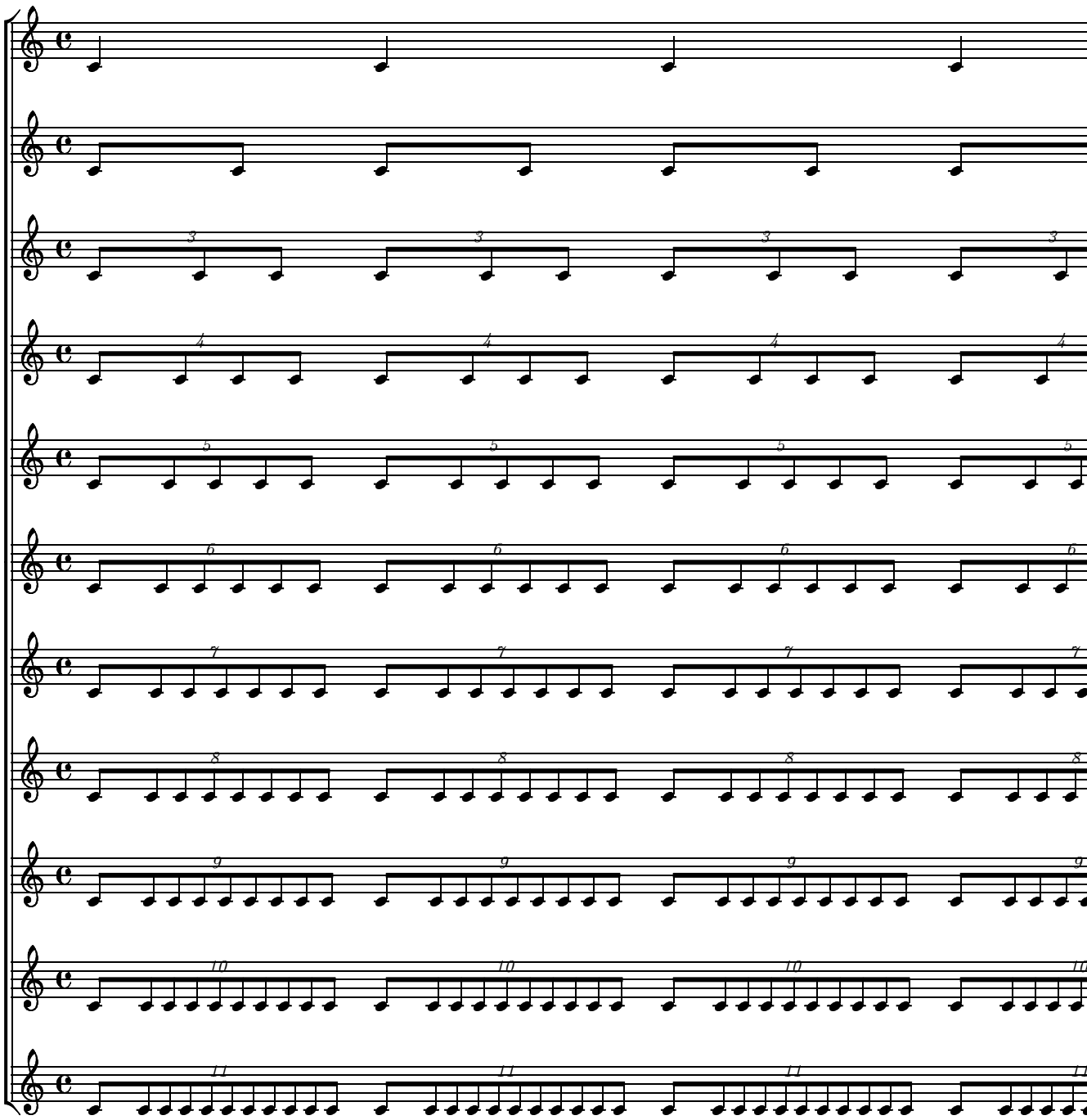
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/solo-a2.ly’:



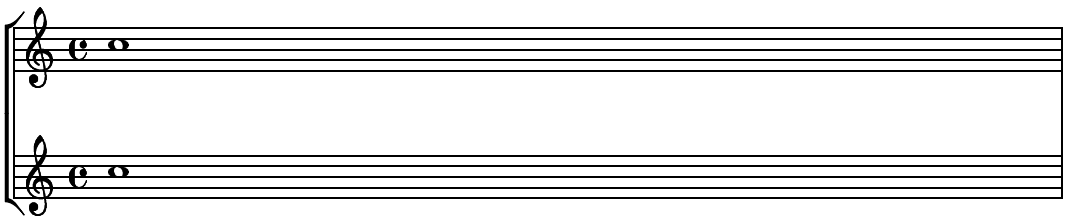
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/spacing-2.ly’:

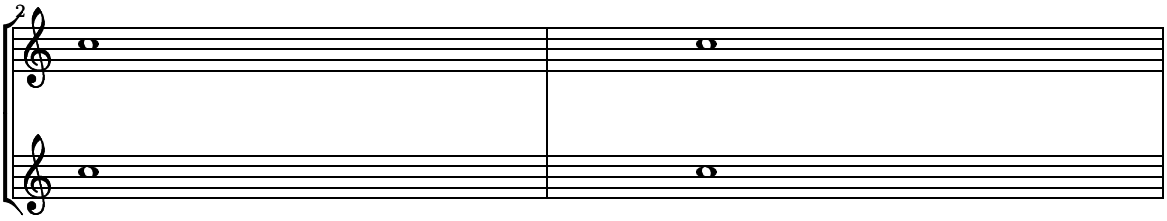


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/spacing.ly’:

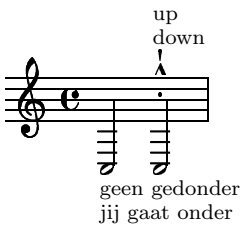


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/span-bars.ly’:

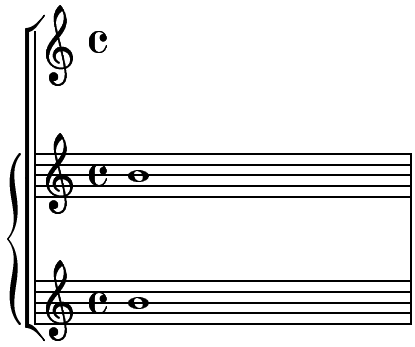




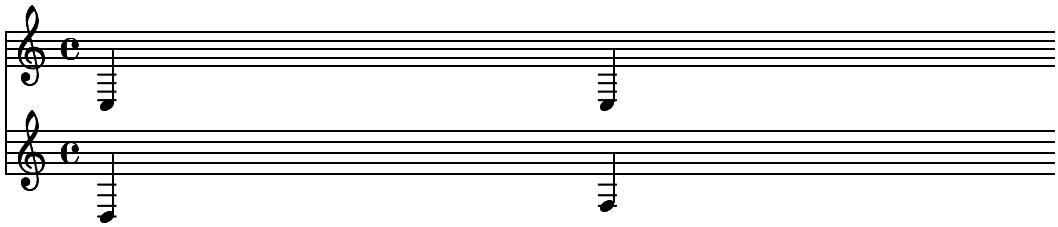
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/stacked-scripts.ly’:



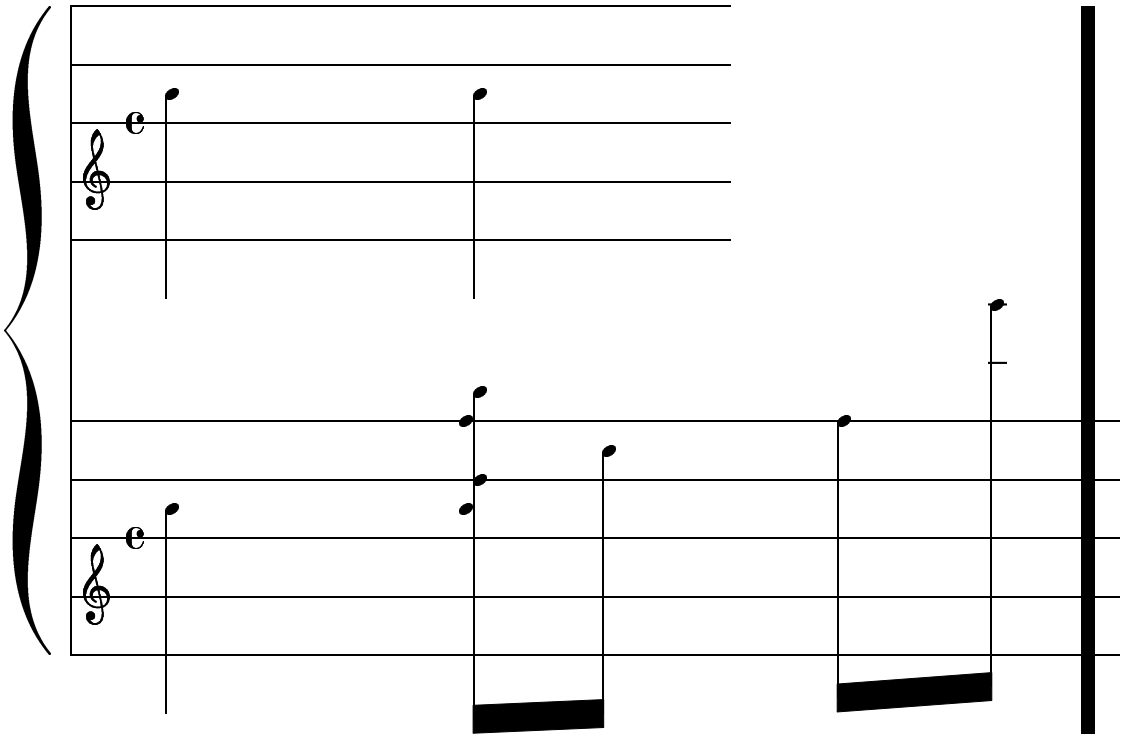
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/staff-bracket.ly’:



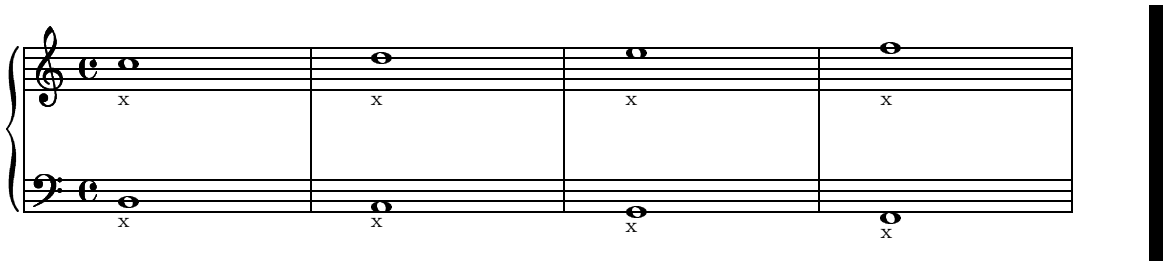
By splitting the grouping (Axis_group_engraver) and creation functionality into separate contexts, you can override interesting things. You can also drop the \consistsend feature. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/staff-container.ly’:



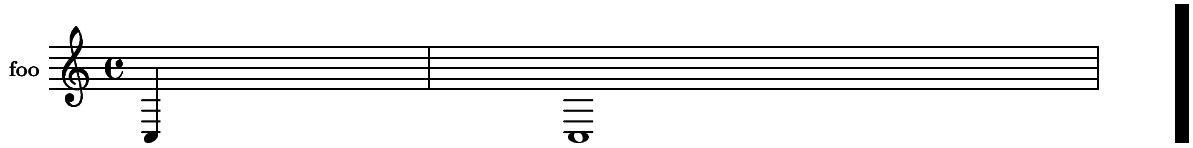
Setting staff space on a staff ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/staff-line-leadin



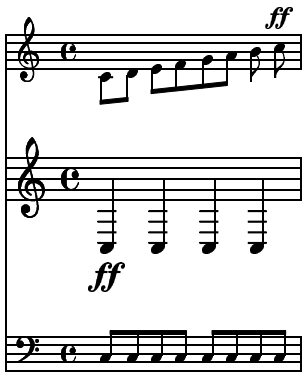
staff symbol property set workaround ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/staff-line



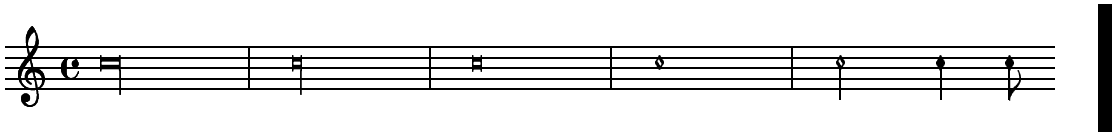
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/staff-margin-partial.ly’:



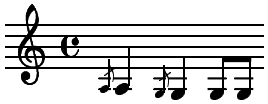
Setting staff sizes is a little clumsy. There are two options: using StaffContainer and override/revert, or \outputproperty. Both methods are shown in this example. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/staff-size.ly’:



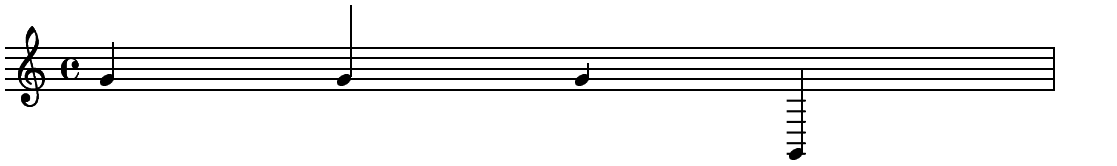
mensural note heads. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/stem-centered.ly’:



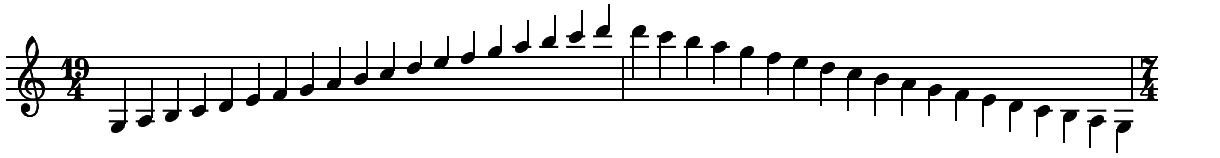
Conventionally, stems and beams extend to the middle staff line. This extension can be controlled through Voice.Stem’s grob-property no-stem-extend: ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/stem-extend.ly’:

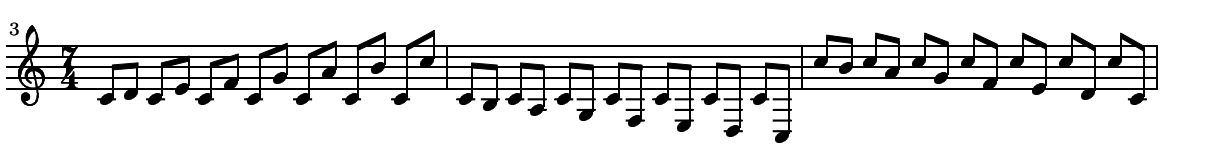


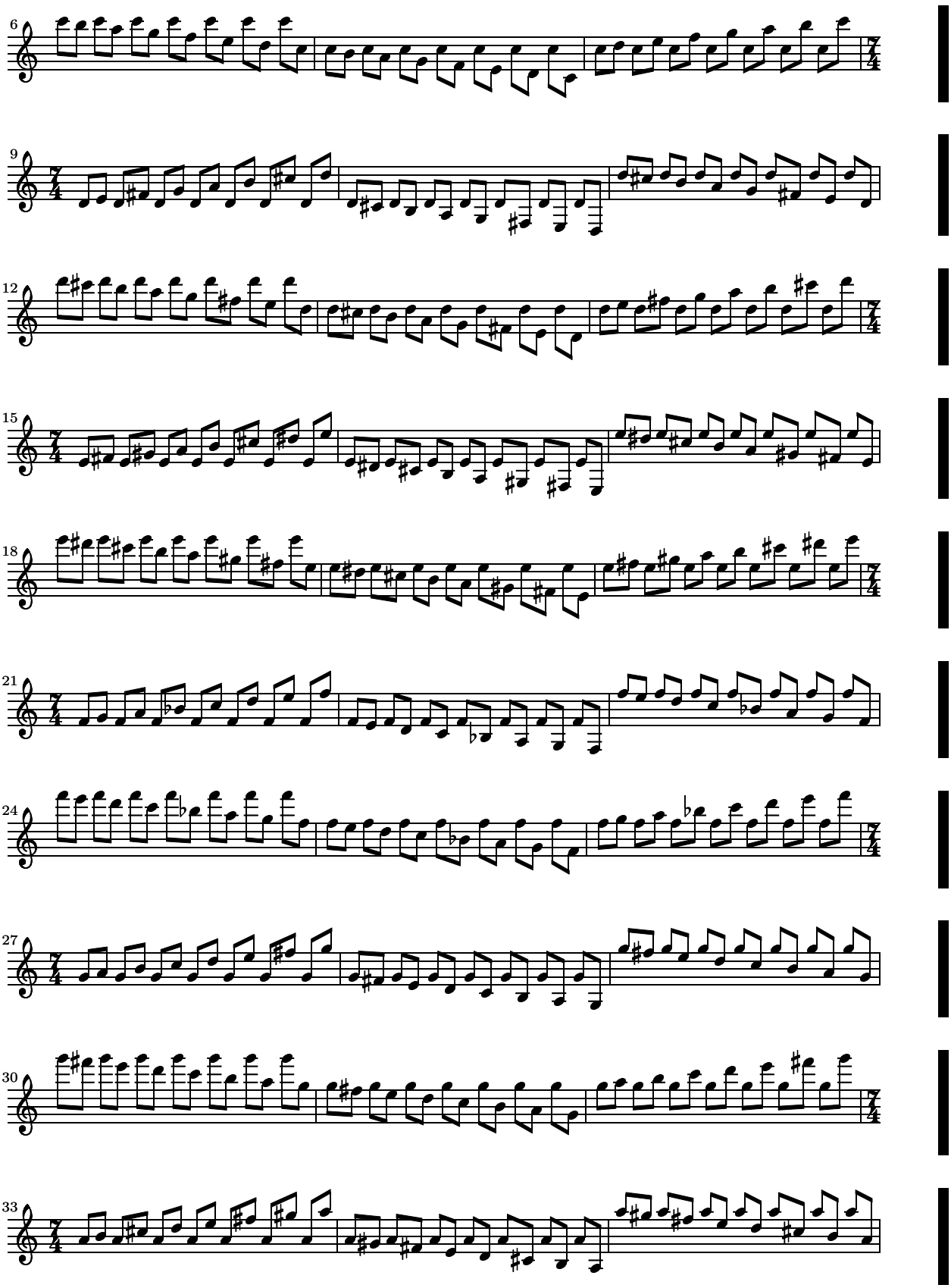
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/stem-length.ly’:

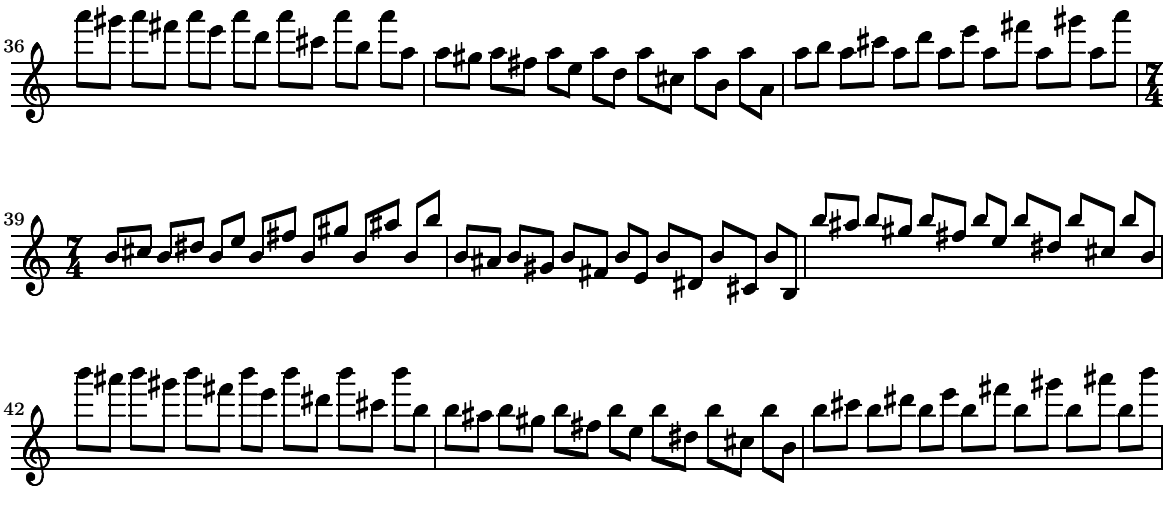


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/stem.ly’:

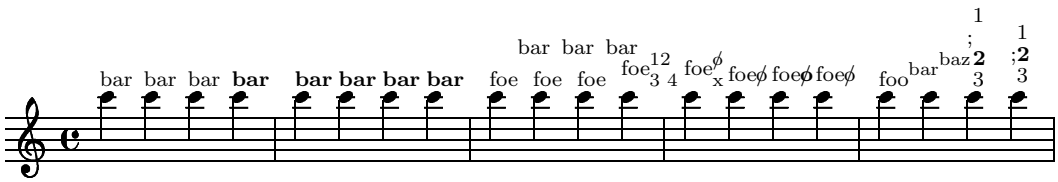




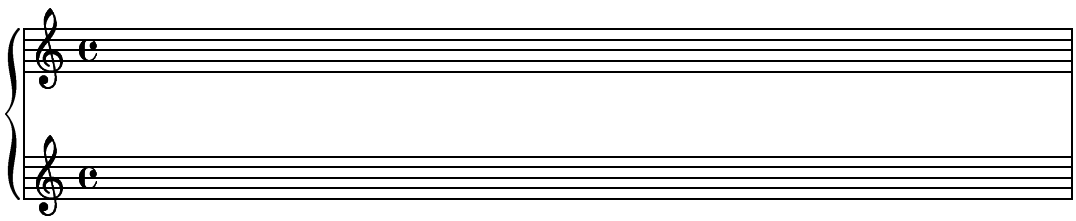




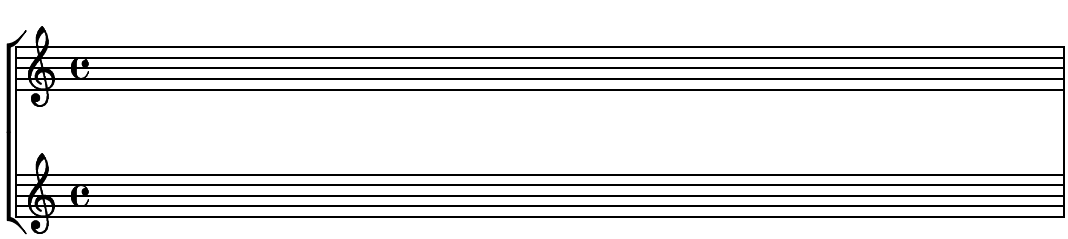
Test super/sub, raise and overstrike. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/super-sub.



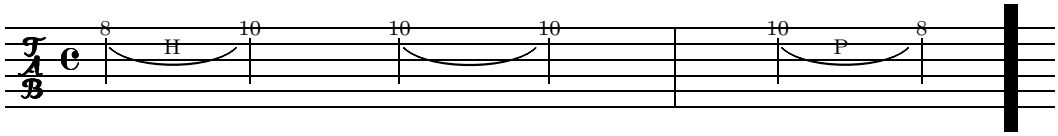
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/system-start-brace.ly’:



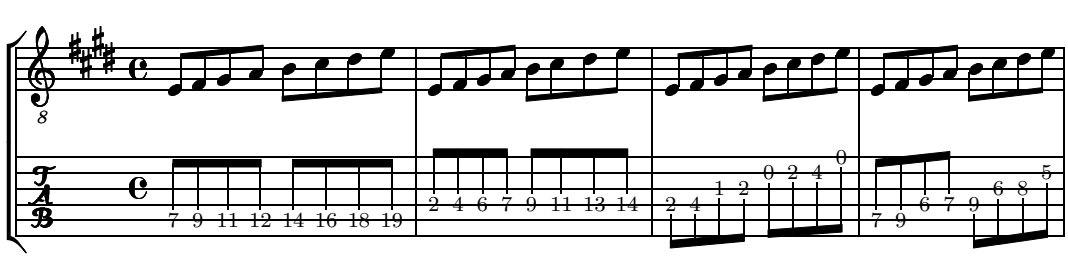
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/system-start-bracket.ly’:



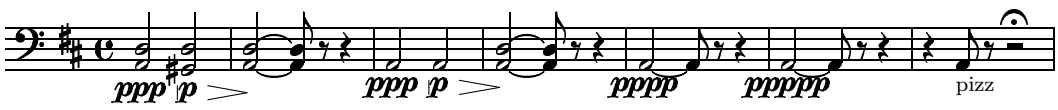
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/tablature-hammer.ly’:



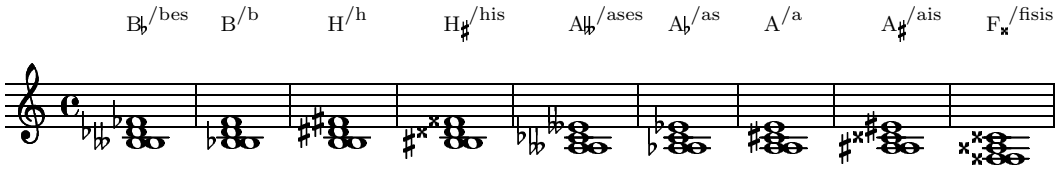
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/tablature.ly’:



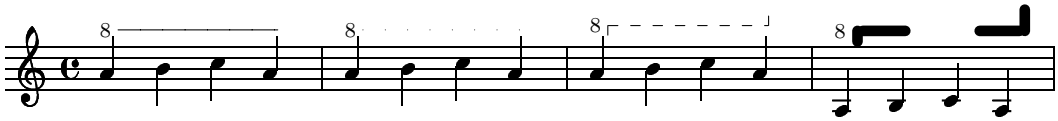
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/tchaikovsky.ly’:



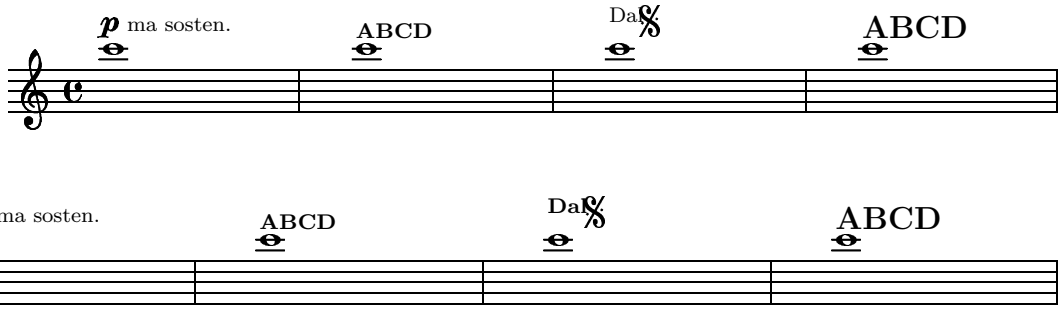
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/test-german-chords.ly’:



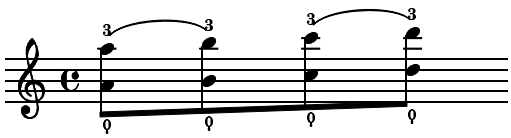
ottava spanners. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/text-spanner.ly’:



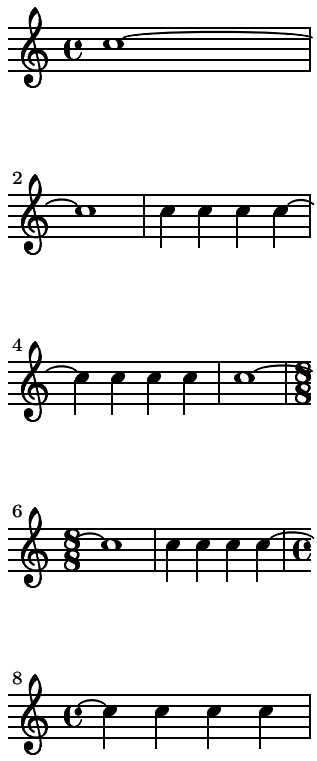
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/textscript.ly’:



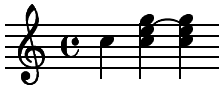
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/thumb.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/tie-broken.ly’:

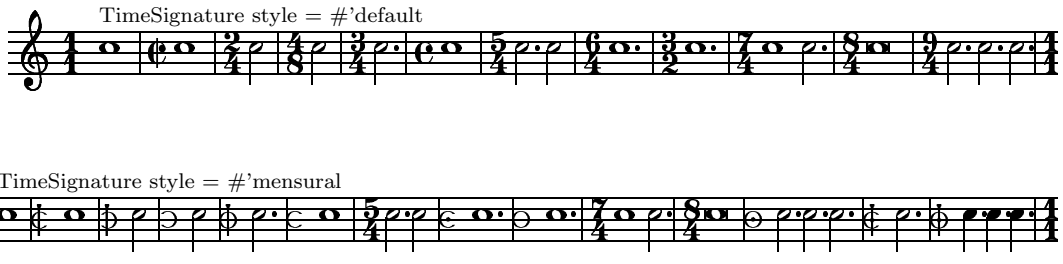


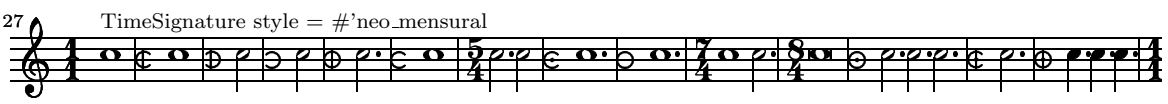
setting sparseTies causes only one tie to be generated per chord pair. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/tie-sparse.ly’:

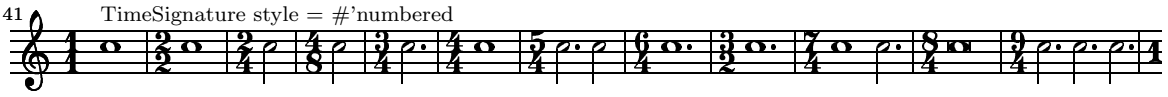


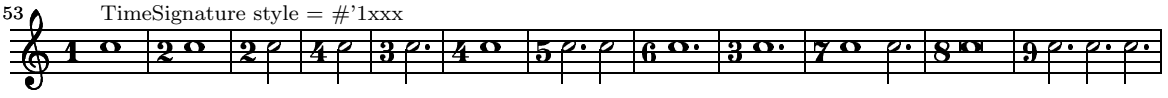
IMPORTANT NOTE: The current selection scheme for time signature symbols is not flexible enough for future extensions such as various flavours of early mensural notation or complex signatures as in contemporary music. Therefore, the semantics of time-signature properties will quite definitely change, and maybe the syntax of the ime request will possibly be extended. See the input file for TODOs.

‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/time.ly’:

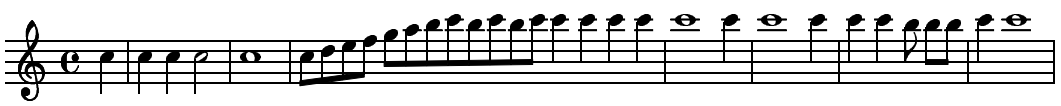




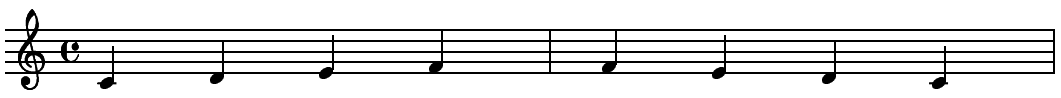


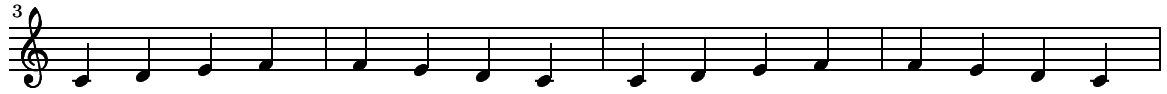


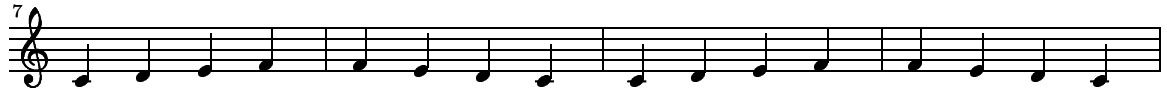
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/timing.ly’:

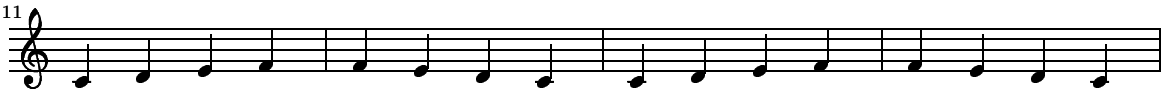


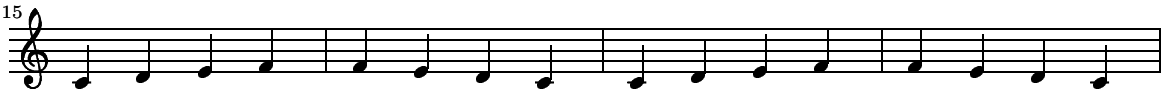
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/title.ly’:

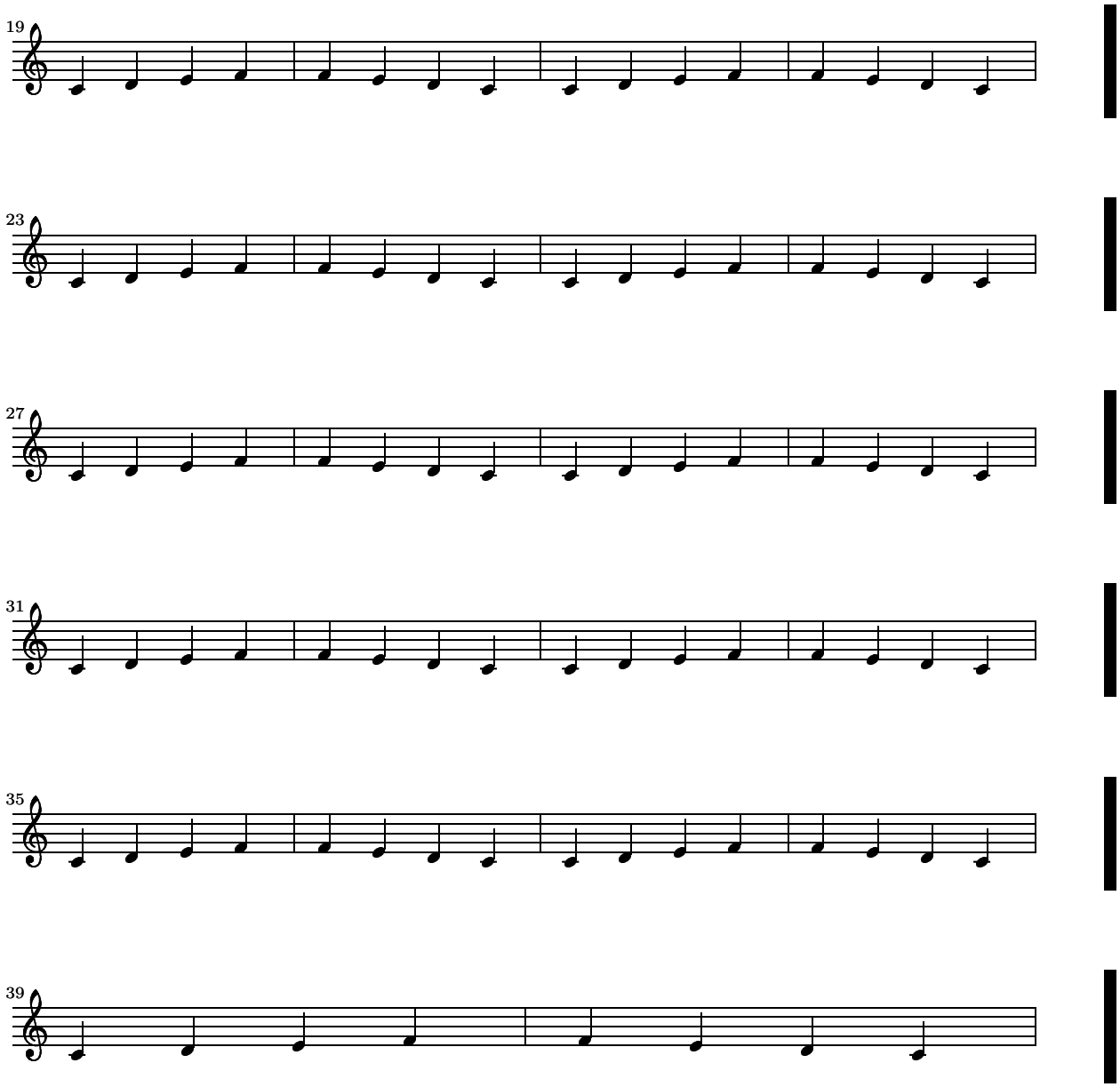




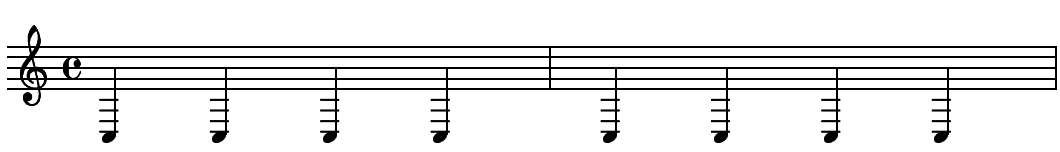




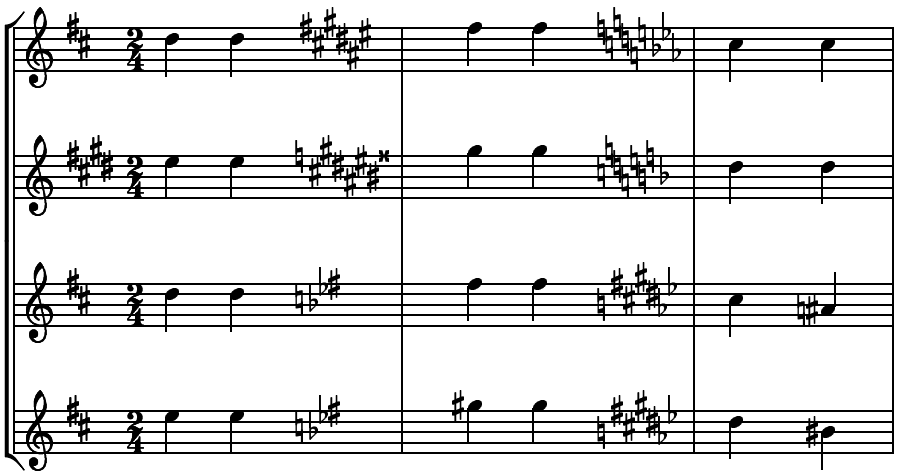




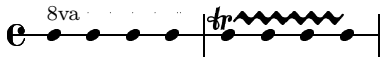
the transposing property leaves output invariant, but has effect on MIDI. ‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/transposing.ly`’:



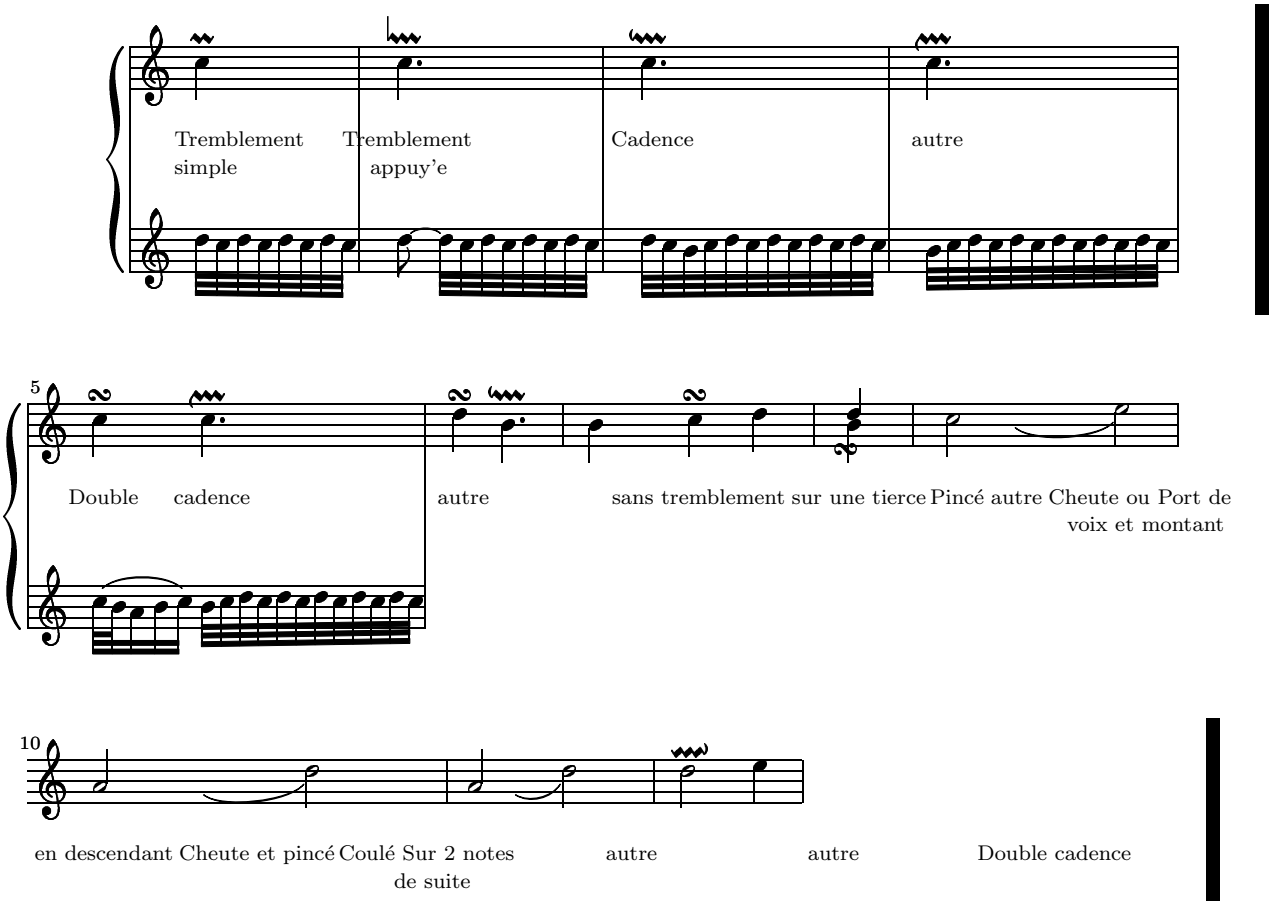
‘`/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/transposition.ly`’:



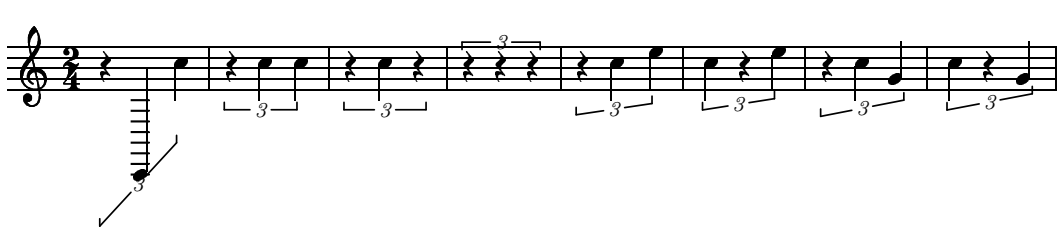
show trill line type ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/trill.ly’:



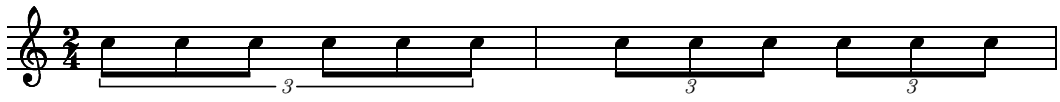
Document trills, pralls and turns ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/trills.ly’:



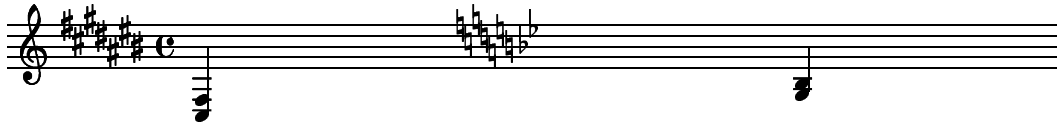
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/tuplet-rest.ly’:



‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/tuplet-spanner-duration.ly’:

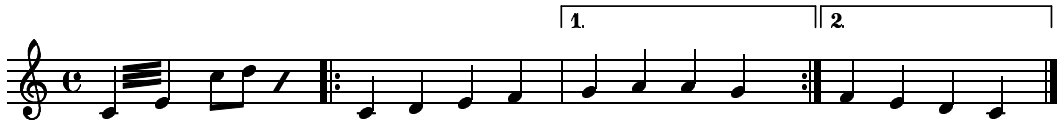


If you specify two different key sigs at one point, a warning is printed. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/two-key.ly’:

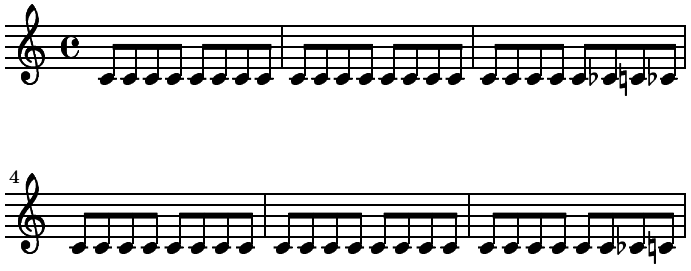


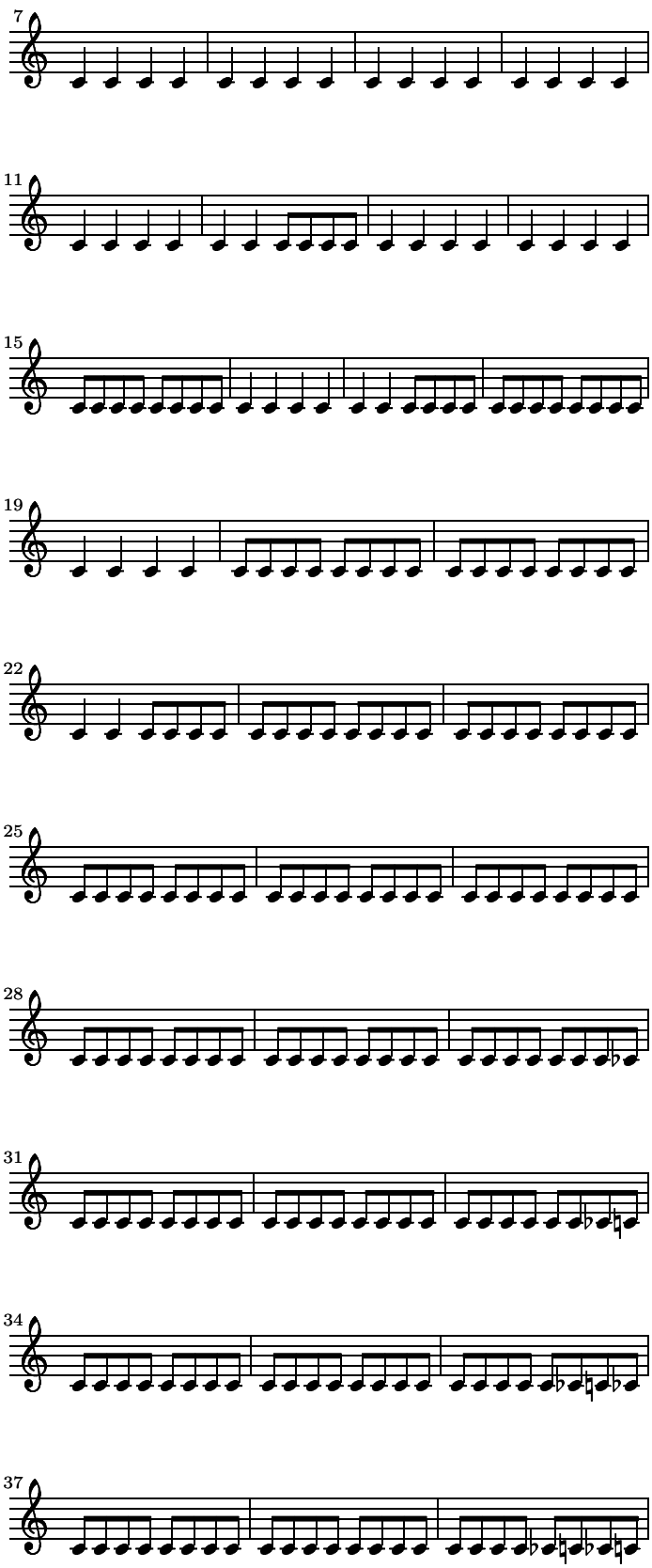
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/two-slurs.ly’:

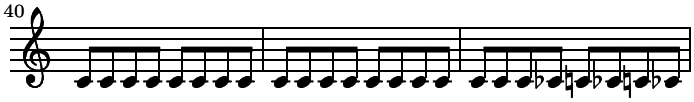
The standard function unfold-repeats will recursively unfold all repeats for correct MIDI output. Thanks to Rune Zedeler. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/unfold-all-repe

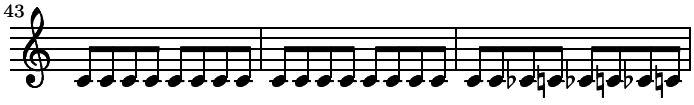


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/uniform-breaking.ly’:

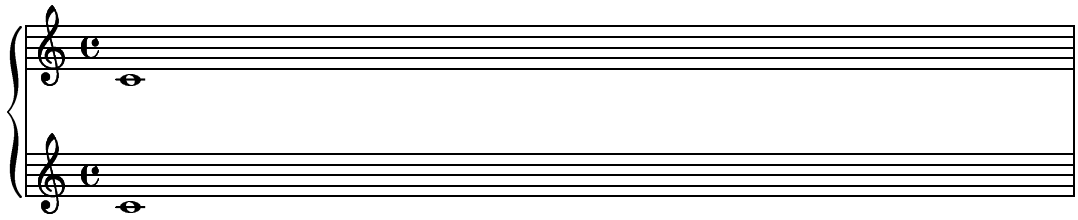


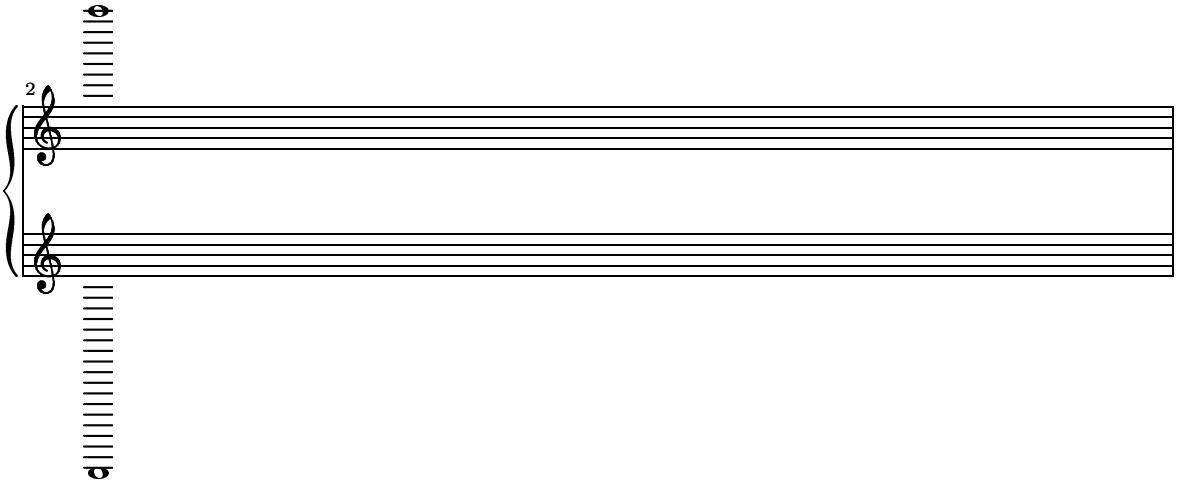

7
11
15
19
22
25
28
31
34
37





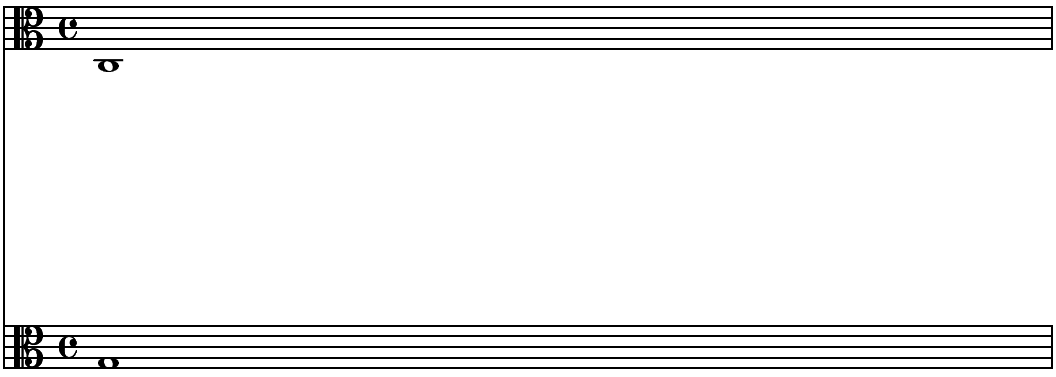
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/vertical-align.ly’:

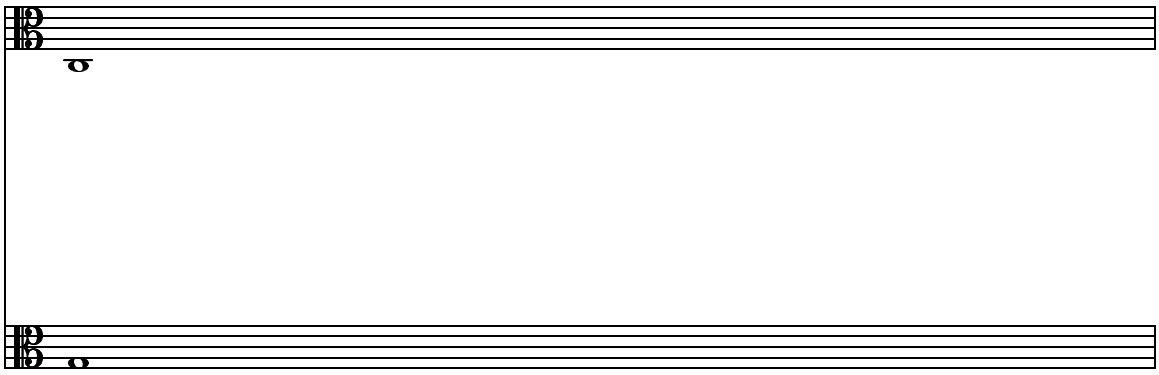




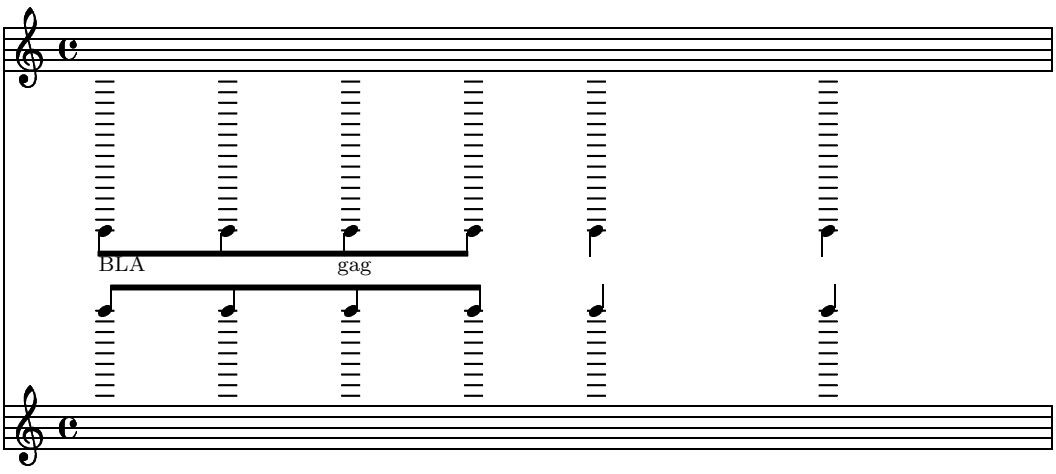
vertical extents may be overriden by verticalExtent, verticalExtent, verticalExtent. These are normal property values, and are written itno the grob when the associated context finishes, so using it in \property works.

