NAGYABOLDOGASSZONY NAPI kánon, 1. hang

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Tropár, 2. hang</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 19.54))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major a'2 f'4( g') a'2 a'4 bes' a' g' a' bes' a' g' f'2 g'4 g' a'2 \break \divisioMaxima a'4 bes' c''2( bes') a'4( bes') c''2 bes'2 a'4 g' f' g' f'( e') d'2 \break \divisioMaxima a'4 bes' c'' bes' a' bes' a'( g') f'( g') f' e' d'2 \divisioMaxima c''2 d'' c'' bes' c''4 c'' bes'2 a' \break c''2 d''4 d'' d''2 d'' c''4 bes' c''2( bes') a'2 \divisioMaxima \divisioMaxima a'4( bes') c'' d''2 c''4 bes' a' g'2 \break \divisioMaxima c''2 bes' a'4 bes' a'2 g' bes'4 a' bes'( c'') a'2 g'4( fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"1." "Is" -- "ten" -- "től" "föl" -- "é" -- "ke" -- "sí" -- "tett," "leg" -- "szen" -- "tebb" "Szűz" "Má" -- "ri" -- "a," "a" "te" "el" -- "huny" -- "tod" "em" -- "lé" -- "ke" "egy" -- "be" -- "gyűj" -- "ti" "dí" -- "csé" -- "re" -- "ted" -- "re" "a" "hí" "vők" "se" -- "re" -- "gét," "kik" "é" -- "nek" -- "kel" "ör" -- "ven" -- "dez" -- "ve" "tánc" -- "kar" -- "ral" "e" -- "gyütt" "é" -- "ne" -- "kel" -- "ték" "a" "te" "egy" -- "szü" -- "löt" -- "ted" -- "nek:" "mert" "ő" "di" -- "cső" -- "ség" -- "gel" "meg" -- "di" -- "cső" -- "ít" -- "te" -- "tett." }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.5       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.5       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>1. Is- ten- től föl- é- ke- sí- tett, leg- szen- tebb Szűz Má- ri- a, m2     d4 r m2  m4   f  m   r   m     f    m     r    d2   r4  r   m2 / a  te el-  huny- tod em- lé- ke egy- be- gyűj- ti m4 f  s2 f m4 f  s2  f2  m4  r  d    r   d t,  l,2 / dí- csé- re- ted- re a hí  vők se- re- gét, m4  f    s   f    m  f m r d r d   t,  l,2   kik é- nek- kel  ör- ven- dez- ve  tánc- kar- ral e- gyütt é- ne- kel- ték s2  l  s    f    s4  s    f2   m / s2    l4   l   l2 l     s4 f   s2 f m2   a    te egy- szü- löt- ted- nek: m4 f s  l2   s4   f    m    r2   / mert ő di- cső- ség- gel meg- di- cső- ít- te-   tett. s2   f m4  f    m2   r   f4   m   f s  m2  r4 di r2   </EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.5</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Tropár, 2. hang1</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 72.55))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major a'\breve \unHideNotes f'4 f' g'2 a' \divisioMaxima a'\breve \hideNotes a' a' \unHideNotes g'4 f'2 g' a' \divisioMaxima a'\breve \unHideNotes bes'4 c'' bes' a' bes' a' g' f'\breve \unHideNotes f'4 g' f' e' d'2 \divisioMaxima c''2 d''4 c'' bes'2 c''2( bes') a' \break \divisioMaxima c''4 bes' a'2 g' bes'4( a' bes' c'') a'2( g'4 fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"3." \alignAt #-0.786 "Mindeneknek" "Al" -- "ko" -- "tó" -- "ja," \alignAt #-1.273 "isteni" \alignAt #-0.854 "bölcsesség" \alignAt #0 "és" "e" -- "rő," "Krisz" -- "tus," \alignAt #-0.902 "változhatatlan" "és" "meg" "nem" "in" -- "gat" -- "ha" -- "tó" \alignAt #-0.73 "Szentegyházat" "a" -- "la" -- "pí" -- "tot" -- "tál," "ki" "e" -- "gye" -- "dül" "szent" "vagy" "és" "a" "szen" -- "tek" -- ".ben" "nyug" -- "szol." }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.5       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.5       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>3. Mindeneknek Al- ko- tó- ja, m0             d4  d   r2  m   isteni bölcsesség és e- rő, Krisz- tus, m0                   r4 d2  r      m     változhatatlan és meg nem in- gat- ha- tó Szentegyházat a- la- pí- tot- tál, m0             f4 s   f   m   f    m   r  d0            d4 r   d   t,   l,2   ki e- gye- dül szent vagy s2 l4 s    f2  s2 f  m    / és a szen- tek-.ben      nyug-     szol. s4 f m2    r    f4 m f s  m2 r4 di r2   </EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.5</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Tropár, 2. hang2</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 117.62))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major a'4 bes' a' g' f' f' g' g' a'2 a'2 bes'4 a'\breve \unHideNotes g'4 f'2 g' a' \divisioMaxima a'4 bes' c'' bes' a' bes' a' g' f'\breve \unHideNotes g'4 f' e' d'2 \break \divisioMaxima d''2 d'' c'' bes' c''4 c'' bes' a'2 d''4 d'' d''2 c'' bes' c'' bes' a'4 bes' c'' c'' bes'2 a' \break \divisioMaxima a'4 bes' c'' d''2 c''4 bes' a' g'2 a'4 bes' c'' d''2 c''4( bes' a') g'2 \divisioMaxima c''4 bes' a'2 bes'4 a' g' g'2 g' \break bes'4( a') bes'( c'') a'2 g'4( fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"4." "A" "pró" -- "fé" -- "ták" "hir" -- "de" -- "té" -- "se" -- "i" "és" "jö" -- \alignAt #-0.826 "vendölé" -- "se" -- "i," "Krisz" -- "tus," "ki" -- "je" -- "len" -- "ték" "Szűz" -- "től" "va" -- "ló" \alignAt #-0.681 "megteste" -- "sü" -- "lé" -- "se" -- "det," "Fé" -- "nyes" -- "sé" -- "ged" "su" -- "ga" -- "ra" "i" "vi" -- "lá" -- "gos" -- "sá" -- "gul" "szol" -- "gál" -- "nak" "a" "nem" -- "ze" -- "tek" -- "nek," "s a" "mély" -- "sé" -- "gek" "ör" -- "ven" -- "dez" -- "ve" "han" -- "goz" -- "tat" -- "ják" "né" -- "ked:" "Di" -- "cső" -- "ség" "a" "te" "e" -- "rőd" -- "nek," "Em" -- "ber" -- "sze" -- "re" -- "tő!" }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.5       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.5       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>4. A pró- fé- ták hir- de- té- se- i  és jö- vendölé- se- i, Krisz- tus, m4   f    m   r   d    d   r   r   m2 m2 f4  m0       r   d2 r      m     ki- je- len- ték Szűz- től va- ló megteste- sü- lé- se- det, m4  f   s    f   m     f   m   r  d0        r4  d   t,  l,2 / Fé- nyes- sé- ged su- ga - ra i  vi- lá- gos- sá- gul szol- gál- nak a nem- ze- tek- nek, l2  l     s   f   s4  s    f  m2 l4  l   l2   s   f   s     f    m4  f s    s   f2   m    / s  a mély- sé- gek ör- ven- dez- ve  han- goz- tat- ják né-     ked: m4   f     s   l2  s4  f    m    r2  m4   f    s    l2  s4 f m  r2    Di- cső- ség a  te e- rőd- nek,   Em-  ber- sze- re-   tő! s4  f    m2  f4 m  r  r2   r    / f4 m f s  m2   r4 di r2 </EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.5</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Tropár, 2. hang3</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 166.11))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major f'4 g' a'\breve \hideNotes a' a' a' \unHideNotes bes'4 a' g' f'2 g' a' \divisioMaxima \divisioMaxima a'4( bes') a'( g') f'2 g' a'( bes'4 a' g'2) \divisioMaxima g'2 a'4( bes') c'' bes' a'2 bes'4 a' g'2 g'4 g' \break bes'4 a' bes'( c'') a'2( g'4 fis') g'2 \divisioMaxima f'4 g' a'\breve \unHideNotes a'4 bes' c'' bes' a' bes' a' g' f' g' f' e' d'2 \divisioMaxima a'4 bes' c''2 c'' d''\breve \hideNotes d'' \break \unHideNotes d''\breve \unHideNotes c''4 bes' c''2( bes') a' \divisioMaxima d''2 c''4 bes' c'' bes' a'2 \divisioMaxima bes'2 c''4 bes' a'( bes') a'2 g' bes'4 a' bes' c'' a'2( g'4 fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"5." "A" "te" \alignAt #-1.124 "erényeid" \alignAt #-1.273 "isteni" \alignAt #0 "és" \alignAt #-0.851 "kifejez" -- "he" -- "tet" -- "len" "jó" -- "sá" -- "gát" "hir" -- "det" -- "jük," "Krisz" -- "tus," "mert" "az" "A" -- "tyá" -- "nak" "ö" -- "rök" -- "ké" -- "va" -- "ló" "di" -- "cső" -- "sé" -- "gé" -- "ből" "a" "te" \alignAt #-0.785 "személyedben" "e" -- "gyüt" -- "te" -- "sen" "ö" -- "rök" "vi" -- "lá" -- "gos" -- "ság" "la" -- "koz" -- "ván," "ki" "a" "Szűz" -- "nek" \alignAt #-0.687 "méhében" \alignAt #-0.929 "történt" \alignAt #-0.745 "megtestesü" -- "lé" -- "sed" "ál" -- "tal" "Nap" "gya" -- "nánt" "tün" -- "dö" -- "kölsz" "a" "sö" -- "tét" -- "ség" -- "ben" "és" "ár" -- "nyék" -- "ban" "ü" -- "lők" -- "nek." }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.5       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.5       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>5. A te erényeid isteni és kifejez- he- tet- len jó- sá- gát d4   r  m0                          f4  m    r   d2  r   m    hir- det- jük, Krisz- tus,      m4 f m r  d2   r      m f4 m r2  mert az   A- tyá- nak ö- rök- ké- va- ló   di- cső- sé- gé-      ből r2   m4 f s  f    m2  f4 m    r2  r4  r  / f4  m    f s m2 r4 di r2   a te személyedben e- gyüt- te-sen ö- rök vi- lá- gos- ság la- koz-ván, d4 r  m0          m4 f     s   f  m  f   m   r   d    r   d   t,   l,2   ki a Szűz- nek méhében történt   megtestesü- lé- sed ál-  tal m4 f s2    s   l0              / l0          s4  f   s2 f m    Nap gya- nánt tün- dö- kölsz l2  s4   f    s    f   m2     a  sö- tét- ség- ben és ár- nyék- ban ü- lők-     nek. f2 s4  f    m f  m2  r  f4  m     f   s  m2 r4 di r2  </EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.5</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Tropár, 2. hang4</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 0.06))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major a'\breve \hideNotes a' a' a' a' a' \unHideNotes f'4 f' g'2 a' \divisioMaxima a'\breve \hideNotes a' a' a' \unHideNotes bes'4 a' g' f' f' g'2 a' \break \divisioMaxima \divisioMaxima a'4( bes') c'' bes' a' bes' a'( g') f' g' f' e' d'2 \divisioMaxima c''2 d'' d'' c'' bes' c''\breve \unHideNotes bes'4 a'2 \divisioMaxima d''2 d'' c''4 bes' c'' c'' bes' a'2 \break \divisioMaxima c''4 bes' a' bes' a' g'2 g' g'\breve \unHideNotes g'4 bes'( a') bes'( c'') a'2( g'4 fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"6." \alignAt #0 "A" \alignAt #-0.946 "tenger" \alignAt #-0.847 "habjaiból" \alignAt #0 "a" \alignAt #-0.805 "cethal" \alignAt #-0.651 "gyomrában" "e" -- "re" -- "dő" "tűz" \alignAt #0 "a" \alignAt #0 "te" \alignAt #-0.867 "háromnapi" \alignAt #-1.144 "eltemet" -- "te" -- "té" -- "sed" "e" -- "lő" -- "ké" -- "pe," "ki" "meg" -- "sza" -- "ba" -- "dul" -- "ván," "mi" -- "képp" "küld" -- "ve" "volt." "mert" "ő" "ép" -- "ség" -- "ben" \alignAt #-0.709 "megmene" -- "kül" -- "vén," "sér" -- "tet" -- "le" -- "nül" "han" -- "goz" -- "ta" -- "tá:" "A" "di" -- "csé" -- "ret" "sza" -- "vá" -- "val" \alignAt #-1.135 "áldozok" "te" -- "né" -- "ked," "U–" -- "ram!" }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.5       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.5       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>6. A tenger habjaiból a cethal gyomrában e- re- dő tűz m0                                       d4 d   r2 m    a  te háromnapi   eltemet- te- té- sed e- lő- ké- pe, m0                         f4  m   r   d  d   r2  m   / ki   meg- sza- ba- dul-  ván, mi- képp küld- ve volt. m4 f s    f    m   f     m r  d   r    d     t, l,2    mert ő ép- ség- ben megmene- kül- vén,  s2   l l   s    f   s0       f4   m2     sér- tet- le- nül han- goz- ta- tá: l2   l    s4  f   s    s    f   m2  / A  di- csé- ret sza- vá- val áldozok te- né- ked, U--      ram! s4 f   m    f   m    r2  r   r0      r4  f m f s  m2 r4 di r2  </EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.5</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Tropár, 2. hang5</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 37.06))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major a'\breve \hideNotes a' a' \unHideNotes f'2 g' a' \divisioMaxima a'4 bes' c'' bes' a'( bes') a'( g') f'2 f'4 g' f' e' d'2 \divisioMaxima c''2 d''4 d'' d''2 c''4 bes' c''2( bes') a' \divisioMaxima c''2 d''\breve \unHideNotes c''4 bes' c'' c'' bes'2 a'2 \break \divisioMaxima a'4 bes' c'' bes' a'( bes') a' g' f'\breve \hideNotes f' f' \unHideNotes g'4 a' a' bes'( a') g' a'2 \divisioMaxima g'4 a'\breve \hideNotes a' a' \unHideNotes bes'4 c'' d'' c''( bes') a' g'2 \divisioMaxima c''2 bes' a' bes' a'4 g' g' g'\breve \unHideNotes bes'4 a' bes' c'' a'2 g'4( fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"7." \alignAt #0 "A" \alignAt #-0.832 "csillapíthatatlan" \alignAt #-0.844 "haragnak" "és" "tűz" -- "nek" "el" -- "le" -- "ne" "sze" -- "gült" "az" "is" -- "te" -- "ni" "a" -- "ka" -- "rat," "a" "tü" -- "zet" "har" -- "mat" -- "tal" "olt" -- "va," "a" \alignAt #-0.809 "haragot" "pe" -- "dig" "ki" -- "gú" -- "nyol" -- "va," "hol" "a" "szent" "if" -- "jak" "is" -- "te" -- \alignAt #0 "ni" \alignAt #-1.19 "ihletésű" \alignAt #-1.199 "okos" "és" "hár" -- "mas" "szó" -- "csö" -- "ve" "a" \alignAt #-0.95 "lángok" \alignAt #-0.75 "között" \alignAt #0 "é" -- "ne" -- "kel" -- "ve" "hir" -- "de" -- "té:" "Ál" -- "dott" "vagy" "te," "a" "mi" "és" \alignAt #-1.097 "atyáinknak" "di" -- "cső" -- "í" -- "tett" "Is" -- "te" -- "ne!" }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.5       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.5       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>7. A    csillapíthatatlan haragnak és tűz- nek m0                                 d2 r    m    el- le- ne sze- gült az  is- te- ni a- ka- rat, m4  f   s  f    m f  m r d2  d4  r  d  t,  l,2   a  tü- zet har- mat- tal olt- va,  s2 l4  l   l2   s4   f   s2 f m     a  haragot pe- dig ki- gú- nyol- va, s2 l0      s4  f   s   s   f2    m2  / hol a szent if- jak is- te- ni ihletésű okos és hár- mas szó- csö- ve m4  f s     f   m f m   r   d0               r4 m    m   f m  r    m2  a  lángok között é- ne- kel- ve hir- de- té: r4 m0               f4  s    l  s f  m   r2   Ál- dott vagy te, a  mi és atyáinknak di- cső- í- tett Is- te-   ne! s2  f    m    f   m4 r  r  r0         f4  m    f  s    m2  r4 di r2</EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.5</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Tropár, 2. hang6</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 73.84))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major d'\breve \unHideNotes e'4 f' g'\breve \hideNotes g' g' g' g' g' \unHideNotes g'2 \divisioMaxima g'\breve \hideNotes g' \unHideNotes f'4 e'( d') e'4( f') g' a'2 g'4( fis' g' a') e'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \alignAt #-0.783 "Dicsérjük" "áld" -- "juk," \alignAt #-1.18 "imádjuk" \alignAt #-1.482 "az" \alignAt #-1.207 "urat," \alignAt #-1.109 "énekeljük" \alignAt #-1.482 "az" \alignAt #0 "U" -- "rat," \alignAt #0 "és" \alignAt #-0.62 "magasz" -- "tal" -- "juk" "őt" "mind" -- "ö" -- "rök" -- "ké!" }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.5       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.5       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>Dicsérjük áld- juk, imádjuk az urat, énekeljük az U- rat, l,0       t,4  d    r0                               r2     és magasz- tal- juk   őt    mind- ö- rök-      ké! r0         d4   t, l, t,4 d r     m2 r4 di r m t,2</EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.5</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>Konták, 7. hang</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 87.6))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major a'\breve \hideNotes a' a' a' \unHideNotes f'4( g') a'2 \divisioMaxima a'4 bes' c'' bes' a' bes' a' g' f'( g') f'( e') d'2 \divisioMaxima c''2 d''\breve \hideNotes d'' d'' \unHideNotes c''4 bes' \break c''2( bes') a' \divisioMaxima c''2 d''\breve \unHideNotes c''4 bes' c'' c'' bes' a'2 \divisioMaxima \divisioMaxima a'4( bes') c'' bes' a' bes' c''2 bes' a'4 g' f' g' f' e' d'2 \divisioMaxima \divisioMaxima f'4( g') a' a' bes'2 a'4 g' a'2 \divisioMaxima \divisioMaxima f'4( g') a'2 a'4 a' \break bes'2 a'4( g') a'( bes') c''2 bes' a' g' \divisioMaxima c''2 d''4 c''4 bes'2 c''4 c'' bes' a'2 \divisioMaxima g'4 a' bes' c'' d''2 c''4 bes' a'2 g' \divisioMaxima c''4 bes' a' g'2 g' bes'4( a') bes'( c'') a'2( g'4 fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"8." \alignAt #-1.135 "Az" \alignAt #-1.273 "isteni" \alignAt #0 "és" \alignAt #-0.796 "mindenható" "an" -- "gyal" "o" -- "lyan" "lán" -- "got" "mu" -- "ta" -- "tott" "az" "if" -- "jak" -- "nak," "mely" \alignAt #-0.795 "harmat" \alignAt #-0.52 "gyanánt" \alignAt #-0.375 "hű" -- "sí" -- "té" "ő" -- "ket," "és" \alignAt #-0.634 "megége" -- "té" "a" "go" -- "no" -- "szo" -- "kat," "s az" "é" -- "let" -- "a" -- "dó" "For" -- "rás" "pe" -- "dig" "Szü" -- "zet" "te" -- "rem" -- "tett," "ki" "meg" -- "ron" -- "tá" "a" "ha" -- "lált," "és" "é" -- "le" -- "tet" "ad" "az" "így" "é" -- "nek" -- "lők" -- "nek:" "A" "min" -- "de" -- "nek" "Te" -- "rem" -- "tő" -- "jét" "mi," "meg" -- "vál" -- "tot" -- "tak," "é" -- "ne" -- "kel" -- "jük" "és" "ma" -- "gasz" -- "tal" -- "juk" "mind" -- "ö" -- "rök" -- "ké!" }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     ragged-last = ##f     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.6       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.6       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>8. Az isteni és mindenható an-  gyal m0                         d4 r m2    o- lyan lán- got mu- ta- tott az if- jak- nak, m4 f    s    f   m   f   m    r  d r d t, l,2   mely harmat gyanánt hű- sí- té   ő-   ket, s2   l0                 s4  f  / s2 f m     és megége- té a go- no- szo- kat, s2 l0      s4 f s   s   f    m2    s__az é- let- a- dó For- rás pe- dig Szü-zet te- rem- tett, m4 f  s  f    m  f  s2   f   m4  r   d   r   d   t,   l,2    ki   meg- ron- tá a  ha- lált, d4 r m    m    f2 m4 r   m2     és   é- le- tet   ad az   így é- nek- lők- nek: d4 r m2 m4  m   / f2 m4 r m f s2 f    m    r  A  min- de- nek Te- rem- tő- jét  s2 l4   s4  f2  s4  s    f   m2    mi, meg- vál- tot- tak, é- ne- kel- jük  r4  m    f    s    l2   s4 f   m2   r     és ma- gasz- tal- juk mind- ö-  rök-     ké! s4 f   m     r2   r   f4 m  f s m2 r4 di r2 </EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.6</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()

![<?xml version="1.0" encoding="utf-16"?> <item>   <title>9. irmosz</title>   <LilyPond>\version "2.18.2"  \include "gregorian.ly"  \paper {   paper-width = 185.0143\mm   paper-height = 272.0093\mm   top-margin = 0\mm   bottom-margin = 0\mm   left-margin = 0\mm   right-margin = 2\mm   ragged-bottom = ##t   print-page-number = ##f   page-breaking = #ly:minimal-breaking   system-system-spacing =     #'((basic-distance . 12)        (minimum-distance . 12)        (padding . 0)        (stretchability . 0)) %  page-breaking-system-system-spacing = #'((padding . 2))   markup-system-spacing = #'((padding . 124.6))   top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0)) }  #(set-global-staff-size 14)  #(define currBSign 0) #(define hideStems #t)  #(define (set-breathe bsign)  (begin   (set! currBSign bsign)  (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))    (system (ly:grob-parent columnOfBSign X))       (columns (ly:grob-object system 'columns))    (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))   )   (if (eqv? columnOfBSign lastColumn)    (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))   )  )  ) )  #(define (staff-print staffgrob)  (let* ((stencil (ly:staff-symbol::print staffgrob))    (Xextent (ly:stencil-extent stencil X))    (Yextent (ly:stencil-extent stencil Y))             (X0 (car Xextent))    (X1 (cdr Xextent))    (X1sc (+ X1 0.89))    )    (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))                          (cons X0 X1sc)                          Yextent    )  ) )  #(define (staff-break-callback staffgrob)     (let*   (     (system (ly:grob-parent (ly:grob-parent staffgrob X) X))     (columns (ly:grob-object system 'columns))     (columns-count (ly:grob-array-length columns))     (goI 1)     (goJ 1)   )   (if (ly:grob? currBSign)    (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))       (lastColumn (ly:grob-array-ref columns (- columns-count 1)))      )      (if (eqv? columnOfBSign lastColumn)       (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))       (set! currBSign 0)      )      (set! currBSign 0)    )   )   (let loopI  ((i 0)      )        (if (&lt; i columns-count)       (loopI (let* ((column (ly:grob-array-ref columns i))           (columnname (grob-name column))          )          (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))           (let*            (            (elems (ly:grob-object column 'elements))            (elems-count (ly:grob-array-length elems))            )            (let loopJ ((j 0)                 )                 (if (&lt; j elems-count)                 (loopJ (let* ((grob (ly:grob-array-ref elems j))                    (grobname (grob-name grob))                   )                   (if (eqv? grobname 'NoteColumn)                    (begin                     (show-notes grob)                     (set! j elems-count)                    )                   )                   (1+ j)                  )                )               )            )            (set! i columns-count)           )          )          (1+ i)         )       )      )   )  ) )  #(define (grob-name grob)    (if (ly:grob? grob)        (assoc-ref (ly:grob-property grob 'meta) 'name)        #f))  #(define (show-flag flag)  (ly:grob-set-property! flag 'transparent hideStems) )     #(define (show-stem stem)  (let*  ((beam (ly:grob-object stem 'beam))    )    (begin     (ly:grob-set-property! stem 'transparent hideStems)     (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))    )  ) )  #(define (show-note notehead)  (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))         (beam (ly:grob-object stem 'beam))      (flag (ly:grob-object notehead 'flag))         (acci (ly:grob-object notehead 'accidental-grob))         (dots (ly:grob-object notehead 'dot)))  (begin   (ly:grob-set-property! notehead 'transparent #f)   (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))   (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))   (ly:grob-set-property! stem 'transparent hideStems)   (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))   (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))  )) )  #(define (show-notes note-column)   (begin    (let* ((elems (ly:grob-object note-column 'elements))     (elems-count (ly:grob-array-length elems))      )         (let loop ((staff-pos-sum 0)       (i 0)         )         (if (&lt; i elems-count)        (loop (let* ((grob (ly:grob-array-ref elems i))            (grobname (grob-name grob))           )           (if (eqv? grobname 'NoteHead)            (show-note grob)            (if (eqv? grobname 'Stem)             (show-stem grob)             (if (eqv? grobname 'Flag)              (show-flag grob)             )            )           )          )         (1+ i)        )       )     )   )  ) )   leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT } centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER } rightAlign = {   \once \override LyricText #'self-alignment-X = #1 } hairpinToBarline = { \override Hairpin #'to-barline = ##t  } hairpinToNote = { \override Hairpin #'to-barline = ##f  }  alignAt = #(define-music-function      (parser location xOffset)      (number?)   (display xOffset)    #{       \once \override LyricText #'self-alignment-X = $xOffset    #})  stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length } stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }  bigNotes = { \override NoteHead #'font-size = #3 } smallNotes = { \override NoteHead #'font-size = #-3 } normalNotes = { \override NoteHead #'font-size = #0 }  bN = { \once \override NoteHead #'font-size = #3 } sN = { \once \override NoteHead #'font-size = #-3 } nN = { \once \override NoteHead #'font-size = #0 }  shOn = { \override NoteColumn #'force-hshift = #0.18 } shOff = { \revert NoteColumn #'force-hshift }  midRest = { \override Voice.Rest #'staff-position = #0 } shRest = { \revert Voice.Rest #'staff-position }  modernAccidentals = #(set-accidental-style 'modern) modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary) defaultAccidentals = #(set-accidental-style 'default) pianoAccidentals = #(set-accidental-style 'piano) voiceAccidentals = #(set-accidental-style 'voice) modernVoiceAccidentals = #(set-accidental-style 'modern-voice) modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary) pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary) neoModernAccidentals = #(set-accidental-style 'neo-modern) neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary) dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic) teachingAccidentals = #(set-accidental-style 'teaching) noResetAccidentals = #(set-accidental-style 'no-reset) forgetAccidentals = #(set-accidental-style 'forget)  go = { \grace {s16 }  }    chant = { \key f \major a'2 bes' a'4( bes') a'( g') f'2 g' a' \divisioMaxima \divisioMaxima bes'4( a') bes'( c'') d''2 c''4( bes') c''2( bes' a') \divisioMaxima \divisioMaxima a'4( bes') c''( bes') a'2 bes' a'( g'4 fis') g'2 \break \divisioMaxima c''2 d'' d'' d''\breve \unHideNotes c''4 d'' c'' bes' a'2 \divisioMaxima \divisioMaxima bes'4( c'') d''2 c''4 bes' a'( bes') a'( g') bes'( a' bes' c'') a'2( g'4 fis') g'2 \break \divisioMaxima a'4 bes' c'' d''2 c''4( bes') a'( g'2) \divisioMaxima a'4 bes' c'' d''2 c''4( bes') a' g'2 \divisioMaxima c''4 bes' a'2 bes'4 a' g'2 g' g'\breve \break \unHideNotes bes'4( a') bes'( c'') a'2( g'4 fis') g'2  \hideNotes \grace {s16 } \finalis }  verba = \lyricmode { \set stanza = #"9." "Meg" -- "győ" -- "ze" -- "tik" "te" -- "ben" -- "ned," "tisz" -- "ta" -- "sá" -- "gos" "Szűz," "a" "ter" -- "mé" -- "szet" "rend" -- "je," "mert" "ná" -- "lad" \alignAt #-0.726 "szűzességet" "hir" -- "det" "a" "szü" -- "lés," "és" "é" -- "le" -- "tet" "í" -- "gér" "a" "ha" -- "lál;" "szü" -- "lés" "u" -- "tán" "mint" "Szűz," "s ha" -- "lál" "u" -- "tán" "mint" "é" -- "lő," "min" -- "den" -- "kor" "üd" -- "vö" -- "zí" -- "ted" \alignAt #-1.074 "örökségedet," "Is" -- "ten" -- "szü" -- "lő." }   \score {   \new Staff &lt;&lt;     #(set-accidental-style 'modern-cautionary)     \new Voice = "melody" \chant     \new Lyrics = "one" \lyricsto melody \verba   &gt;&gt;   \layout {     line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)     \context {       \Staff       \override BarLine  #'stencil = ##f       \remove Time_signature_engraver       \override Stem #'transparent = #hideStems       \override Beam #'transparent = #hideStems       \override Flag #'transparent = #hideStems       \override StaffSymbol #'after-line-breaking = #staff-break-callback       \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)       \override StaffSymbol #'stencil = #staff-print     }     \context {       \Voice       \override Stem #'length = #0    \override BreathingSign #'after-line-breaking = #set-breathe     }     \context {       \Score       \override  RehearsalMark #'self-alignment-X = #LEFT       \override RehearsalMark #'break-align-symbols = #'(key-signature)       \override  NonMusicalPaperColumn #'padding = #0.7       skipBars= ##t       barAlways = ##t       lyricMelismaAlignment = #CENTER     }     \context {       \Lyrics       \override LyricSpace #'minimum-distance = #0.7       \override LyricText #'font-size = #1.065       \override LyricText #'font-name = #"TimesNewRoman"       \override LyricText #'color = #(rgb-color 0 0 0)       \override LyricHyphen #'color = #(rgb-color 0 0 0)       \override LyricExtender #'color = #(rgb-color 0 0 0)     }   } }  \header { tagline = ""  title = "" } </LilyPond>   <EasyChant>9. Meg- győ- ze-  tik te- ben- ned, m2      f    m4 f m r d2  r    m     tisz- ta- sá- gos  Szűz,  f4 m  f s l2  s4 f s2 f m  a    ter- mé- szet rend-   je, m4 f s  f m2  f    m r4 di r2  / mert ná- lad szűzességet hir- det a szü- lés, s2   l   l   l0          s4   l   s f    m2    és   é- le- tet í-  gér a       ha-      lál; f4 s l2 s4  f   m f m r f m f s m2 r4 di r2   / szü- lés u- tán mint Szűz, m4   f   s  l2  s4 f m r2   s ha- lál u- tán mint é- lő, m4    f   s  l2  s4 f m  r2   min- den- kor üd- vö- zí-ted örökségedet,  Is-  ten- szü-     lő.  s4   f    m2  f4  m   r2  r  r0          / f4 m f s  m2 r4 di r2   </EasyChant>   <Clef>0</Clef>   <Key>8</Key>   <LyricsSpacing>0.7</LyricsSpacing>   <TimeOption>0</TimeOption>   <TimeText>N/A (cadenza)</TimeText>   <StaffSizeOption>2</StaffSizeOption>   <StaffSizeText>14</StaffSizeText>   <LyricsSize>1.1</LyricsSize>   <LyricsFontName>Times New Roman</LyricsFontName>   <LyricsFontBold>False</LyricsFontBold>   <LyricsFontItalic>False</LyricsFontItalic>   <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>   <LineSpacing>12</LineSpacing>   <LineThickness>4</LineThickness>   <Options>0,1,2,4,6,7,8</Options>   <pages>1</pages>   <page>1</page> </item>]()