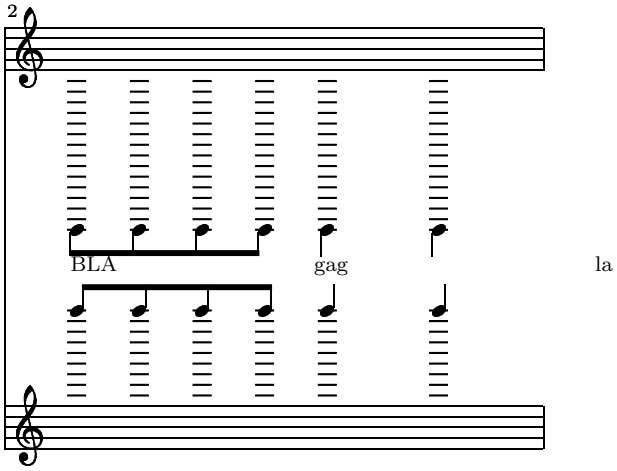
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/vertical-extent.ly’:



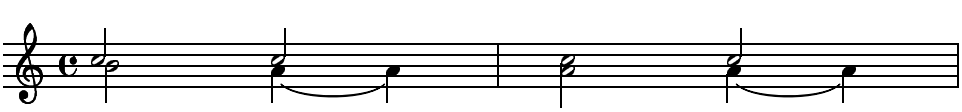


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/vertical-text.ly’:

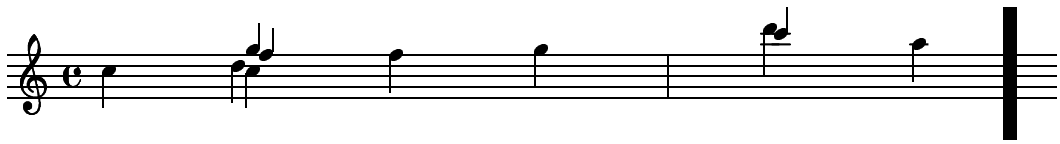




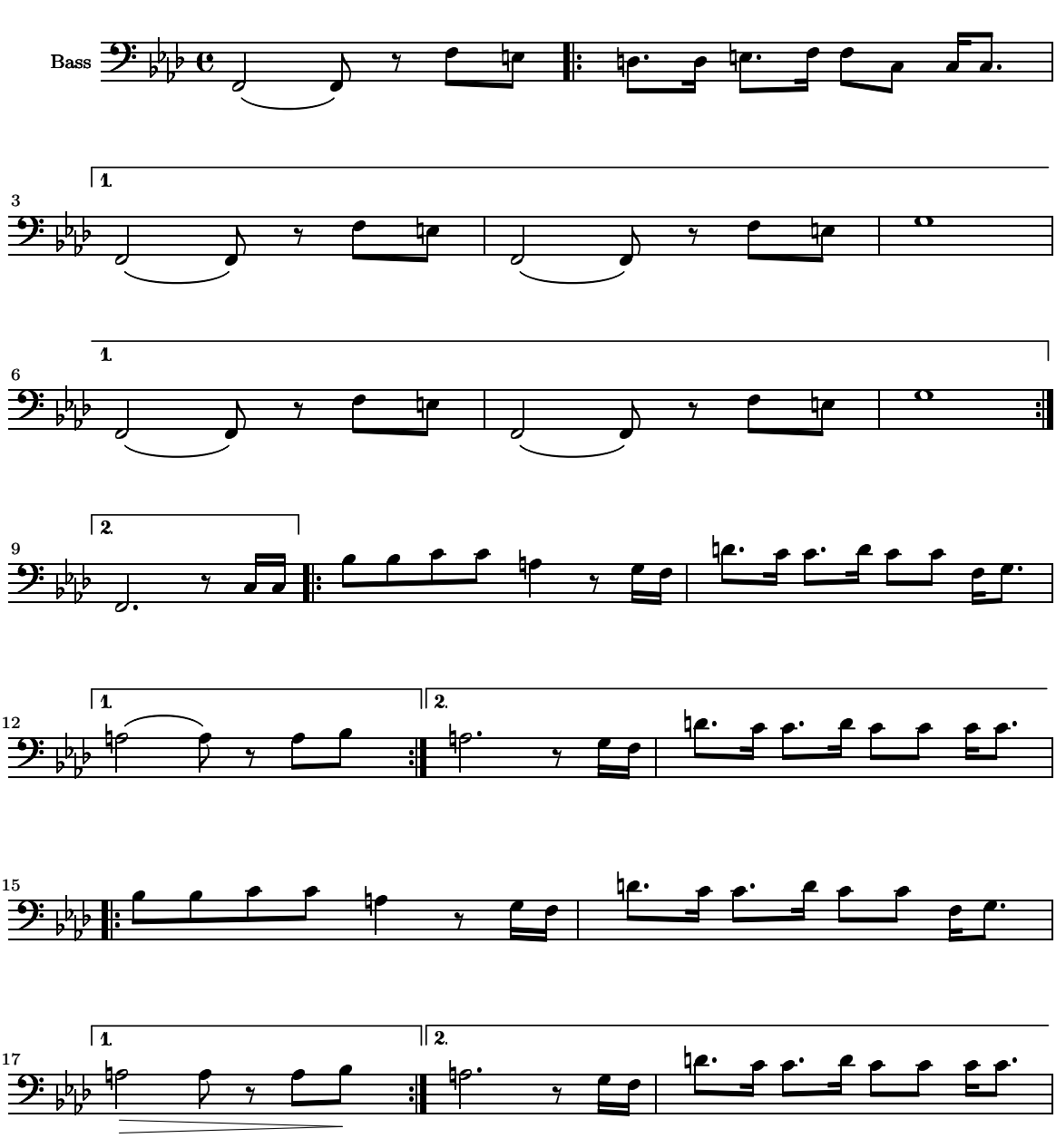
‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/voice-switch.ly’:

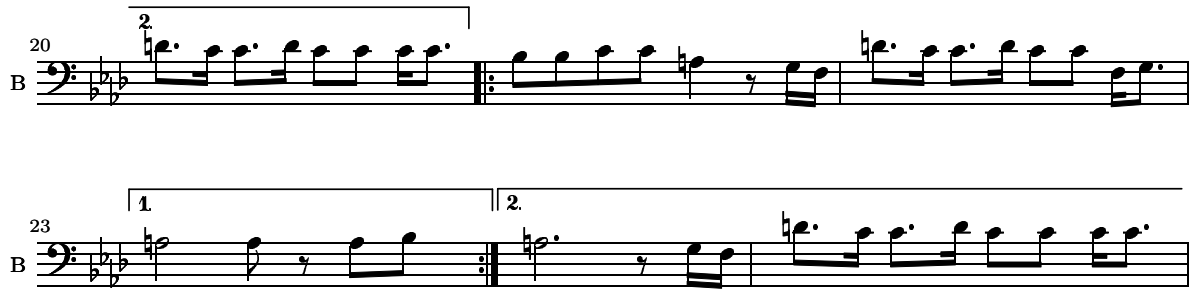
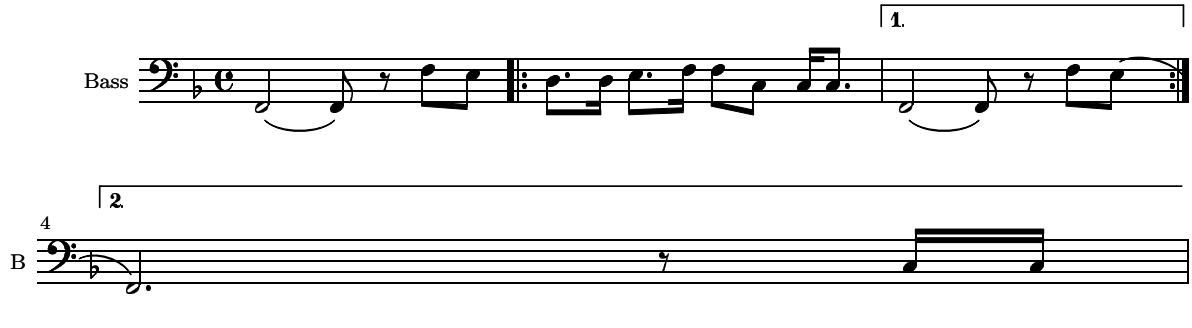


‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/voicify-chords.ly’:



Volta braces are hung on barlines. At the start of the line, they should always start after the prefatory matter. ‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/volta-start.ly’:





‘/home/inger/RPM/BUILD/lilypond-1.6.6/input/test/volta.ly’: