# RANDOM PRECISION:
# SOME APPLICATIONS OF FRACTALS AND CELLULAR AUTOMATA IN MUSIC COMPOSITION

DOCUMENT

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Musical Arts in the Graduate School of The Ohio State University

By

Igor Karaca, M.A.

*****

The Ohio State University
2005

Dissertation Committee:

Professor Thomas Wells, Adviser

Professor Donald Harris

Professor Russel Mikkelson

Approved by

______________________________

Adviser
School of Music Graduate Program

## ABSTRACT

Automated composition is a form of algorithmic composition which always involves mechanical, or digital elements of decision-making, to make music with minimal human intervention. This is often done on a computer, with the aid of random number generation, rule-based systems, and various other algorithms. The idea of composing music by stochastic means, and using computers as an aid in music composition is still alive; there are many musicians and programmers such as Yo Kubota, Phil Thompson, David H. Singer, Phil Jackson and others, trying to develop software for automated composition, which is – most of the time – available as a free download from their respective websites. Some of the most popular programs, such as Gingerbread, FractMus, Texture and Fmusic use fractals, chaotic equations, and cellular automata simulations to create music.

This thesis shows how algorithmic music is generated, and – at the same time – focuses on Fmusic, program that uses both fractals and cellular automata to generate music, and is flexible enough to allow composer to have control over many important musical parameters, such as rhythm, loudness, pitch distribution, panning, patch (timbre), patterns and transitions.

Dedicated to my wife Natasa

## ACKNOWLEDGMENTS

I would like to thank my committee members for their intellectual support, encouragement and enthusiasm. They each gave of their time, and were responsive to my needs.

I wish to thank my adviser, Dr. Thomas Wells, for his guidance and support; thank you for being there for me every step of the way.

I thank Dr. Daniel Boord for stimulating discussions. Thank you for your input on this project, for your time, expertise and encouragement.

I wish to express my gratitude to my parents, sister and wife who have inspired me, and been a primary source of support. Thank you for your commitment to my success.

## VITA

26. October, 1974……….Born – Sarajevo, Bosnia and Herzegovina

1996………………………B.A. Music Composition, Music Academy Sarajevo

2000………………………M.A. Music Composition, The Ohio State University

1999 – Present…………..Graduate Teaching Associate, The Ohio State University

## FIELDS OF STUDY

Major Field: Music Composition

TABLE OF CONTENTS

Page

Abstract……………………………………………………………………………ii

Dedication…………………………………………………………………………..iii

Acknowledgments………………………………………………………..………iv

Vita…………………………………………………………………..……………v

List of Figures……………………………………………………………………vii

Introduction………………………………………………………………………..1

Chapters:

Chapter 1 – Fractals in Music..……………………………………………….....6

Chapter 2 – Artificial life and music composition: Cellular Automata………..19

Chapter 3 – Fmusic: computer software for automated music

composition…………………………………………………………………..….29

Conclusion……………………………………………………..……………..41

Bibliography………………………………………………………………………47

Appendix A she turned from Ares her bright eyes – Movement 1……..……49

Appendix B she turned from Ares her bright eyes – Movement 2……………63

Appendix C she turned from Ares her bright eyes – Movement 3..…………..88

Appendix D she turned from Ares her bright eyes – Movement 4……..…..102

Appendix E “Sarajevo Marlboro” excerpts …………………………..……….126

Appendix F Random Poetry ………………..…………………….....……….131

LIST OF FIGURES

Figure Page

1.1 Self-similarity, trapezoid……………………………………………………7

1.2 Self-similarity, L-shape…………………………………………………..8

1.3 Sierpinski's Triangle……………………………………………………..8

1.4 Koch Curve………………………………………………………….....9

1.5 Koch Snowflake…………………………………………………………9

1.6 Fractal Fern…………………………………………………………….17

2.1 "Birth"……………………………………………………………………..21

2.2 "Survival"………………………………………………………………….21

2.3 "Death"…………………………………………………………………….21

2.4 Simple CA rules…………………………………………………………24

3.1 Event Parameters……………………………………………………….31

3.2 CA and Misc. option page………………………………………………35

3.3 Fractal Notes page………………………………………………………37

3.4 Percussion and Bass settings page……………………………………39

# INTRODUCTION

Automated composition is a form of algorithmic composition which always involves mechanical or digital elements of decision-making, to make music with minimal human intervention. This is often done on a computer, with the aid of random number generation, rule-based systems, and various other *algorithms*.

The simplest form of automated composition involves using random numbers to piece together melodic fragments. One very early example of this is the *Musikalisches Wurfelspiel* (Dice Music) attributed to Wolfgang Amadeus Mozart. The Dice Music game involved assembling a number of small musical fragments, and combining them by chance, piecing together a new piece from randomly chosen parts.

Chance has also played a part in the compositions of more modern composers. American composer John Cage, for example, composed his *Book of Changes* for piano with the help of the old Chinese oracle book *I-Ching.* In his own words: "*I-Ching told me to continue what I was doing, and to spread joy and revolution."*

Cage has also used natural phenomena to compose music (his Atlas Eclipticalis, 1961, composed from astronomical charts), as has Charles Dodge, with his piece entitled *The Earth's magnetic Field* (1970), for which a computer translated fluctuations in the magnetic field of the Earth into music. Lejaren Hiller (with the help of Leonard Isaacson and Robert Baker) was one of the first to successfully utilize a computer to compose music. His *Computer Cantata* (1963) was one of the very first compositions to be created with digital assistance. At a time when all computers were slow, bulky mainframes, and software had to be created from scratch using awkward computer languages designed for the physical sciences, Hiller and his assistant Robert Baker created a musical tour-de-force in which a computer not only selected all the notes, but also created the singer's text and generated the sounds for an electronic tape.

Some other composers, like Iannis Xenakis, used computers to aid in the composition of scores for live ensembles, using statistical and probabilistic methods. His "stochastic music program" would "deduce" a score from a list of note densities and probabilistic weights supplied by the programmer, leaving specific decisions to a random number generator. *Morsima-Amorsima*, for 4 instruments, was composed in this manner. This way of composing automated music is usually more flexible, and allows greater degree of human intervention than methods of Cage or Hiller.
Xenakis' work was strongly influenced by the aesthetics of mathematics, and also by earlier works in serial polyphony, which he sometimes criticized: *“Linear polyphony destroys itself by its very complexity; what one hears is in reality nothing but a mass of notes in various registers. The enormous complexity prevents the audience from following the intertwining of the lines and has as its macroscopic effect an irrational and fortuitous dispersion of sounds over the whole extent of the sonic spectrum.”*

The idea of composing music by stochastic means, and using computers as an aid in music composition is still alive; there are many musicians and programmers such as Yo Kubota, Phil Thompson, David H. Singer, Phil Jackson and others, trying to develop software for automated composition, which is – most of the time – available as a free download from their respective websites. Some of the most popular programs, such as Gingerbread, FractMus, Texture and FMusic use fractals, chaotic equations, and cellular automata simulations to create music. More importantly, the system requirements for these programs are very low. Everything we need to start composing algorithmic music today is a PC with 486 or higher processor and a Windows 95 or higher (for Fmusic), Pentium PC Compatible Computer with 10Mb of free hard disk space and >32Mb RAM (for Gingerbread), or Windows 95 or higher and a moderately fast CPU (for FractMus). However, in order to be able to manipulate our composition even further, it is often necessary to use additional software for notation, such as Finale or Sibelius, and a sequencer (for electronic music and/or high-quality acoustic demos) such as Cakewalk's Sonar, Steinberg's Cubase, MOTU's Digital Performer etc., so the optimal computer to run all these programs fairly quickly would be:

- AMD Athlon XP processor up to 2400+ (2.00GHz) or Intel Pentium 4 (up to 2.0GHz)
- 256-512MB PC2700 DDR SDRAM
- >30 GB Ultra DMA hard drive
- CD writer
- Graphic card, such as NVIDIA or ATI board, (with at least 64MB allocated video memory)
- Sound card such as Sound Blaster Live! Or Audigy
- Speakers

All the examples mentioned in this thesis are composed and recorded on a following configuration:

<u>HP Pavilion 734n</u>

- AMD Athlon XP processor 2400+ (2.0GHz)
- 512MB PC2700 DDR SDRAM memory
- 80GB Ultra DMA hard drive
- CD writer
- NVIDIA GeForceFX 9600 graphics with 256MB allocated video memory
- Creative Sound Blaster Live! Sound card
- Speakers

This thesis shows how algorithmic music is generated, and – at the same time – focuses on Fmusic, program that uses both fractals and cellular automata to generate music, and is flexible enough to allow composer to have control over many important musical parameters, such as rhythm, loudness, pitch distribution, etc.

# CHAPTER 1

## Fractals in music

We can see those bright, strange, beautiful shapes we call fractals everywhere, but what are they, really? Fractals are geometric figures, just like rectangles, circles and squares, but fractals have special properties that those figures do not have. Pioneering mathematician Benoit Mandelbrot is known as the creator of fractal geometry. “Mandelbrot is one of the few living mathematicians whose originality has given birth to entire disciplines,” said physicist Philip Morrison. He was largely self-taught, allowing him to think in unconventional ways and develop a highly geometrical approach to mathematics. In 1958, Mandelbrot joined IBM, delving into processes with unusual statistical properties and geometric features. This led to his famous contributions in fractal geometry.
His 1967 article in Science, “How long is the coast of Britain?” is generally considered a milestone in science and mathematics. He coined the word “fractal” to describe objects, shapes or behaviors that have similar properties at all levels of magnification. In his *The Fractal Geometry of Nature* he states: *“I coined fractal from the Latin adjective fractus. The corresponding Latin verb frangere means – to break*

*– to create irregular fragments. It is therefore sensible – and how appropriate for our needs! – that, in addition to 'fragmented' (as in fraction or refraction) fractus should also mean 'irregular', both meanings preserved in fragment."*

Many objects in nature aren't formed of squares or triangles, but of more complicated geometric figures. Many natural objects - ferns, coastlines, etc. – are shaped like fractals. Coastlines, for example could be measured with very large – or very small rulers. If we use infinitesimally short rulers, the coastline would become infinitely long. Of course, no real object can have infinite details; however, most real objects, up to limits imposed by the "granularity" of matter, do continue to display additional details when magnified and can be better modeled with 'fractals' than with simple classical geometrical figures.

One of the most important fractal characteristics is *self-similarity.* In order for one figure to be similar to another, you must be able to zoom in on one of the figures, and it will look exactly like the other one (you might have to rotate it or flip it first), for example (fig. 1.1):

Figure 1.1: Self-similarity, trapezoid

The outline of this figure is a trapezoid, and all of its components are also trapezoids. Let us now look at Figure 1.2: Self-similarity, L-shape (fig. 1.2). The outline, and all of its components are "L" shapes:

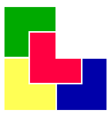


Figure 1.2: Self-similarity, L-shape

Two of the most famous fractals that show obvious self-similarity are *Sierpinski's Triangle* (fig. 1.3) and *Koch's Curve* (fig. 1.4).

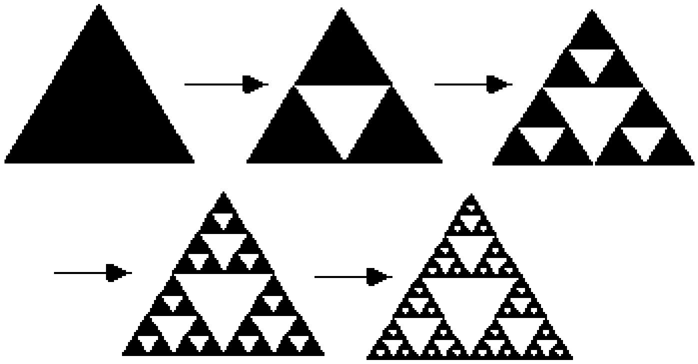


Figure 1.3: Sierpinski's Triangle

This design is called Sierpinski's Triangle (or gasket), after the Polish mathematician Waclaw Sierpinski who described some of its interesting properties in 1916. Among these is its fractal or self-similar character. The large triangle consists of three smaller black triangles, each of which itself consists of three smaller black triangles, each of which... *ad infinitum*. This is obviously a process of subdivision which could, with adequate resolution, be seen to continue indefinitely.

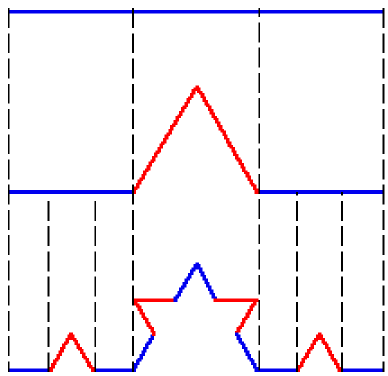


Figure 1.4: Koch Curve

**E. Cesaro**, *quoted in Measure, Topology, and Fractal Geometry, by G.* Edgar: *"It is this similarity between the whole and its parts, even infinitesimal ones, that makes us consider this curve of von Koch as a line truly marvelous among all. If it were gifted with life, it would not be possible to destroy it without annihilating it whole, for it would be continually reborn from the depths of its triangles, just as life in the universe is."*

Three copies of the Koch curve placed around the three sides of an equilateral triangle, form a simple, but very beautiful closed curve known as the *Koch Snowflake* (fig. 1.5).

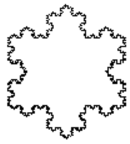


Figure 1.5: Koch Snowflake

What is the connection between fractal geometry and music, or more precisely – automated music composition? The term *Fractal music* typically refers to music composed wholly or in-part using the same types of iterative or recursive processes used to create fractal images, “translating” the results of fractal formulae into musical scales – major, minor, chromatic, blues, or any custom scale, and/or rhythmic modus, creating melodies, harmonies and rhythms during computer-based fractal music composition. The process of converting from the “fractal” value type into a musical parameter is usually called *mapping.* Computer will typically map a floating point x, y position value, for example -1.234, 45.8765, onto some musical value such as a MIDI pitch – or duration value.

There are numerous, if not infinite, ways to map numerical values to musical parameters. Different mappings can have a substantial impact on the musical results.

Fractal music is, therefore, a result of a process where an algorithm is applied multiple times to process its previous output. In wider perspective – all musical forms, both in micro and macro level can be modeled with this process. Fractals provide extremely interesting musical results, and the field is becoming one of the the most exciting fields of new music research.

Fractal sequences of numbers make excellent raw material for melodies and musical rhythms. Many of the musically most interesting fractal sequences come from number theory – some of those sequences can be very simple, such as Morse-Thue sequence, while others can be extremely complex. To calculate musical parameters from a complex fractal image or number string, we need a computer; but in some cases we can generate melodies by mapping fractal values into a tone-row manually.

That's obviously not an easiest way to compose fractal music, but it makes the whole process much more clear. Let's look at Sierpinski's triangle again: we can use fourth triangle from fig. 1.3 to generate a string of binary numbers. Here is a discrete version of Sierpinski Triangle (gasket) where "1" corresponds to black and "0" to white:

```
                1
              1   1
            1   0   1
          1   1   1   1
        1   0   0   0   1
      1   1   0   0   1   1
    1   0   1   0   1   0   1
  1   1   1   1   1   1   1   1
1   0   0   0   0   0   0   0   1
```

Now, let's count the number of ones for each integer. This yields the sequence:

1, 2, 2, 4, 2, 4, 4, 8, 2

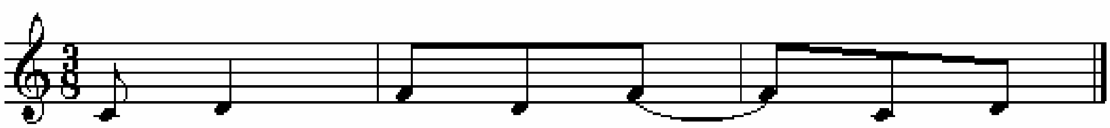
This string could be converted to sound by mapping its individual values onto a musical scale. If we use C-major scale (1=C, 2=D, 3=E, 4=F, 5=G, 6=A, 7=B, 8=C etc.) our musical “score” will look like this:

CDDFDFFCD

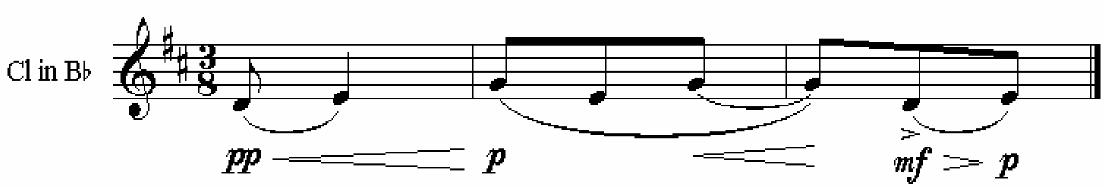
To introduce an element of rhythm into this little tune, we may decide to play a repeated note only once and hold it until different note comes. This turns our melody into:

CD FDF CD

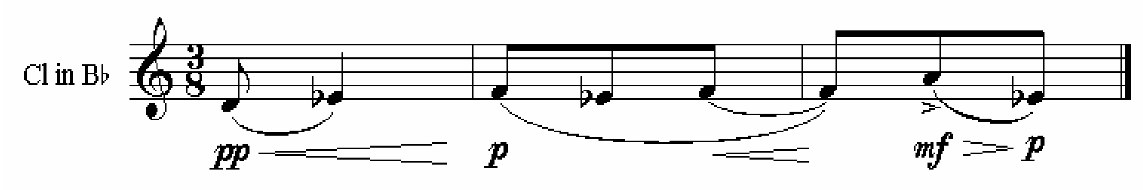
If the smallest rhythmical value we want to use is, for example, 8th-note – our score becomes



Dynamic values can be created freely, or derived from the same series of numbers: 1=pp, 2=p, 3=mp, 4=mf, 5=f, 6=ff, 7 wraps around to pp, 8=p etc. With some additional phrasing and articulation, we could have a small “invention” for clarinet:



This, of course, is just one of many possible mappings. We can use other scales: church modes, folk scales, Messiaen's modes, chromatic scale etc. Our string of numbers mapped onto chromatic scale would produce this little piece:




If we use computer to generate very long string of numbers from Sierpinski Triangle, we can decide to keep only every 3rd, 5th, 10th, or 63rd number for example. Melodies constructed in this manner become decidedly more engaging when we "undersample" certain sequences obtained from number theory. Specifically, keeping only every third term of the ones-counting sequence results in a much improved melody; using bigger numbers for undersampling can give even better results. But, most of the time, experimenting with different numerical values is the key.

Let us now look at every 63rd term of the ones-counting sequence translated into the C-major scale; this will produce the following sequence of pitch classes:

BAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAGABACABADABACABAEABACA
BADABACABAFABACABADABACABAEABACABADABACABAGGAABBAACCAABBAADDAABBAACCAABBAAEEAABBAACCAABBAA
DDAABBAACCAABBAAGFGACBCADCDACBCAEDEACBCADCDACBCAFEFACBCADCDACBCAEDEACBCADCDACBCAGGGGAAAABB
BBAAAACCCCAAAABBBBAAAADDDDAAAABBBBAAAACCCCAAAABBBBAAAAGEFEGABADBCBDABAECDCEABADBCBDABAFDED
FABADBCBDABAECDCEABADBCBDABAGGFFGGAACCBBCCAADDCCDDAACCBBCCAAEEDDEEAACCBBCCAADDCCDDAACCBBCC
AAGFGEGFGADCBDCDAEDECEDEADCDBDCDAFEFDFEFADCDBDCDAEDECEDEADCDBDCDAGGGGGGGGAAAAAAAABBBBBBBBA
AAAAAAACCCCCCCCAAAAAAAABBBBBBBBAAAAAAAAGDEDFDEDGABACABAEBCBDBCBEABACABAFCDCECDCFABACABAEBC
BDBCBEABACABAGGEEFFEEGGAABBAADDBBCCBBDDAABBAAEECCDDCCEEAABBAADDBBCCBBDDAABBAAGFGDFEFDGFGAC
BCAEDEBDCDBEDEACBCAFEFCEDECFEFACBCAEDEBDCDBEDEACBCAGGGGFFFFGGGGAAAACCCCBBBBCCCCAAAADDDDCCC
CDDDDAAAACCCCBBBBCCCCAAAAGEFEGDEDGEFEGABAECDCEBCBECDCEABAFDEDFCDCFDEDFABAECDECBCBECDCEABAG
GFFGGEEGGFFGGAADDCCDDBBDDCCDDBBDDCCDDAAEEDDEECCEEDDEEAADDCCDDBBDDCCDDAAGFGEGFGDGFGEGFGAEDE
CEDEBEDECEDEAFEFDFEFCFEFDFEFAEDECEDEBEDECEDEAGGGGGGGGGGGGGGGG…

Let’s search for some patterns in this tone-row. For example - motive ABACABA can be found 12 times:

BAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAGABACABADABACABAEABACA
BADABACABAFABACABADABACABAEABACABADABACABAGGAABBAACCAABBAADDAABBAACCAABBAAEEAABBAACCAABBAA
DDAABBAACCAABBAAGFGACBCADCDACBCAEDEACBCADCDACBCAFEFACBCADCDACBCAEDEACBCADCDACBCAGGGGAAAABB
BBAAAACCCCAAAABBBBAAAADDDDAAAABBBBAAAACCCCAAAABBBBAAAAGEFEGABADBCBDABAECDCEABADBCBDABAFDED
FABADBCBDABAECDCEABADBCBDABAGGFFGGAACCBBCCAADDCCDDAACCBBCCAAEEDDEEAACCBBCCAADDCCDDAACCBBCC
AAGFGEGFGADCBDCDAEDECEDEADCDBDCDAFEFDFEFADCDBDCDAEDECEDEADCDBDCDAGGGGGGGGAAAAAAAABBBBBBBBA
AAAAAAACCCCCCCCAAAAAAAABBBBBBBBAAAAAAAAGDEDFDEDGABACABAEBCBDBCBEABACABAFCDCECDCFABACABAEBC
BDBCBEABACABAGGEEFFEEGGAABBAADDBBCCBBDDAABBAAEECCDDCCEEAABBAADDBBCCBBDDAABBAAGFGDFEFDGFGAC
BCAEDEBDCDBEDEACBCAFEFCEDECFEFACBCAEDEBDCDBEDEACBCAGGGGFFFFGGGGAAAACCCCBBBBCCCCAAAADDDDCCC
CDDDDAAAACCCCBBBBCCCCAAAAGEFEGDEDGEFEGABAECDCEBCBECDCEABAFDEDFCDCFDEDFABAECDECBCBECDCEABAG
GFFGGEEGGFFGGAADDCCDDBBDDCCDDBBDDCCDDAAEEDDEECCEEDDEEAADDCCDDBBDDCCDDAAGFGEGFGDGFGEGFGAEDE
CEDEBEDECEDEAFEFDFEFCFEFDFEFAEDECEDEBEDECEDEAGGGGGGGGGGGGGGGG…

But, that’s not all; here are some augmentations of the motive ABACABA – for example “AABBAACCAABBAA.” This motive is symmetrically augmented:

BAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAGABACABADABACABAEABACA
BADABACABAFABACABADABACABAEABACABADABACABAGGAABBAACCAABBAADDAABBAACCAABBAAEEAABBAACCAABBAAD
DAABBAACCAABBAAGFGACBCADCDACBCAEDEACBCADCDACBCAFEFACBCADCDACBCAEDEACBCADCDACBCAGGGGAAAABBBB
AAAACCCCAAAABBBBAAAADDDDAAAABBBBAAAACCCCAAAABBBBAAAAGEFEGABADBCBDABAECDCEABADBCBDABAFDEDFAB
ADBCBDABAECDCEABADBCBDABAGGFFGGAACCBBCCAADDCCDDAACCBBCCAAEEDDEEAACCBBCCAADDCCDDAACCBBCCAAGF
GEGFGADCBDCDAEDECEDEADCDBDCDAFEFDFEFADCDBDCDAEDECEDEADCDBDCDAGGGGGGGGAAAAAAAABBBBBBBBAAAAAA
AACCCCCCCCAAAAAAAABBBBBBBBAAAAAAAAGDEDFDEDGABACABAEBCBDBCBEABACABAFCDCECDCFABACABAEBCBDBCBE
ABACABAGGEEFFEEGGAABBAADDBBCCBBDDAABBAAEECCDDCCEEAABBAADDBBCCBBDDAABBAAGFGDFEFDGFGACBCAEDEB
DCDBEDEACBCAFEFCEDECFEFACBCAEDEBDCDBEDEACBCAGGGGFFFFGGGGAAAACCCCBBBBCCCCAAAADDDDCCCCDDDDAAA
ACCCCBBBBCCCCAAAAGEFEGDEDGEFEGABAECDCEBCBECDCEABAFDEDFCDCFDEDFABAECDECBCBECDCEABAGGFFGGEEGG
FFGGAADDCCDDBBDDCCDDBBDDCCDDAAEEDDEECCEEDDEEAADDCCDDBBDDCCDDAAGFGEGFGDGFGEGFGAEDECEDEBEDECE
DEAFEFDFEFCFEFDFEFAEDECEDEBEDECEDEAGGGGGGGGGGGGGGGG…

Slight variations are also possible, for example:

BAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAGABACABADABACABAEABACA
BADABACABAFABACABADABACABAEABACABADABACABAGGAABBAACCAABBAADDAABBAACCAABBAAEEAABBAACCAABBAAD
DAABBAACCAABBAAGFGACBCADCDACBCAEDEACBCADCDACBCAFEFACBCADCDACBCAEDEACBCADCDACBCAGGGGAAAABBBB
AAAACCCCAAAABBBBAAAADDDDAAAABBBBAAAACCCCAAAABBBBAAAAGEFEGABADBCBDABAECDCEABADBCBDABAFDEDFAB
ADBCBDABAECDCEABADBCBDABAGGFFGGAACCBBCCAADDCCDDAACCBBCCAAEEDDEEAACCBBCCAADDCCDDAACCBBCCAAGF
GEGFGADCBDCDAEDECEDEADCDBDCDAFEFDFEFADCDBDCDAEDECEDEADCDBDCDAGGGGGGGGAAAAAAAABBBBBBBBAAAAAA
AACCCCCCCCAAAAAAAABBBBBBBBAAAAAAAAGDEDFDEDGABACABAEBCBDBCBEABACABAFCDCECDCFABACABAEBCBDBCBE
ABACABAGGEEFFEEGGAABBAADDBBCCBBDDAABBAAEECCDDCCEEAABBAADDBBCCBBDDAABBAAGFGDFEFDGFGACBCAEDEB
DCDBEDEACBCAFEFCEDECFEFACBCAEDEBDCDBEDEACBCAGGGGFFFFGGGGAAAACCCCBBBBCCCCAAAADDDDCCCCDDDDAAA
ACCCCBBBBCCCCAAAAGEFEGDEDGEFEGABAECDCEBCBECDCEABAFDEDFCDCFDEDFABAECDECBCBECDCEABAGGFFGGEEGG
FFGGAADDCCDDBBDDCCDDBBDDCCDDAAEEDDEECCEEDDEEAADDCCDDBBDDCCDDAAGFGEGFGDGFGEGFGAEDECEDEBEDECE
DEAFEFDFEFCFEFDFEFAEDECEDEBEDECEDEAGGGGGGGGGGGGGGGG…

“C” is here replaced with “D” (ABADABA). There are also variations such as ABAEABA or ABAFABA. Augmentation with interpolation might also appear in long strings:

BAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAGABACABADABACABAEABACA
BADABACABAFABACABADABACABAEABACABADABACABAGGAABBAACCAABBAADDAABBAACCAABBAAEEAABBAACCAABBAAD
DAABBAACCAABBAAGFGACBCADCDACBCAEDEACBCADCDACBCAFEFACBCADCDACBCAEDEACBCADCDACBCAGGGGAAAABBBB
AAAACCCCAAAABBBBAAAADDDDAAAABBBBAAAACCCCAAAABBBBAAAAGEFEGABADBCBDABAECDCEABADBCBDABAFDEDFAB
ADBCBDABAECDCEABADBCBDABAGGFFGGAACCBBCCAADDCCDDAACCBBCCAAEEDDEEAACCBBCCAADDCCDDAACCBBCCAAGF
GEGFGADCBDCDAEDECEDEADCDBDCDAFEFDFEFADCDBDCDAEDECEDEADCDBDCDAGGGGGGGGAAAAAAAABBBBBBBBAAAAA
AAACCCCCCCCAAAAAAAABBBBBBBBAAAAAAAAGDEDFDEDGABACABAEBCBDBCBEABACABAFCDCECDCFABACABAEBCBDBCB
EABACABAGGEEFFEEGGAABBAADDBBCCBBDDAABBAAEECCDDCCEEAABBAADDBBCCBBDDAABBAAGFGDFEFDGFGACBCAEDE
BDCDBEDEACBCAFEFCEDECFEFACBCAEDEBDCDBEDEACBCAGGGGFFFFGGGGAAAACCCCBBBBCCCCAAAADDDDCCCCDDDDAA
AACCCCBBBBCCCCAAAAGEFEGDEDGEFEGABAECDCEBCBECDCEABAFDEDFCDCFDEDFABAECDECBCBECDCEABAGGFFGGEEG
GFFGGAADDCCDDBBDDCCDDBBDDCCDDAAEEDDEECCEEDDEEAADDCCDDBBDDCCDDAAGFGEGFGDGFGEGFGAEDECEDEBEDEC
EDEAFEFDFEFCFEFDFEFAEDECEDEBEDECEDEAGGGGGGGGGGGGGGGG…

This shows augmentation of the original motive – “AABBAACCAABBAA”; there is a symmetrical interpolation of “DDBB” and its mirror image “BBDD” between first and second AACC: AABBAA**DDBB**CC**BBDD**AABBAA. It would be possible to find other motives – shorter or longer – and their variations in this string with 1047 terms. This leads us to a conclusion that fractals make excellent source for music composition that focuses mostly on the principle of “developing variation.”

For Arnold Schoenberg the most important contribution made by Brahms to the cultivated European musical tradition was his perfection of the techniques of “developing variation.” The beginning stages of this technique can be traced back to Beethoven and in the music of Schoenberg it found a 20th Century manifestation. In this technique Schoenberg found a major compositional tool as he began to wade through the murky waters of atonality. In his own words:

*"One may say: coherence is based on repetition, inasmuch as parts of A recur in B, C, etc. And: Coherence comes into being when parts that are somewhat the same, somewhat different, are connected so that those parts that are the same become prominent. Contrast (relational) is likewise based on coherence, insofar as the same parts as mentioned above are connected so that the unlike parts predominantly attract attention. Change and variation are based on repetition, insofar as several of the like parts as well as several of the dissimilar parts become discernible. Development is one such succession of related ideas, in which disparate parts that are initially subordinate in importance gradually become the principal ideas."*

Obviously, music generated from fractal formulae or fractal pictures has a natural tendency towards developing variation. A simple fractal picture can demonstrate this much better than words. It is possible to construct very complex structures using only few basic elements (fig.1.6):

Figure 1.6: Fractal Fern

On examining long fractal melodies, one always discovers interesting patterns: small melodic structures repeat within larger structures that are in turn repeated within even larger structures – that's the beauty of fractal forms; galaxies within clusters of galaxies in the design of our universe.

# CHAPTER 2

## Artificial life and music composition: Cellular Automata

From the theoretical point of view, *Cellular Automata* (CA) were introduced in the late 1940´s by John von Neumann and Stanislaw Ulam. From the more practical point of view it was in the late 1960´s when John Horton Conway developed the *Game of Life*.

CA´s are *discrete dynamical systems* and are often described as a counterpart to *partial differential equations*, which have the capability to describe *continuous* dynamical systems. The meaning of *discrete* is, that space, time and properties of the automaton can have only a finite, countable number of states. The basic idea is not to try to describe a complex system from "above" - to describe it using difficult equations, but simulating this system by interaction of cells following easy rules. In other words – **Not to describe a *complex* system with *complex* equations, but let the complexity emerge by interaction of *simple* individuals following *simple* rules.** Hence the essential properties of a CA are:

- a *regular n-dimensional lattice* (n is in most cases of one or two dimensions), where each *cell* of this lattice has a discrete state,
- a *dynamical behavior*, described by so called *rules*. These rules describe the state of a cell for the next time step, depending on the states of the cells in the neighborhood of the cell.

The first system extensively calculated on computers is – as mentioned above – the *Game of Life*. This game became that popular, that a scientific magazine published regularly articles about the "behavior" of this game. Contests were organized to prove certain problems.

The Game of Life (or simply Life) is not a game in the conventional sense. There are no players, and no winning or losing. Once the "pieces" are placed in the starting position, the rules determine everything that happens later. Nevertheless, Life is full of surprises! In most cases, it is impossible to look at a starting position (or *pattern*) and see what will happen in the future. The only way to find out is to follow the rules of the game. Life is played on a grid of square cells – like a chess board but extending infinitely in every direction. A cell can be **live** or **dead**. A live cell is shown by putting a marker on its square. A dead cell is shown by leaving the square empty. Each cell in the grid has a neighborhood consisting of the eight cells in every direction including diagonals. To apply one step of the rules, we count the number of live neighbors for each cell. What happens next depends on this number:

- A dead cell with exactly three live neighbors becomes a live cell (birth, fig. 2.1):

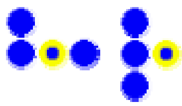


Figure 2.1: “Birth”

- A live cell with two or three live neighbors stays alive (survival, fig. 2.2):

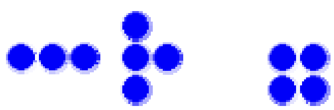


Figure 2.2: “Survival”

- In all other cases, a cell dies or remains dead (overcrowding or loneliness, fig. 2.3):

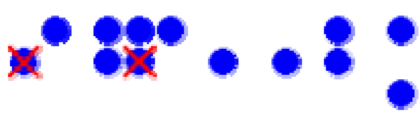


Figure 2.3: “Death”

The number of live neighbors is always based on the cells **before** the rule was applied. In other words, we must first find all of the cells that change before changing any of them.

Life was invented by the mathematician John Conway in 1970. He chose the rules carefully after trying many other possibilities, some of which caused the cells to die too fast and others which caused too many cells to be born. Life balances these tendencies, making it hard to tell whether a pattern will die out completely, form a stable population, or grow forever.

Life is just one example of a **cellular automaton**, which is any system in which rules are applied to cells and their neighbors in a regular grid.

There has been much recent interest in cellular automata, a field of mathematical research. Life is one of the simplest cellular automata to have been studied, but many others have been invented, often to simulate systems in the real world.

In addition to the original rules, Life can be played on other kinds of grids with more complex patterns. There are rules for playing on hexagons arranged in a honeycomb pattern, and games where cells can have more than two states (live cells with different colors).

In the late 1980´s the interest on CA´s arised again, as powerful computers became widely available. Today a set of accepted applications in simulation of dynamical systems are available. Cellular automata have been used to model a wide range of scientific phenomena. They have been studied and developed for over three decades. Although very simple, they can provide models for a wide variety of complex phenomena in a range of disciplines including physics (e.g., dynamic and chaotic systems), genetics and chemistry (e.g., chemical reactions and crystal growth).

Music has always been an interesting domain for the application of new scientific discoveries inviting composers to combine artistic creativity with scientific methods. Today, it is becoming increasingly common for the composer to turn to the sciences to supplement his or her compositional models. One can create very intricate musical phrases translating (or "*mapping*") fractal pictures and fractal formulae onto some musical value such as a MIDI pitch, but cellular automata are becoming quite popular too.

Mapping CA values is very similar to mapping fractal strings. Here is how CA works – We will use a table to illustrate a very simple CA. It consists of an array of 12 cells and each cell can value either 0 or 1, represented by the colors white or black, respectively. The values of all 12 cells change simultaneously according to a set of rules that determines a new value for each cell. In this case, the rules are based upon the values of its two neighbors. For example, if a cell (0 or 1, doesn't matter) has neighbors that are represented with the same number (two zeroes, or two ones), then this cell is equal to zero in the next stage (see figure 2.4 below).

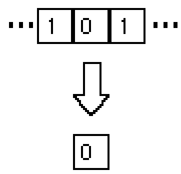



Figure 2.4: Simple CA rules

In every other case – neighbors are 1 and 0 or 0 and 1, the cell will become (or continue to be) “alive”, and its value will be 1.

Now, let’s look at a full pattern:

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |

It will “grow” like this after 5 cycles (this includes original pattern):

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |

Of course, more sophisticated CA configurations use a matrix of cells that can assume values other than 0 and 1 (and consequently, colors other than black and white). In these cases, the transition rules normally computes four or eight neighbors. It is possible to calculate everything manually, but it is not very fast, nor practical; it is much easier to use dedicated computer programs for such calculations, and “mapping” them onto musical values.

But, let’s look at our matrix here. If we translate zeroes to white and ones to black, our “virtual organism” will look like this:

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |

If we decide to count the number of ones before each “white block,” for example, and repeat the process for every line, we will end up with the sequence:

1, 2, 2, 1, 1, 2, 2, 3, 3, 2, 1, 1, 2, 3, 3, 1, 2, 2, 1

This string could be converted to sound by mapping its individual values onto a musical scale. This time, let’s use C 2:1 octatonic scale:

C C D Eb DC DEb CD C

If we want to compose a contrasting line, we can start counting from the beginning, but this time focus on white squares before each “black block”:

1, 1, 1, 1, 1, 3, 1, 4, 2, 1, 3, 1, 1, 1, 1, 2, 5, 1, 1, 3, 1, 1

Since our first string of numbers already “covered” first three notes of the scale, we can map our new string on the upper part of the scale:

F=1,

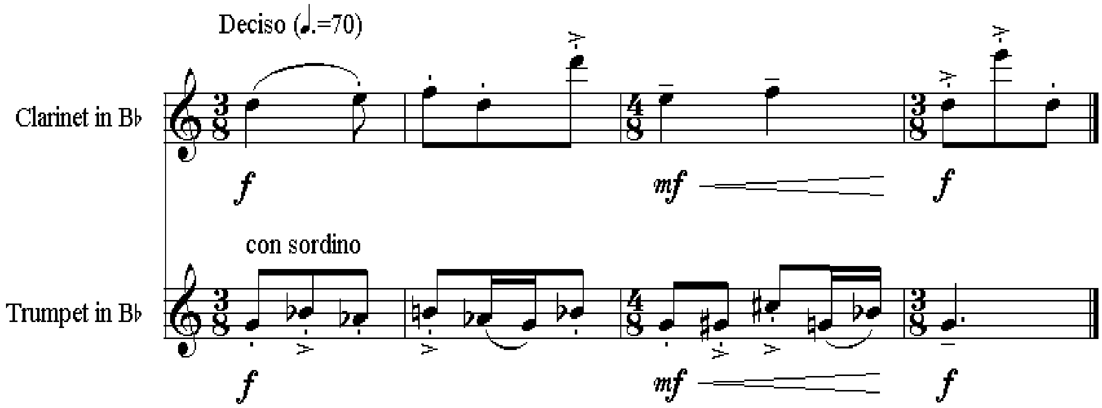
F#=2,

G#=3,

A=4 and

B=5.

This mapping converts the string of numbers into the following tone row: F G#F#AF#FG#F F#BF G#F. Now we can compose rhythm using the method mentioned in chapter one, or create free rhythmical texture for two Bb instruments – trumpet and clarinet, for example:




Obviously, it is possible to organize all – or only few musical elements (melody, harmony, rhythm, timbre, tempo and/or texture) by composers to create compositions. These elements are combined to produce a wide variety of types of music. While many pieces of music may be very linear or have a significant focus on melody, some composers might decide to focus on harmony or rhythm. All these important decisions will define a style or type of music. It is possible to use fractals and cellular automata to compose music similar to serialism, medieval modal music, pop, jazz – and almost anything in between.

In conclusion – music has always been an interesting domain for the application of new scientific discoveries inviting composers to combine artistic creativity with scientific methods. Today, it is becoming increasingly common for the composer to turn to the sciences to supplement his or her compositional models. By the same means, scientists also seem to show interest in the organizational principles to be found in music.

Scientific models carry an important component of human thought, namely formal abstraction, which can be very inspiring for music composition. The game of "Artificial Life" can be used both to model higher-level musical structures, or musical form (e.g., to generate melodies, phrases and entire pieces of music), and lower-level structures, such as the spectra of individual sound events (e.g., sound synthesis research).

# CHAPTER 3

## Fmusic: computer software for automated music composition

Automated composition of original music using fractals and cellular automata is indeed a difficult task, but fortunately, computer programs make it less time consuming. There are few automated music composition programs that use fractals and/or CA for Windows. Some of them can automatically create everything from simple melodies and rock solid drum beats to full arrangements for percussion, bass, strings, woodwinds, brass, and more. These programs give us opportunity to explore huge musical space relatively quickly, but the composer has to have an ability to control many different parameters in a creative way. One of my favorite programs for automated music composition is called Fmusic.

**Fmusic** – The Fractal/CA Music Generator – is designed and written by David H. Singer. This freeware software uses both fractal and cellular automata methods to create and control the music generation, which can be saved in MIDI format. Each Fmusic project can be saved in various parameter files for later explorations. Multitudes of parameters are here at the composer's fingertips, basic instrument parameters such as pitch, pan, patch (timbre), octave (register), volume (loudness) and key. Settings control the type of algorithm used for an instrument, define note patterns (motives), calculate the fractal and control the cellular automata engine. Entire sections can be created and linked together recursively to build quite complex compositions. Unfortunately, being completely free, this software doesn't include manual of any sort, so I will explain how to compose music with it here.

The easiest and fastest way to work with Fmusic is to:

- Create music composition or a short section
- Save it as a MIDI file
- Import this file into a notation program, such as Finale or Sibelius
- "Clean up" the score and make changes if necessary

The Fmusic interface consists of four "tabs." Let's list them: Event Parameters, CA and misc. parameters, Fractal notes and Percussion and bass settings.

**Event parameters** (fig. 3.1)

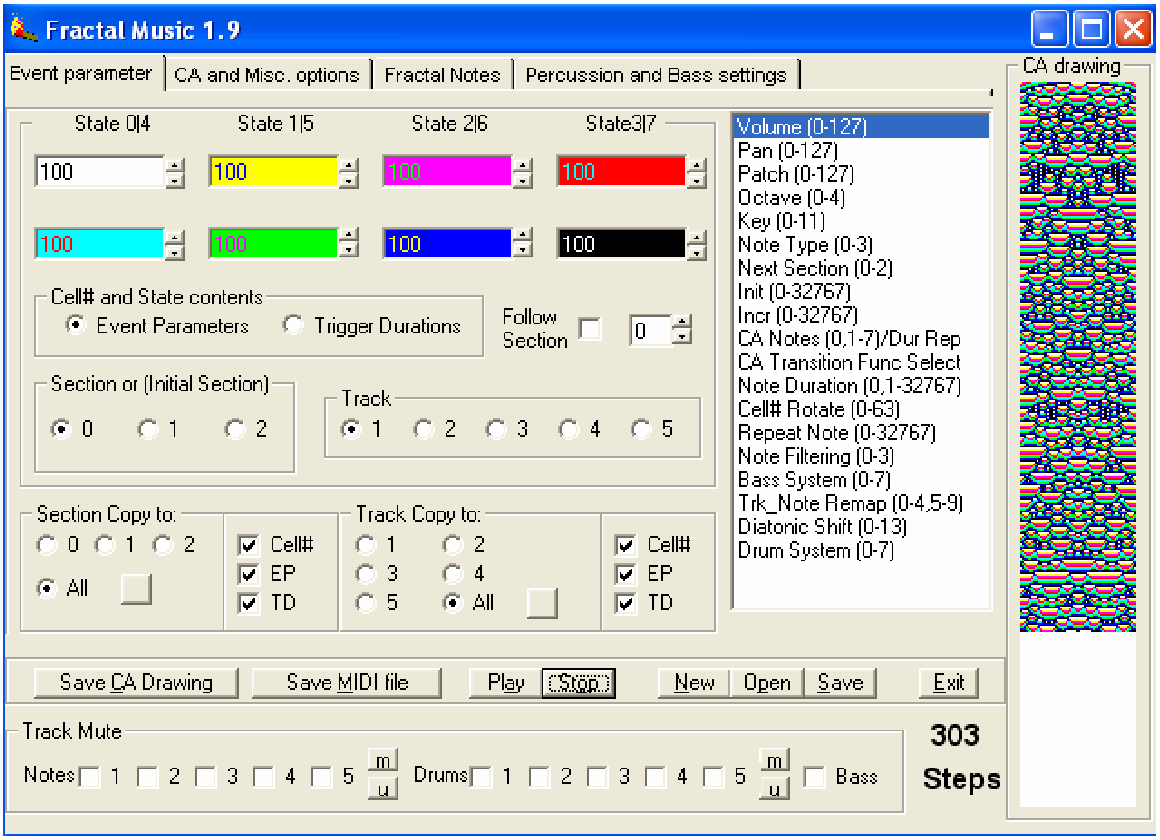



Figure 3.1: Event Parameters

All of the parameters in the list box on the right-hand side can be defined for each of the 8 states (0-7). Event Parameters Tab, lets us control the following parameters:

- Volume
- Pan
- Patch
- Octave
- Key Note type
- Next Section
- Init
- Incr
- CA Notes
- CATransitionFunc Select
- Note Duration
- Cell# Rotate
- Repeat Note

- NoteFiltering
- Bass System
- Trk_Note_Remap
- Diatonic Shift
- Drum System
- Cell# and State Contents Follow Section
- Section (Initial Section)
- Track Section Copy to
- Track Copy to
- Save CA Drawing
- Save Midi File
- Play/Stop
- New/Open/Save

- Exit

- Track Mute

Here we can use standard file functions: Save, Load and create New File. The event parameters are stored in a file with *.ev as the file extension. *Important*: each tabbed page stores it's parameters in a different file, and has to be imported individually. Also, on this page we can start and stop playback, and have the option to save our composition as a MIDI file. It is *very* important to save often before exiting. Fmusic does not have an auto-save feature!

Most of the parameters, such as Volume, Pan, Patch and Octave are self-explanatory. Others, like Note Duration, Cell# Rotate, Repeat Note, and Note Filtering are used for motivic – both melodic and rhythmic – work. Global form is controlled with Diatonic Shift (modulation feature) and Section (InitialSection).

**The CA and Misc. options page** (fig. 3.2)

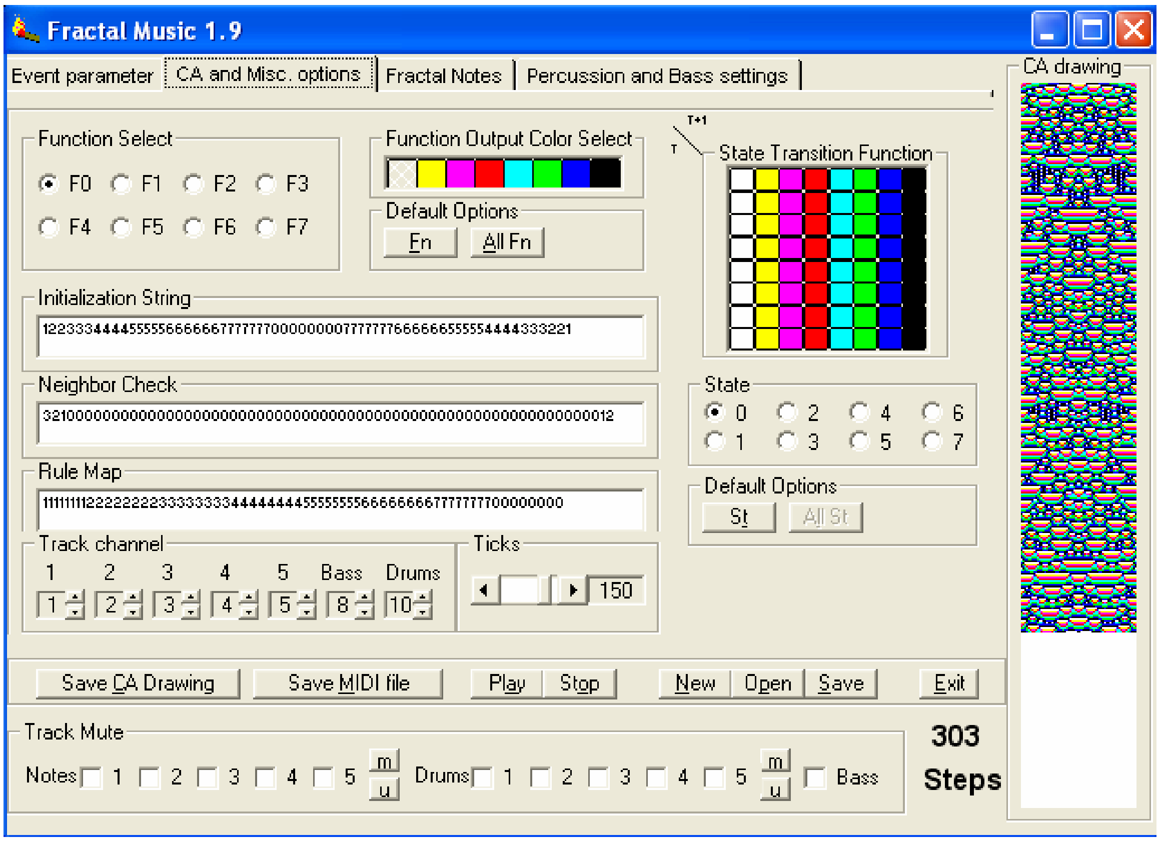



Figure 3.2: CA and Misc. option page

The CA and Misc. options page contains Cellular Automata parameters to set for the 8 different states (Any parameters not mentioned here operate the same as on the Event parameter page – such as save, mute, etc.):

- Function Select/output color select/State transition

- Function Default *Options*

- Initialization String

- Neighbor Check

- Rule Map

- State

- Default Options

- Track channel

- Ticks

On the CA and Misc. options page we can control Cellular Automata processes: By selecting one of the colors and then clicking on squares within the State Transition Function grid, we can force certain state transitions to occur. It is interesting to create simple figurative pictures or draw letters in State transition Function, and see how they translate into musical forms.

Three string values are used in summing levels of the CA matrix, and how each cell is summed with it's neighbors. Here, we can also control which MIDI channel each of the tracks/bass/drums is played on, as well as set the tempo in MIDI ticks – the higher the number, the faster the tempo.

**Fractal Notes page** (fig. 3.3)

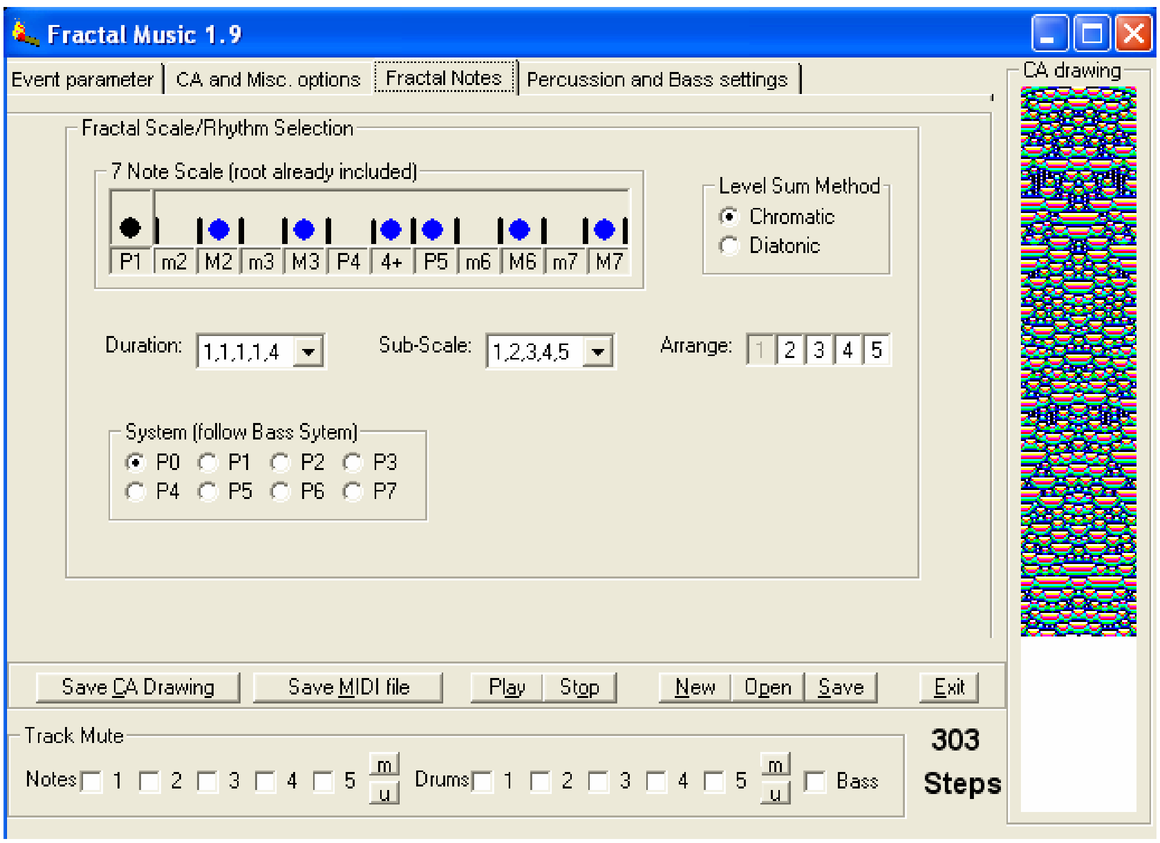



Figure 3.3: Fractal Notes page

This page defines the fractal notes for our composition, and includes the following options (Parameters/Controls not defined here are the same as those on Events Page):

- 7-Note Scale
- Level Sum Method
- Duration
- Sub-Scale Arrange
- System

Most of these controls are self-explanatory. We can define the scale in terms of intervals where: m2 = Minor Second, M3 = Major Second etc., set a duration pattern (1=short, 4=long), select notes within the 7-note scale as a sub-scale, and arrange the order of the notes within the scale. These are excellent tools for composing short motives and intricate motivic work based on repetition and transposition.

**Percussion and Bass settings page** (fig. 3.4)

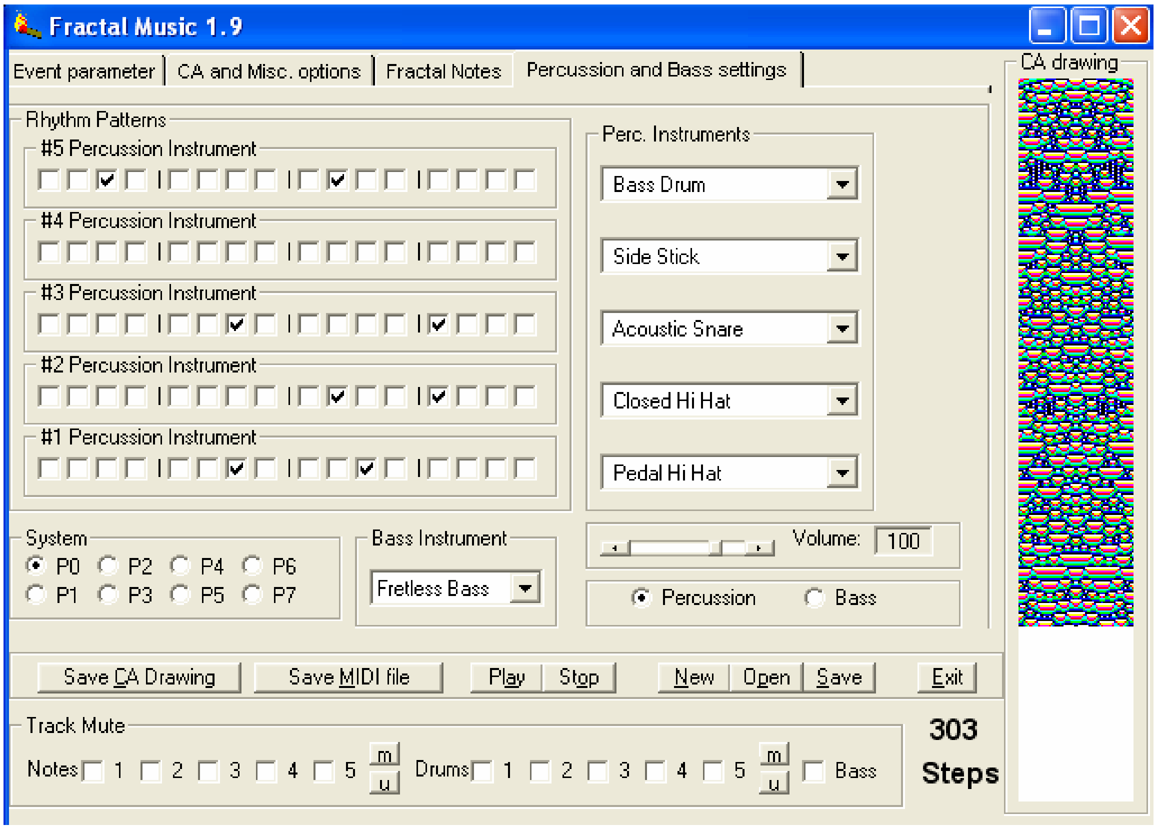



Figure 3.4: Percussion and Bass settings page

Personally, I am not using this page very often, but it certainly has its uses in pop/rock/jazz music. Here we can define the bass and drum patterns for up to eight different drum systems of five drums each, and eight different bass systems of five basses each (see Events page for other common controls):

- Rhythm/Bass Patterns
- Perc. Instruments
- Bass Instrument
- System
- Volume
- Percussion/Bass radio buttons

Here we have a choice of individual percussion instruments from the pull-down list for each of the 5 drum parts, as well as bass timbres available on our sound card or some other General MIDI device, such as synthesizer, sampler, etc.

It is clear – after looking at this long list – there are many parameters to control in Fmusic. As a matter of fact - it takes quite a bit of work on the user's part to create a good composition, but the results are well worth it. The key is, of course, hard work, patience – and experimenting with different settings.

We already mentioned that, for Arnold Schoenberg, the most important contribution made by Brahms to the cultivated European musical tradition was his perfection of the techniques of "developing variation." Let's look at Brahms' philosophy and some of his compositional techniques, and see how we can follow these principles using specific Fmusic parameters:

**1. When you study music, the most important thing to learn is Strict Counterpoint.**

With FMusic, it is relatively easy to compose polyphonic music. Individual melodic lines – and not block chords – are the strengths of this software. It allows us to compose up to 5-part polyphonic texture. Every voice can be muted and “activated” in real time, if necessary.

**2. Writing variations is good for the beginner.**

Fractal images and CE processes are extremely well suited for the technique of “developing variation” due to their complexity and/or self-similarity.

**3. Never begin working the out of a composition before the whole thing has taken definite form as an outline either on paper or in your head.**

Our premeditated global form is created and controlled using Event Parameters, such as Diatonic Shift, CA Transition and Next Section. Many of the parameters can be controlled and changed in real time, which gives us enormous freedom to react to musical texture in an improvisatory manner.

**4. When you are composing a piece, your bass should be vibrant, not sleepy or lazy.**

Bass lines are indeed very important, and vibrant ones can be created simply by copying and pasting certain parameters from the other, traditionally more “active” instrumental lines. In my piece *…she turned from Ares her bright eyes*, bassoon is flexible enough to play melodic lines similar to those of clarinet or flute, but if we write for a bass instrument that is “lazy” by nature, our main motives and melodic ideas can be “stretched” in time using Note Duration in the Event Parameters page, while retaining all desired motivic complexity.

**5. Combining variety with unity can be difficult. It is accomplished by transforming the basic motive more or less recognizably through rhythmic alteration…and through exact or retrograde inversions.**

When we choose our scale, and create melodic motives in the Fractal Notes Page, fractal- and CE processes will automatically transform them thru transposing, inverting, retrograde inverting and rhythmic stretching. We are, of course, responsible for all these parameters; choosing appropriate color patterns and CA rules from the CA and Misc. Options page is of utmost importance. More subtle rhythmic transformations are controlled using Note Duration in the Event Parameters page.

Fmusic gives us all these possibilities and much, much more. For example – interactive, or real-time composition. Almost all of the parameters mentioned in this chapter can be changed "on the fly," which gives us the opportunity to create interesting shifts and contrasts in texture. Of course, for music to be satisfying it must have a path to follow. Too much real-time interaction could lead to something that is no longer a composition but a series of short, independent musical phrases. This turns interactive musical composition into a logical process of determining which components are absolutely necessary to achieve our goal. The artistic side of this process is in determining the goals (Brahms*: "Never begin working the out of a composition before the whole thing has taken definite form as an outline either on paper or in your head"*), the method of achieving these goals, the necessary components, and the component design.

# CONCLUSION

One of my main reasons for using a computer in my compositional work is the element of surprise – unpredictability. Like C.P.E. Bach suggested for musicians that to move their audience they must be moved themselves, I have found out that to surprise the listeners, I have to be surprised myself. The computer has had that function, which means that I give it a task where the direction is clearly defined, but where the details are unpredictable – therefore generating a balance between global order and local disorder that has been very fruitful for me.

Musical form that allows me to achieve my goals could be described as *Invention*. Actually, it might be better to call my algorithmic compositions *processes*, or *procedures* instead of *forms*; it is upon the procedural traits I wish to focus most of the time. To understand this, we should look at these compositions not using traditional "form analysis," but a "procedural analysis" – analysis that is more interested in processes whereby a simple musical idea (motive) is elaborated and re-articulated using fractal- and Cellular Automata parameters to create new material out of old.

My inventions obviously elaborate upon old (and very simple) material to create full-fledged compositions. While a traditional, Bach-style invention has motives, answers, counter-motives usually in a prescribed order – the most important elements in my algorithmic compositions are "inventive" processes whereby the motive is varied using different algorithms, in an unpredictable fashion: transposed, inverted, stretched or compressed in time, counterpointed, and re-textured, to create a complete work.

If I had to use a term to describe this style, I would probably call it *fractal minimalism.*

My composition *…she turned from Ares her bright eyes* is an original musical work for male narrator, alto voice and woodwind quartet (flute, oboe, clarinet, and bassoon), based on selected fragments from short stories about the war in Bosnia, (published as *Sarajevo Marlboro* by award-winning Bosnian writer and journalist Miljenko Jergovic) – and randomly created “computer poetry” about war.

My work is highly programmatic, describing my nation’s experiences during the time of conflict, denouncing the wickedness of war, and warning future generations of the senselessness of taking up arms against fellow men. This composition employs techniques of automated/algorithmic composition, techniques involving using a computer (running Fmusic and Sibelius 2) to generate musical material through my own compositional decision-making and extra-compositional process such as fractals and cellular automata.

While algorithmic/automated composition is often criticized for producing a less-than-intuitive and “cold” result, I intend to challenge these assumptions by employing it in the service of setting Jergovic’s emotionally-charged and moving texts to achieve an emotional and dramatic result.

BIBLIOGRAPHY

Barnsley, M. Fractals Everywhere. Boston, San Diego, New York, Berkeley, London, Sydney, Tokyo, Toronto: Academic Press, Inc., 1988.

Berkowitz, J. Fractal Cosmos. Oakland: Dharma Enterprises/Amber Lotus, 1994.

Pickover, Clifford A., ed. Fractal Horizons: The future Use of Fractals. New York: St. Martin’s Press, 1996.

Garcia, L. The Fractal Explorer. Santa Cruz: Dynamic Press, 1991.

Singer, David. "Fractal Vibes." 7 September 2004 < www.fractal-vibes.com/fm >. (Web page is no longer active)

Alpern, Adam. “Techniques for Algorithmic Composition of Music.” 12. August 2004 < http://alum.hampshire.edu/~adaF92/algocomp/algocomp95.html >

Xenakis, I. Formalized Music: Thought and Mathematics in Composition. Harmonologia Series #6. Hillsdale, NY: Pendragon Press, 2001.

Dodge, C. Profile: “A Musical Fractal.” Computer Music Journal, 1988. 12(3): p. 10-14.

Gogins, M. “Iterated Function Systems Music.” Computer Music Journal, 1991. 15(1): p. 40 - 48.

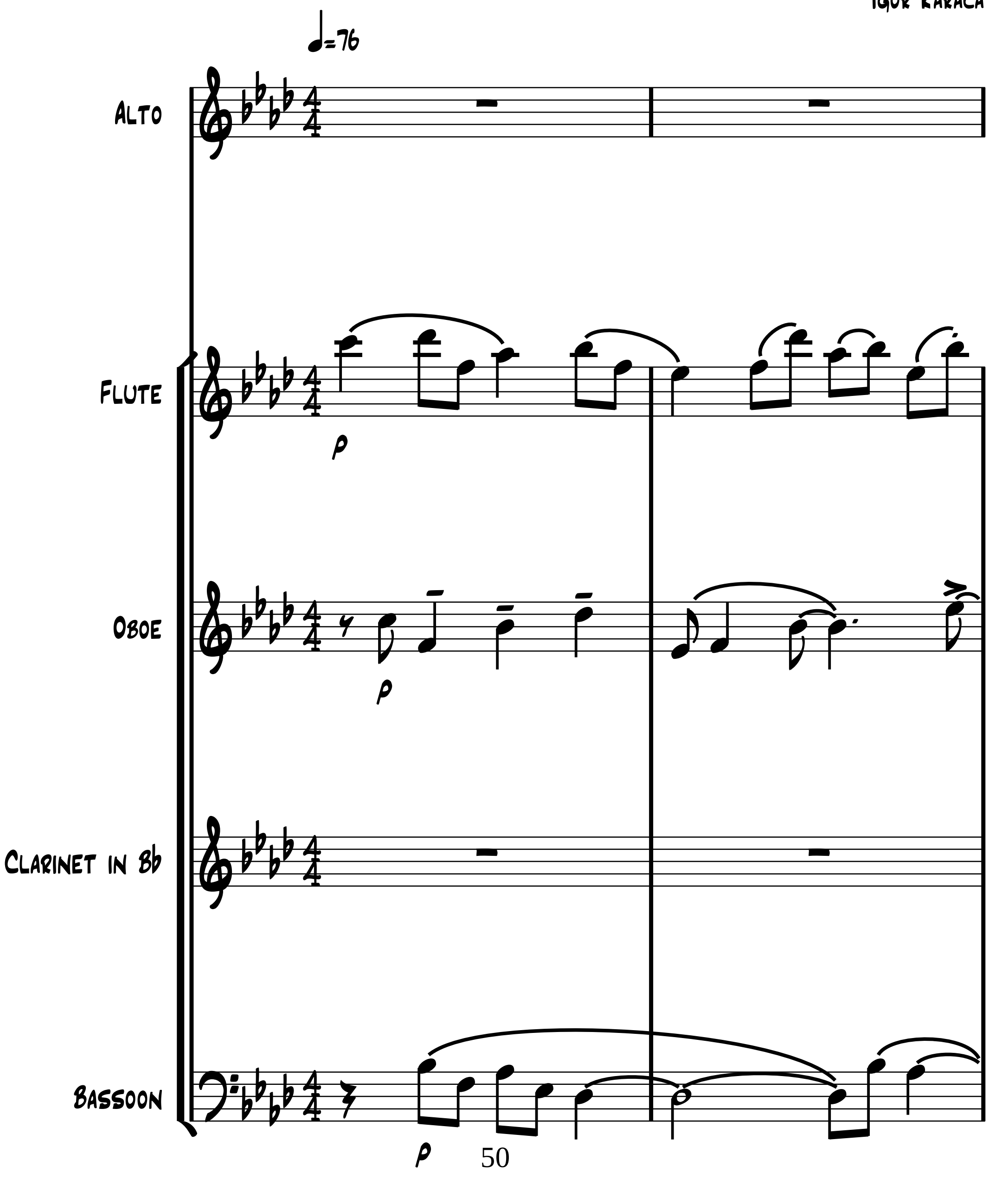
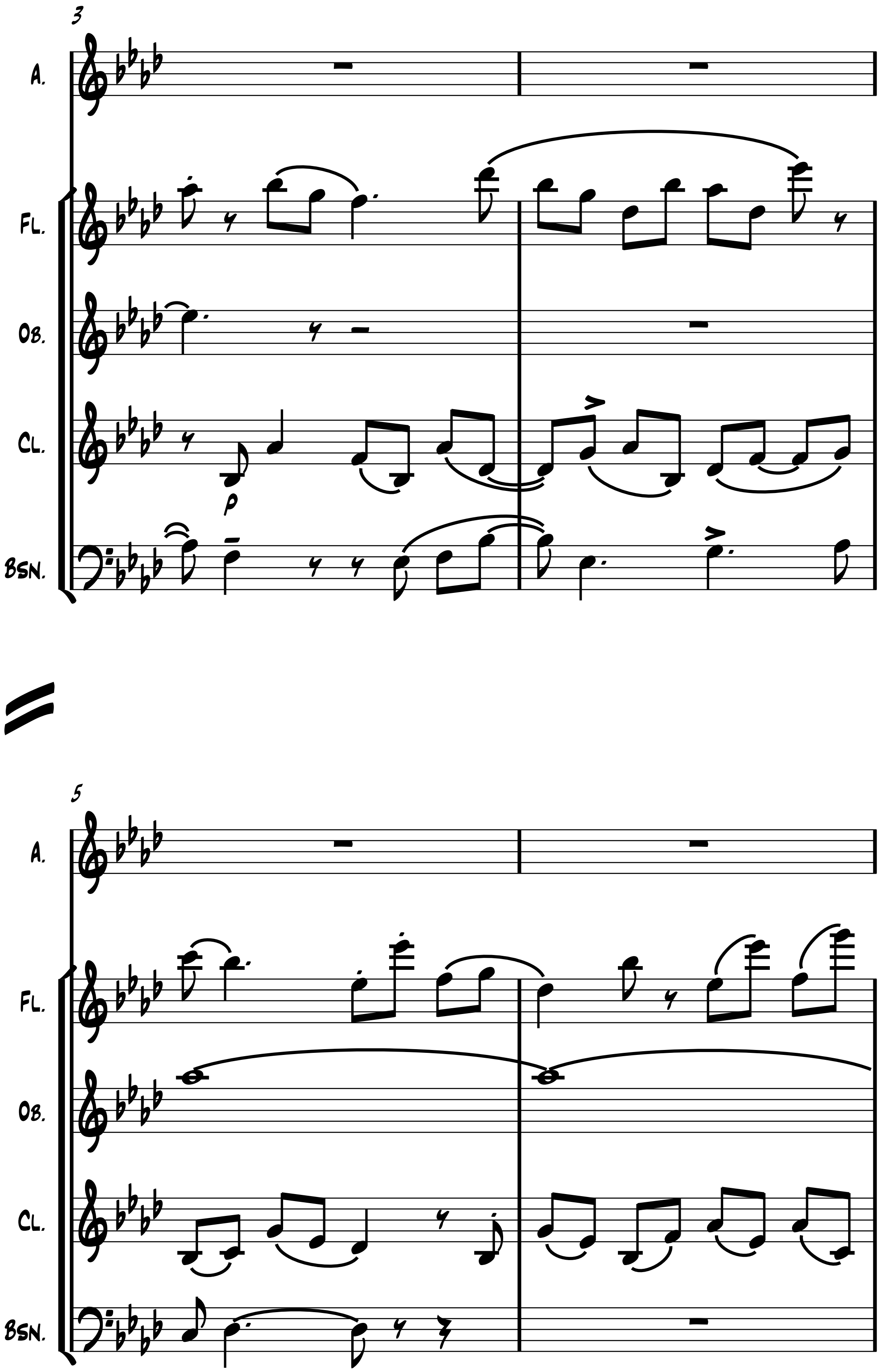
Friberg, A. “Generative Rules for Music Performance: A Formal Description of a rule system.” Computer Music Journal, 1991. 15(2): p. 56-71.

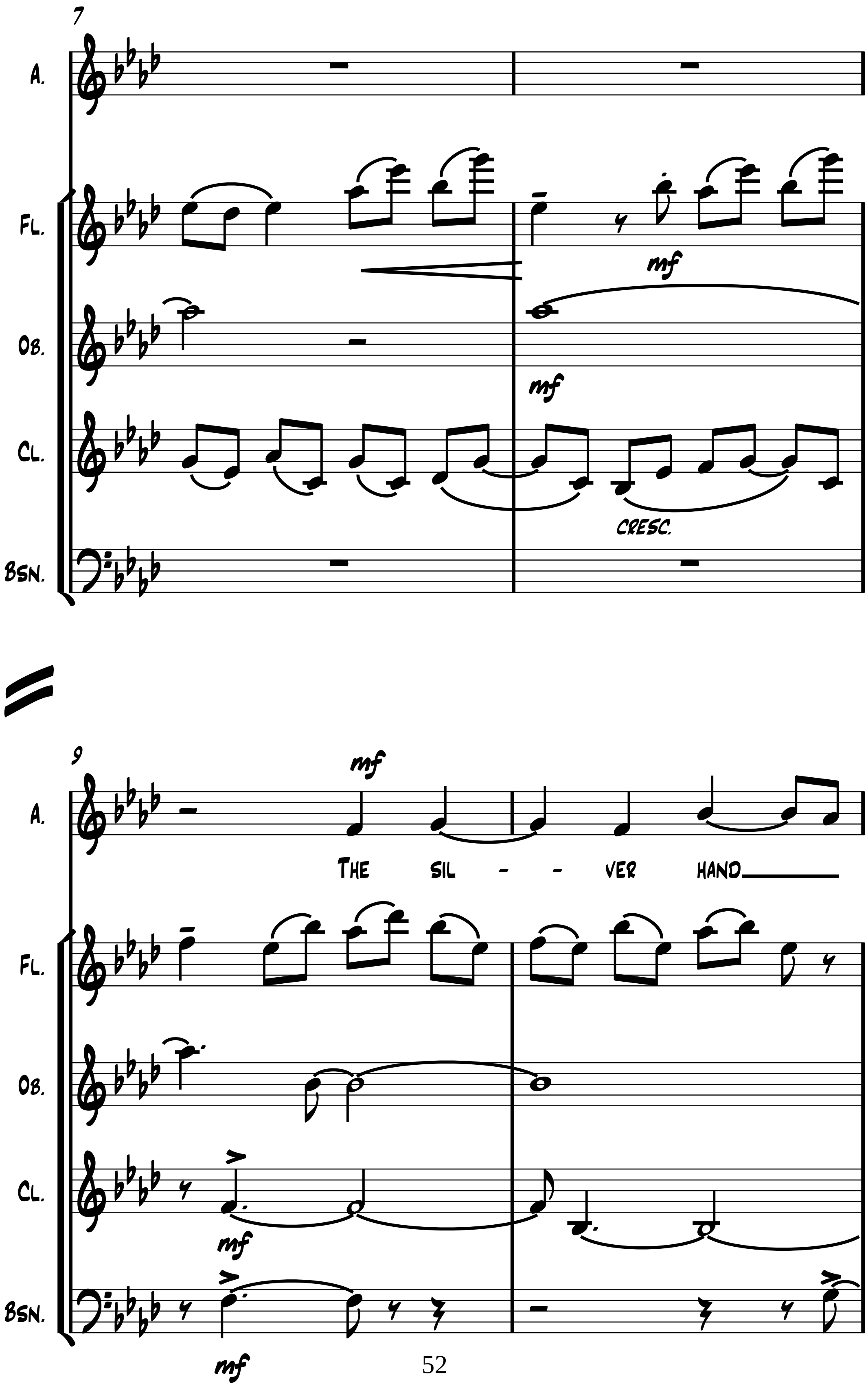
# APPENDIX A

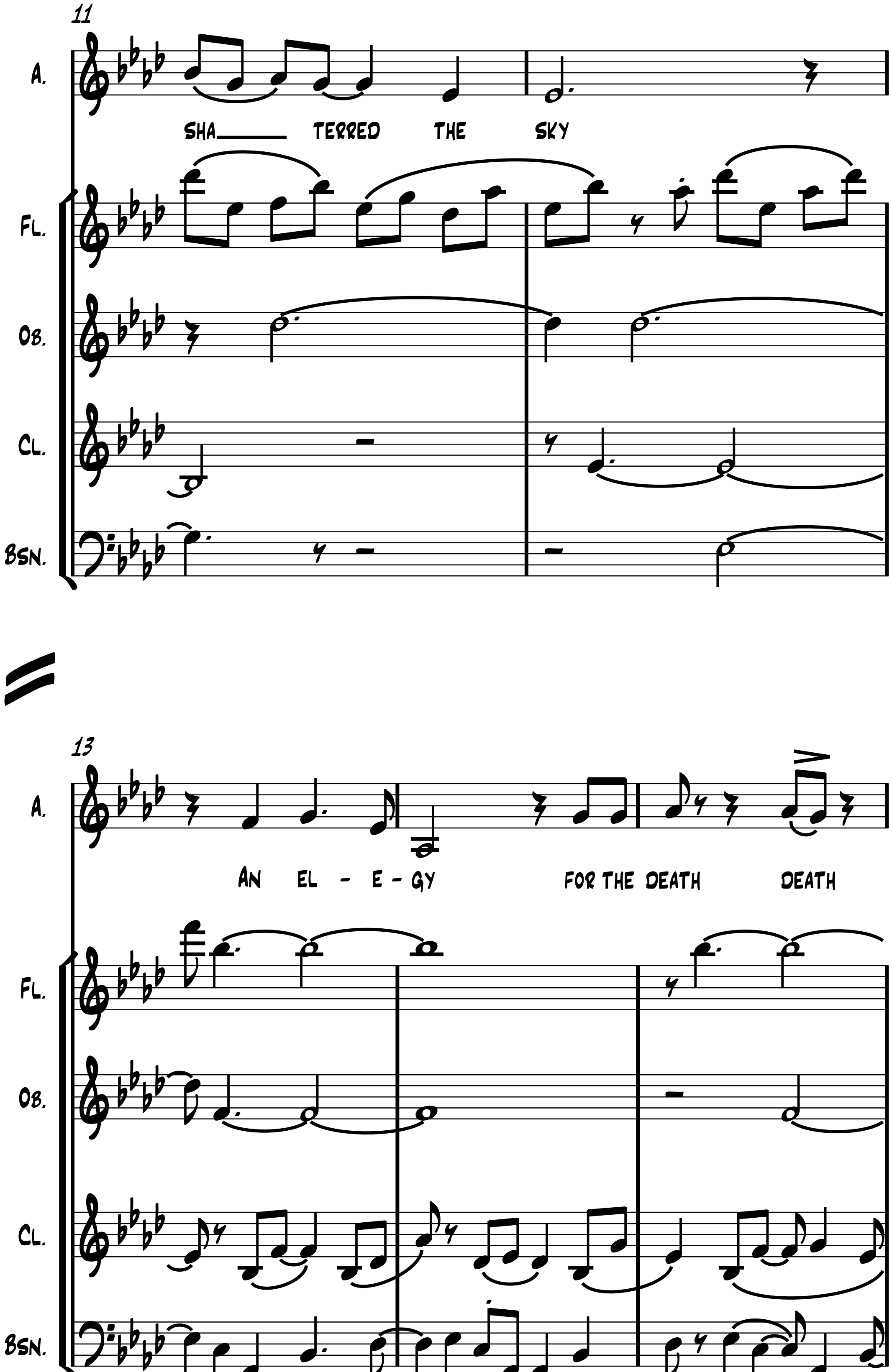
…she turned from Ares her bright eyes, Movement 1

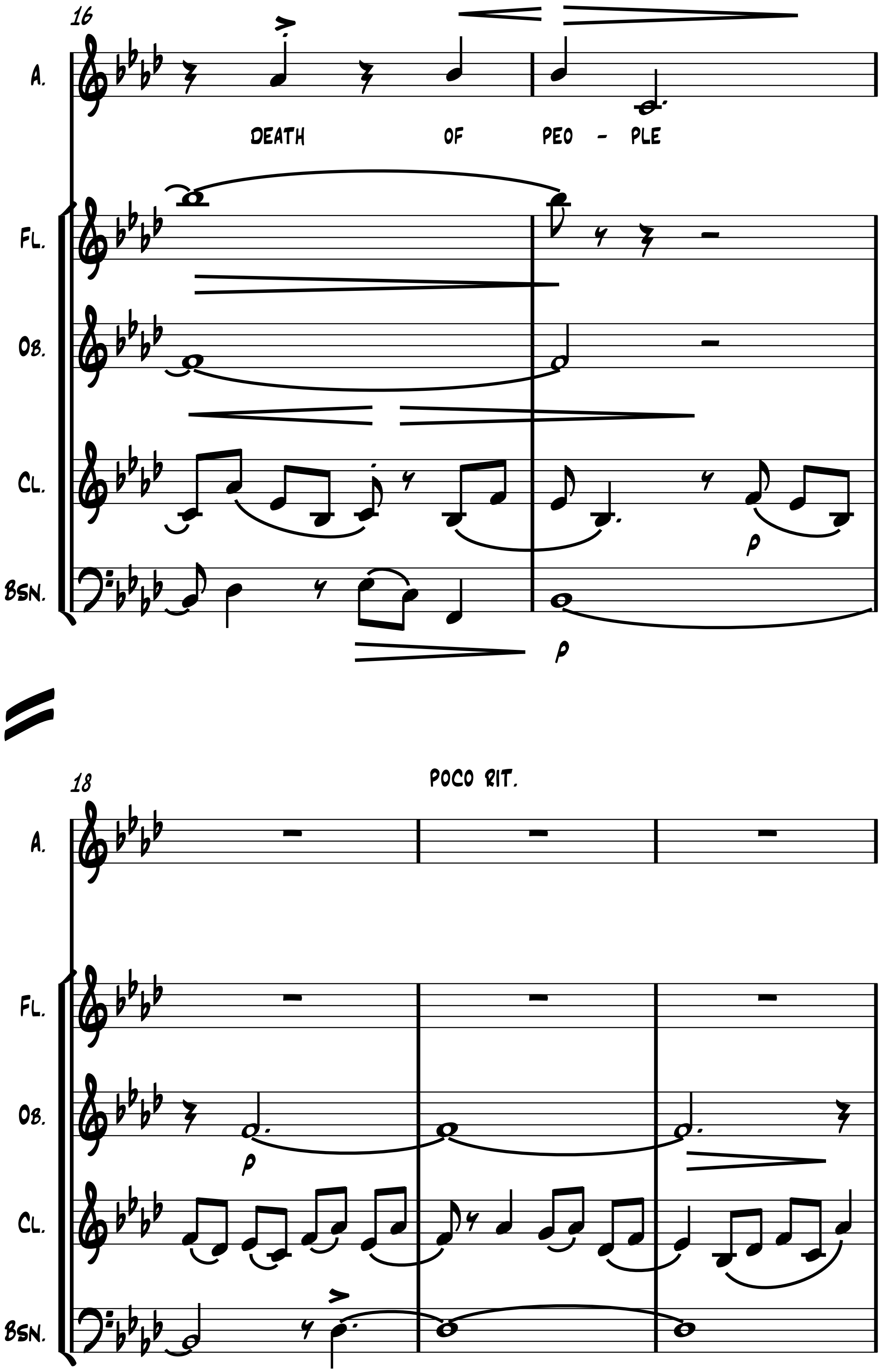
# ...she turned from Ares her bright eyes

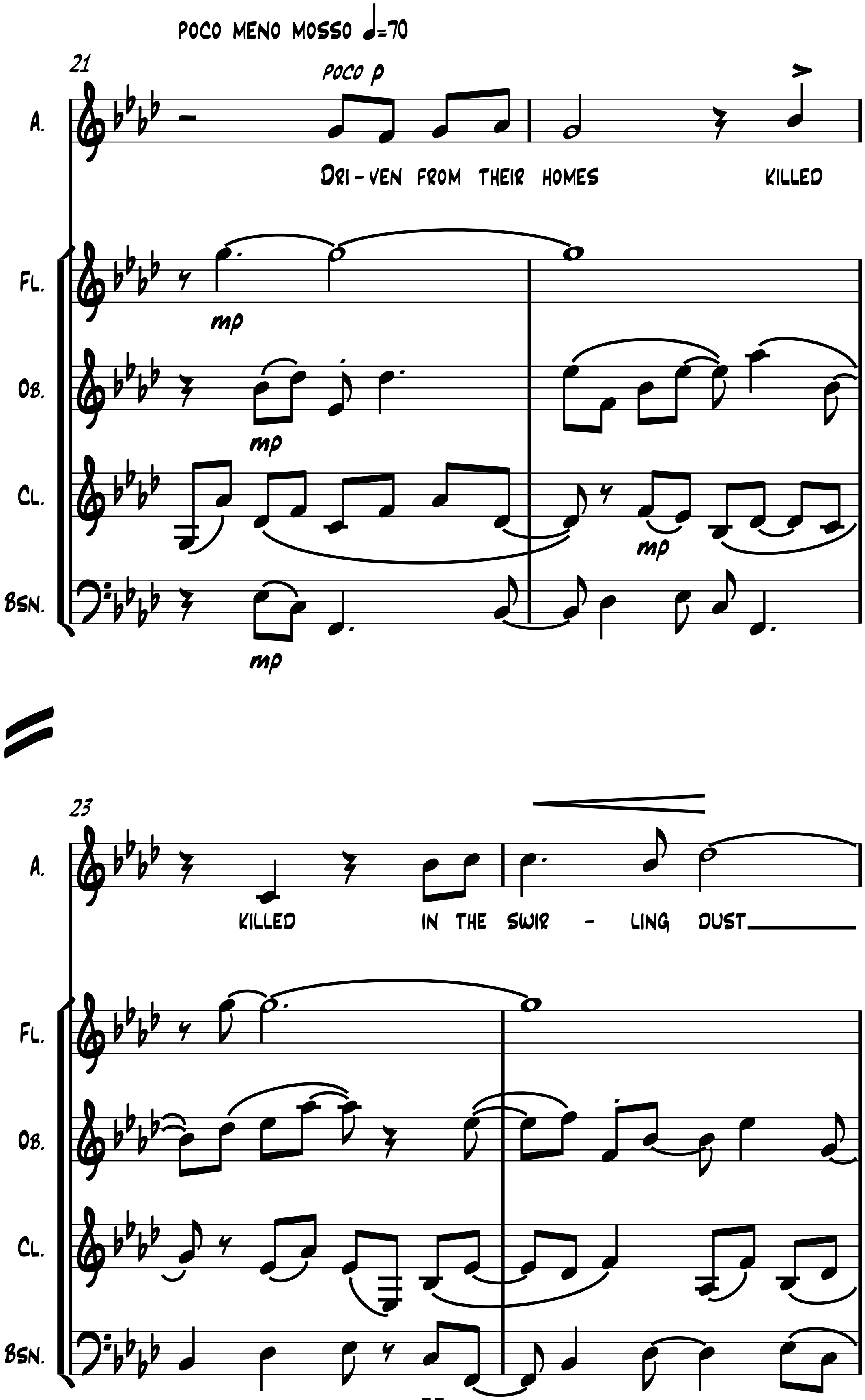
## 1.

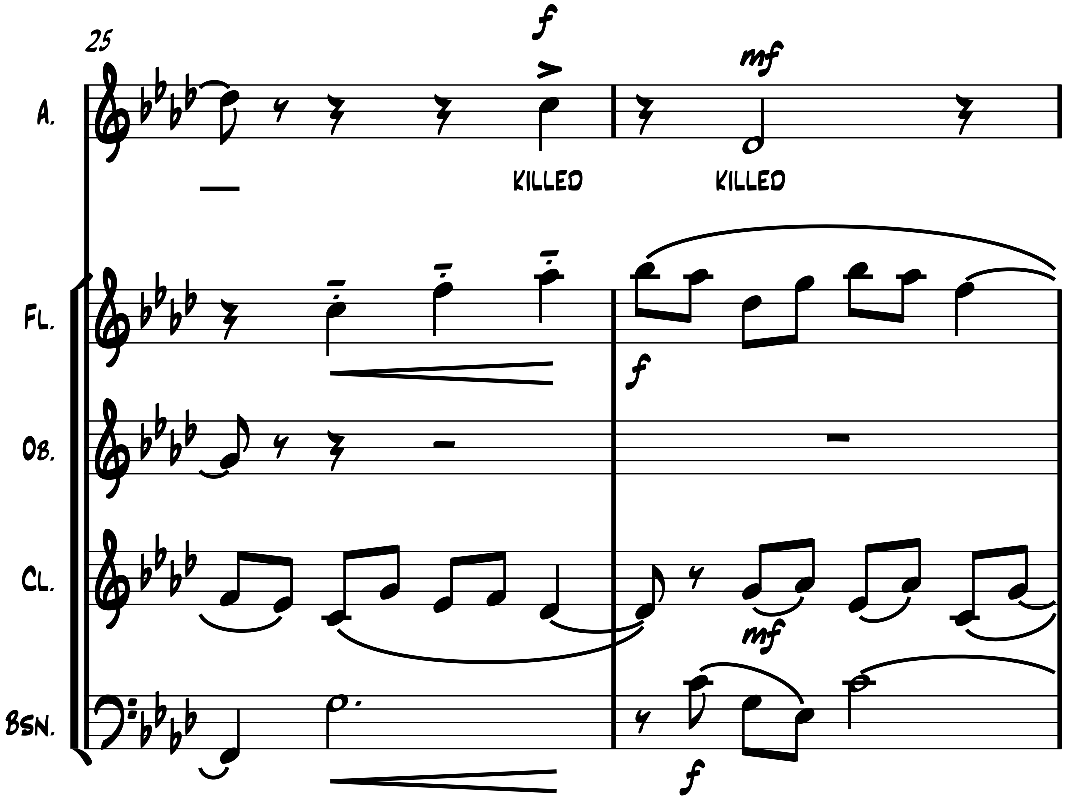
Igor Karaca


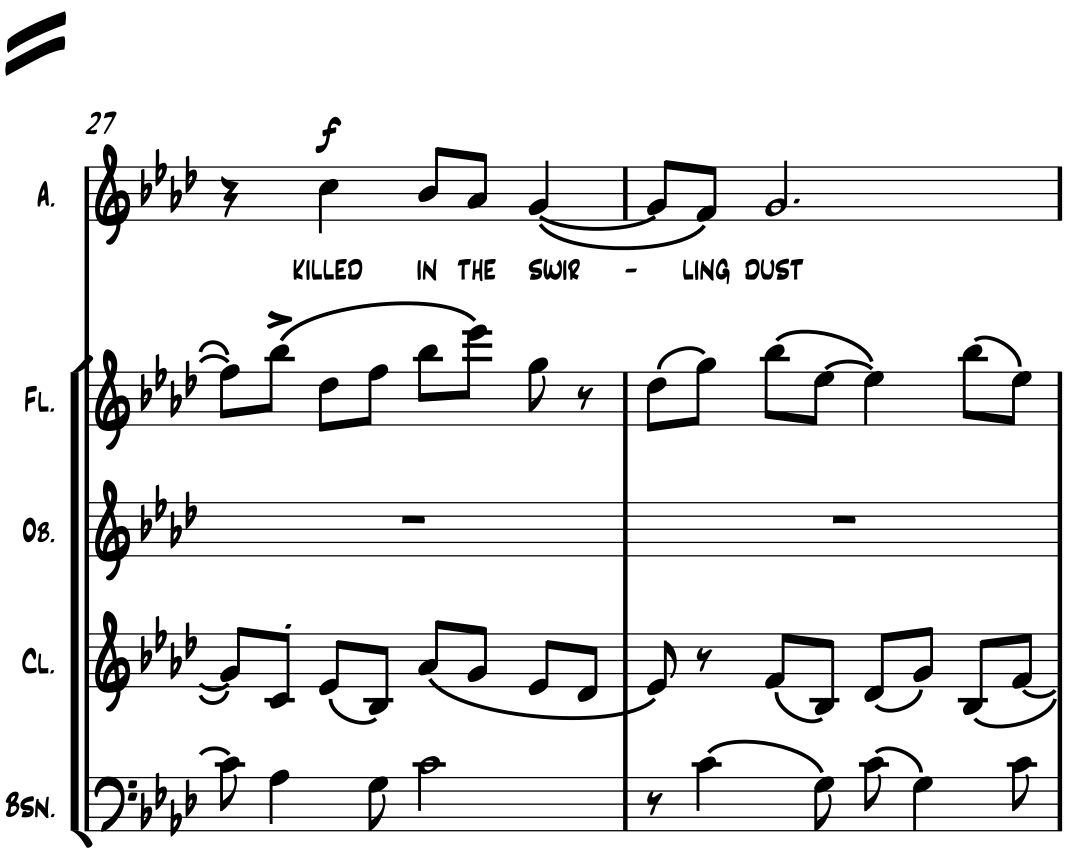
3
A.
Fl.
Ob.
Cl.
p
Bsn.
5
A.
Fl.
Ob.
Cl.
Bsn.


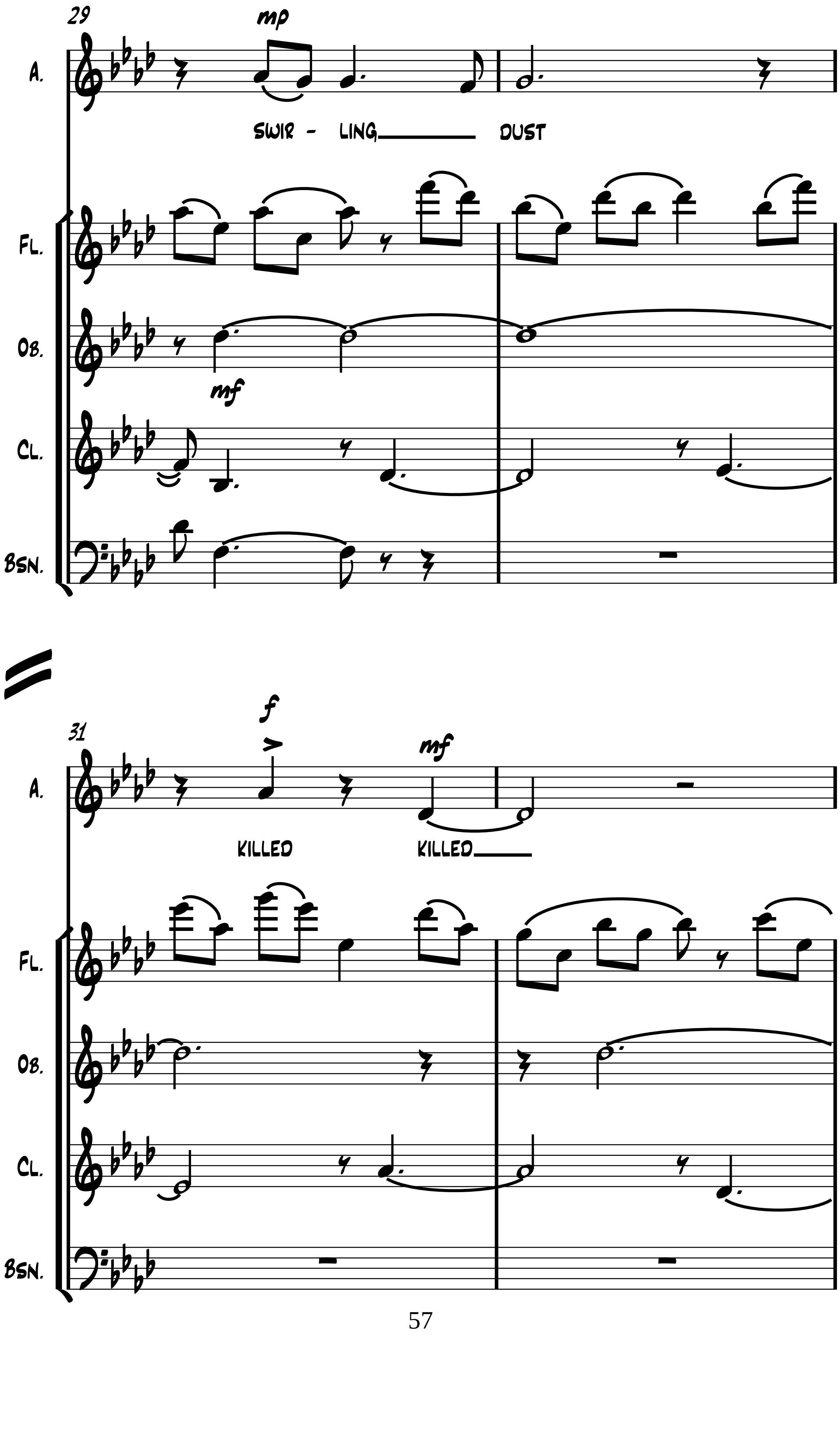
7
A.
Fl.
Ob.
Cl.
Bsn.
mf
mf
cresc.
9
mf
THE SIL - - VER HAND
mf
mf


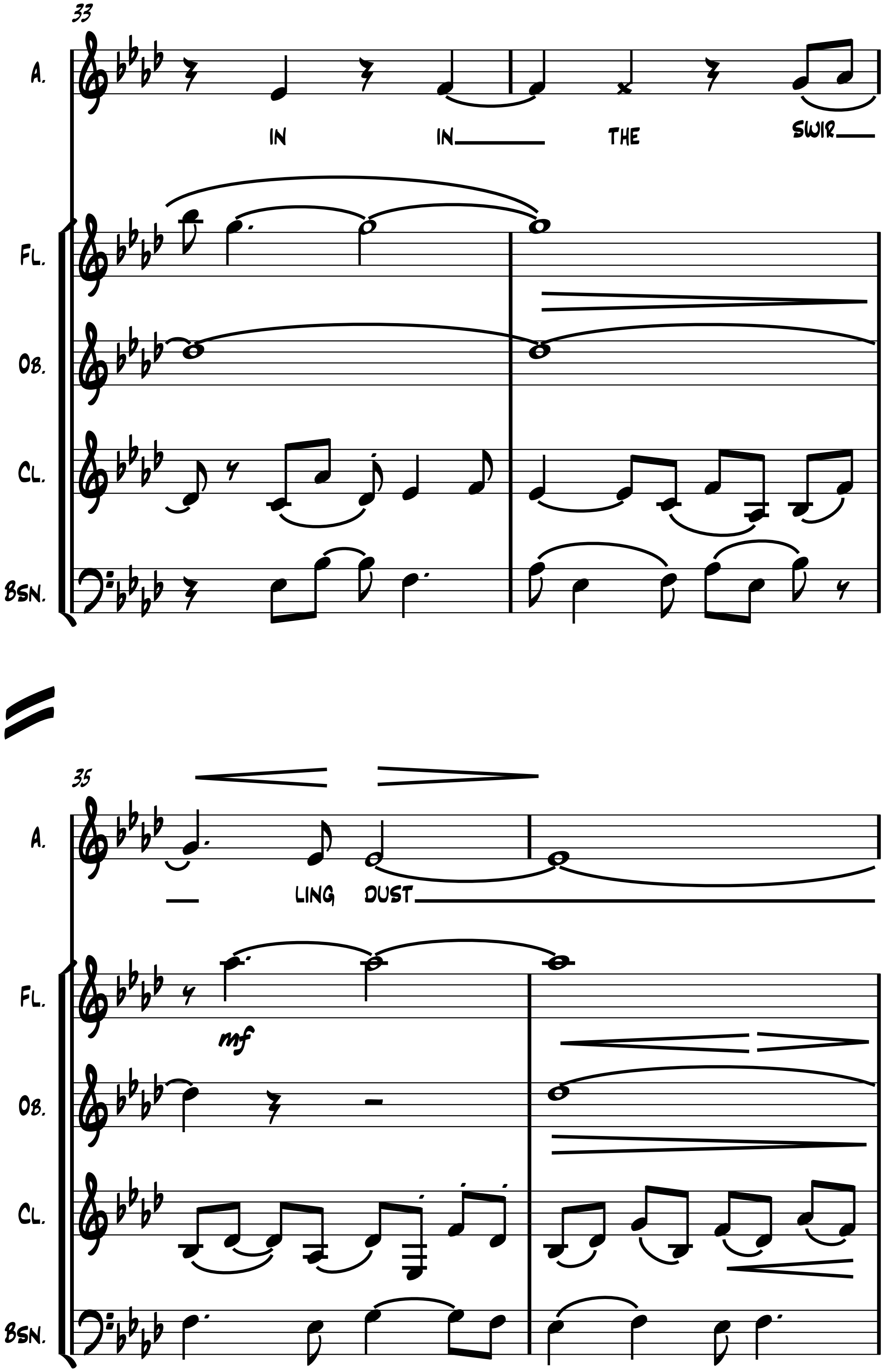
11
A.
SHA___ TERRED THE SKY
Fl.
Ob.
Cl.
Bsn.
13
A.
AN EL - E - GY FOR THE DEATH DEATH
Fl.
Ob.
Cl.
Bsn.


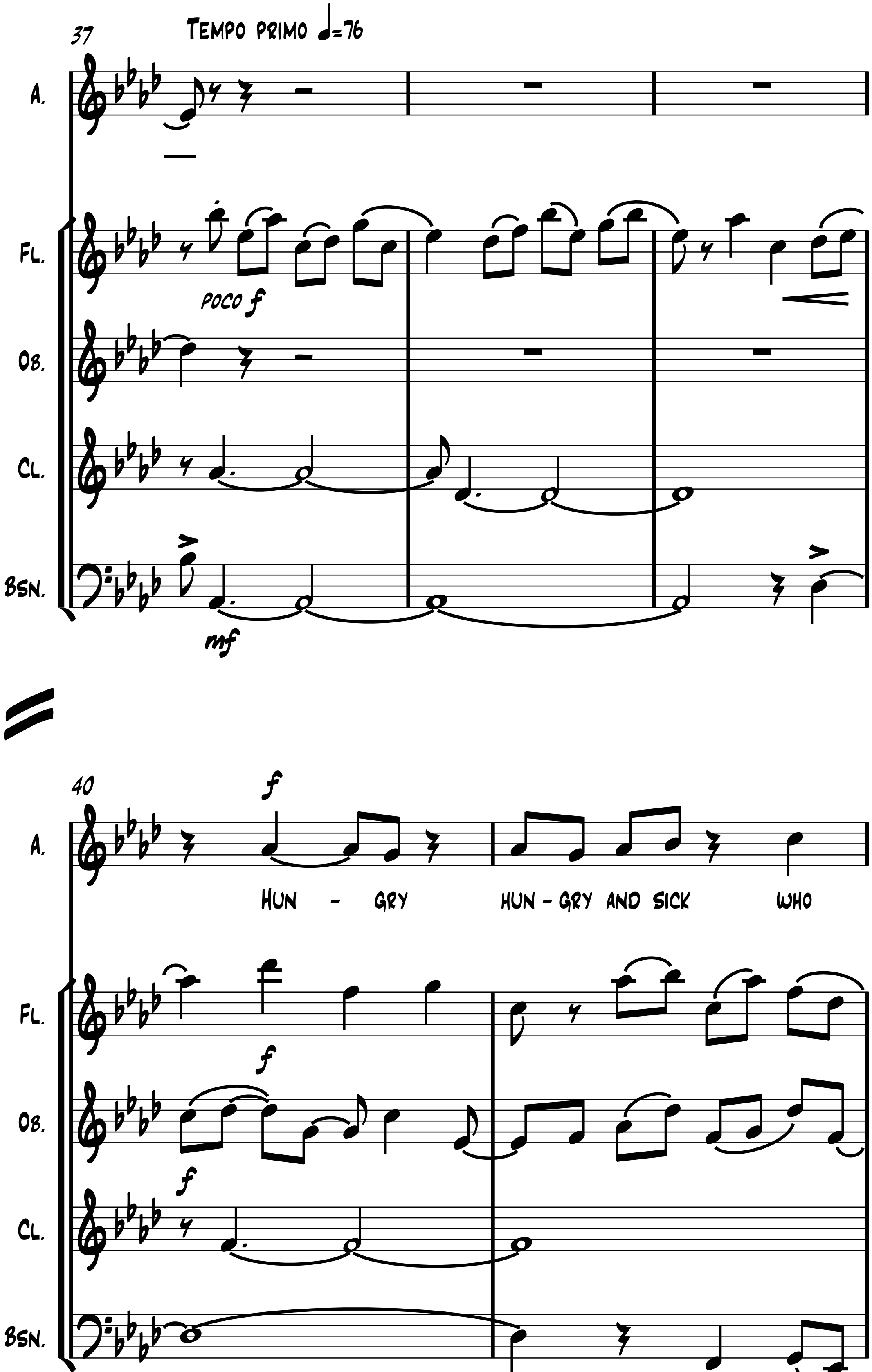
16
A.
DEATH OF PEO - PLE
FL.
OB.
CL.
BSN.
18
POCO RIT.
A.
FL.
OB.
CL.
BSN.


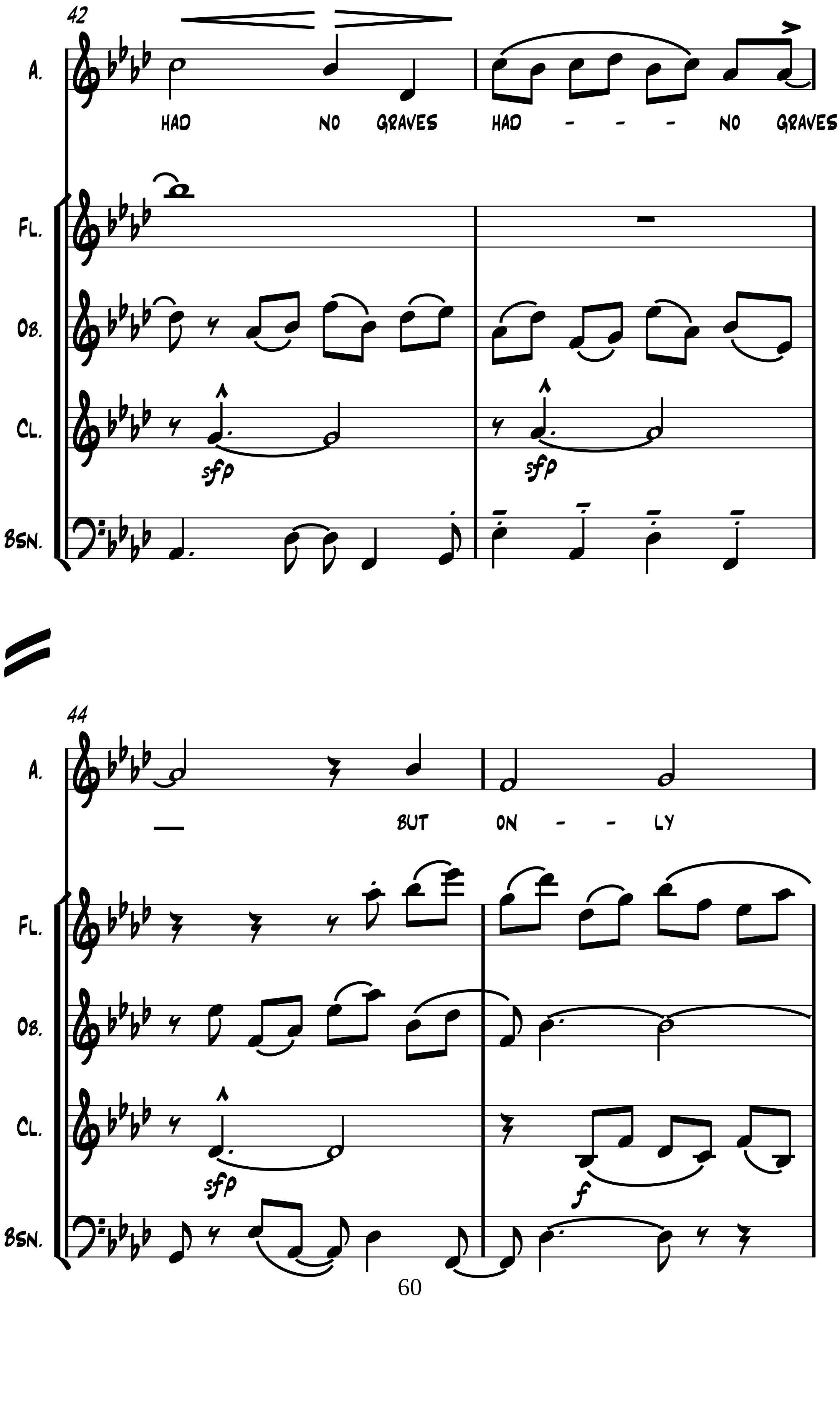
POCO MENO MOSSO ♩=70
21
POCO p
A.
DRI - VEN FROM THEIR HOMES KILLED
FL.
mp
OB.
mp
CL.
mp
BSN.
mp
23
A.
KILLED IN THE SWIR - LING DUST
FL.
OB.
CL.
BSN.


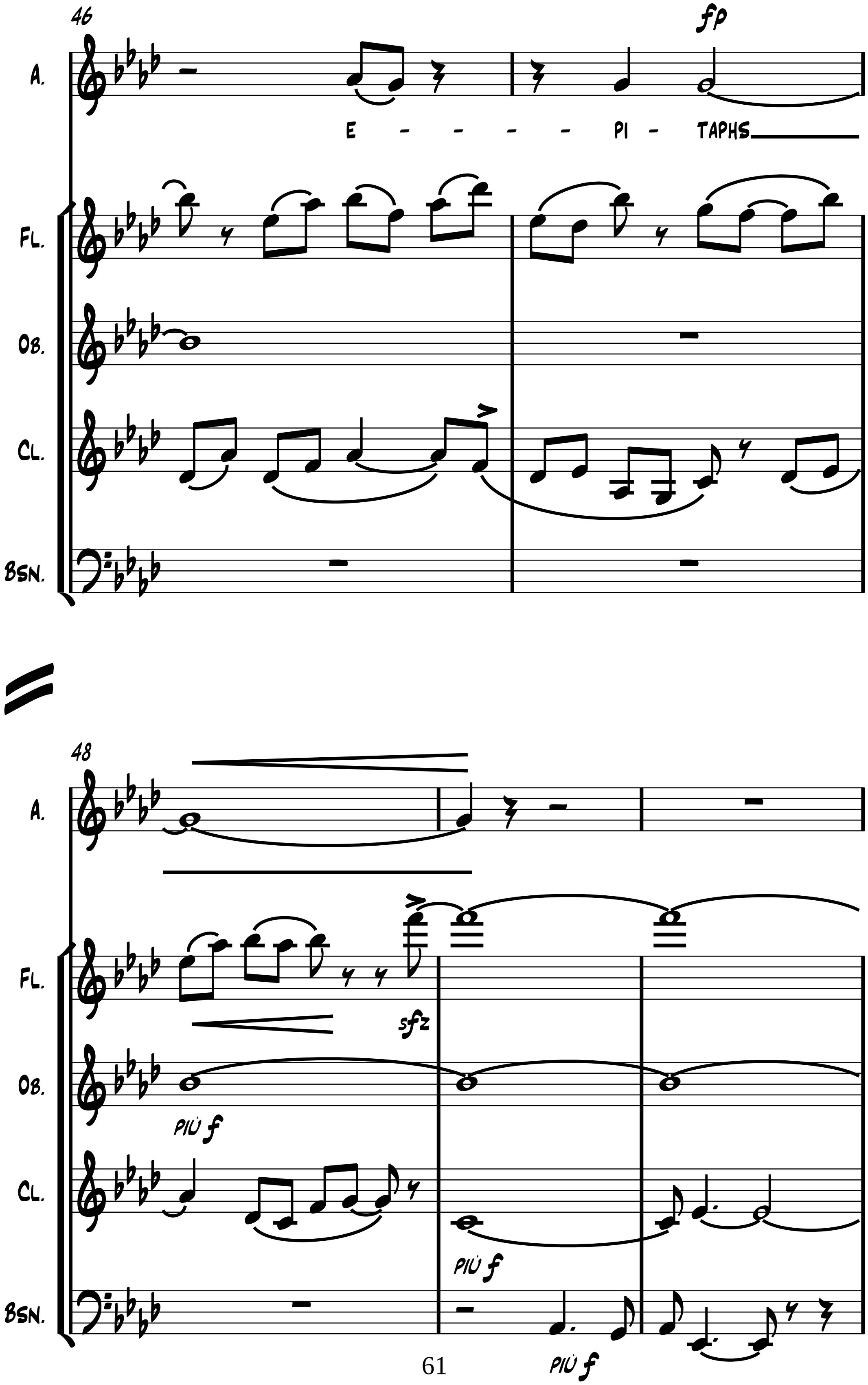
25
A.
Fl.
Ob.
Cl.
Bsn.
f
mf
killed
killed


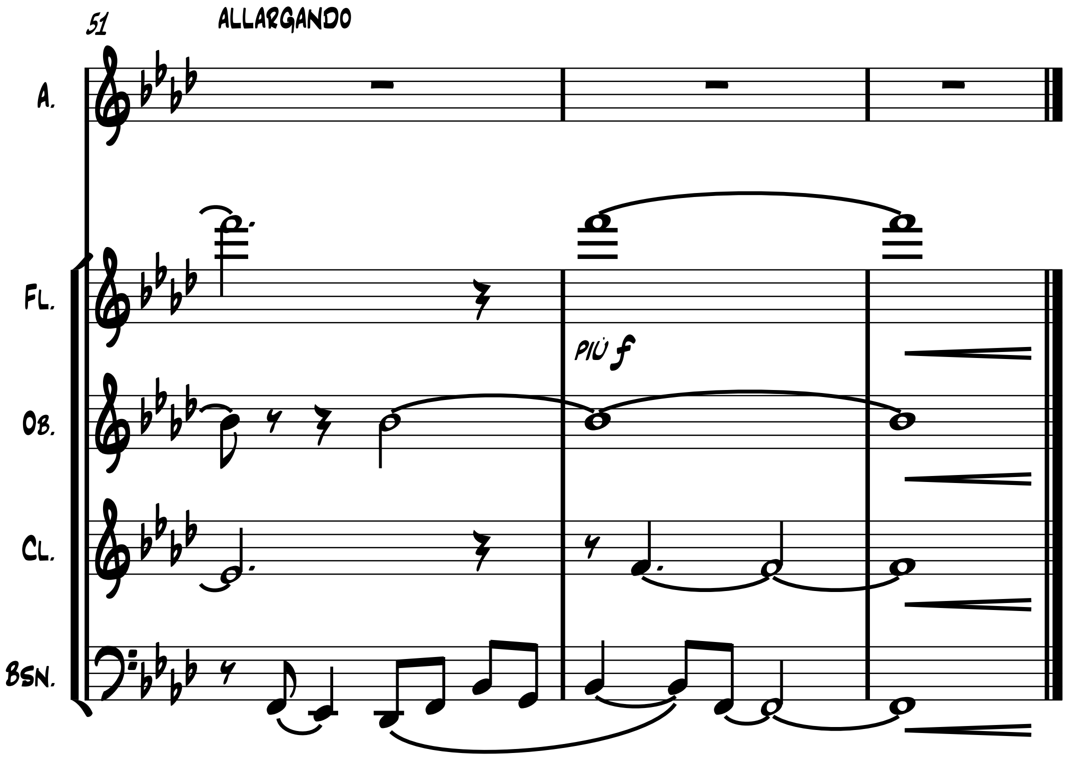
27
A.
Fl.
Ob.
Cl.
Bsn.
f
killed in the swir - ling dust


29
mp
A.
SWIR - LING DUST
Fl.
Ob.
mf
Cl.
Bsn.
f
31
mf
A.
KILLED KILLED
Fl.
Ob.
Cl.
Bsn.


33
A.
IN
IN
THE
SWIR
Fl.
Ob.
Cl.
Bsn.
35
A.
LING
DUST
Fl.
mf
Ob.
Cl.
Bsn.


37
Tempo primo ♩=76
A.
Fl.
poco f
Ob.
Cl.
Bsn.
mf
40
A.
f
Hun - gry
hun - gry and sick
who
Fl.
Ob.
f
Cl.
f
Bsn.


42
A.
HAD NO GRAVES HAD - - - NO GRAVES
Fl.
Ob.
Cl.
sfp
sfp
Bsn.
44
A.
BUT ON - - LY
Fl.
Ob.
Cl.
sfp
f
Bsn.


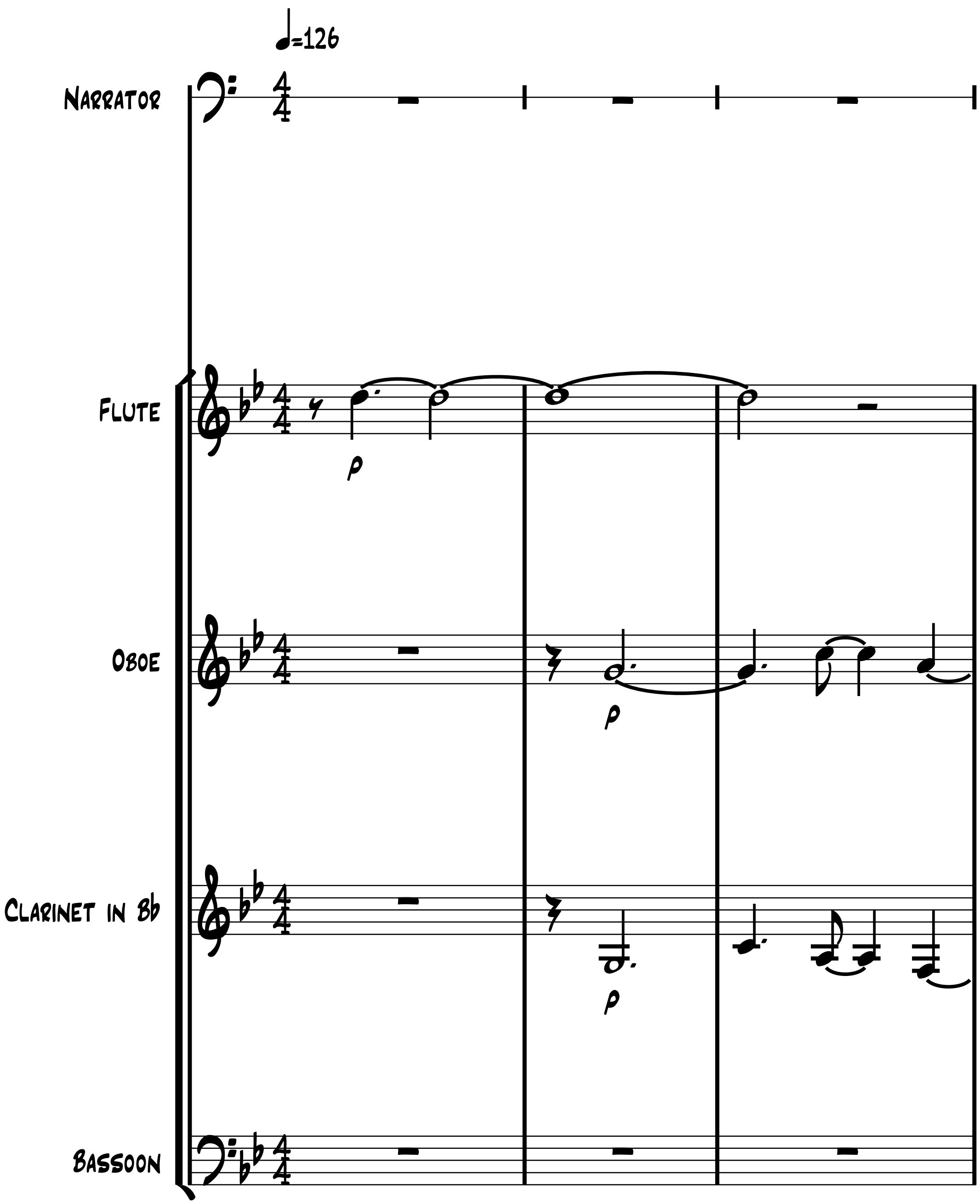
46
fp
A.
E - - - - - PI - TAPHS
Fl.
Ob.
Cl.
Bsn.
48
A.
Fl.
sfz
Ob.
PIÙ f
Cl.
PIÙ f
Bsn.
PIÙ f


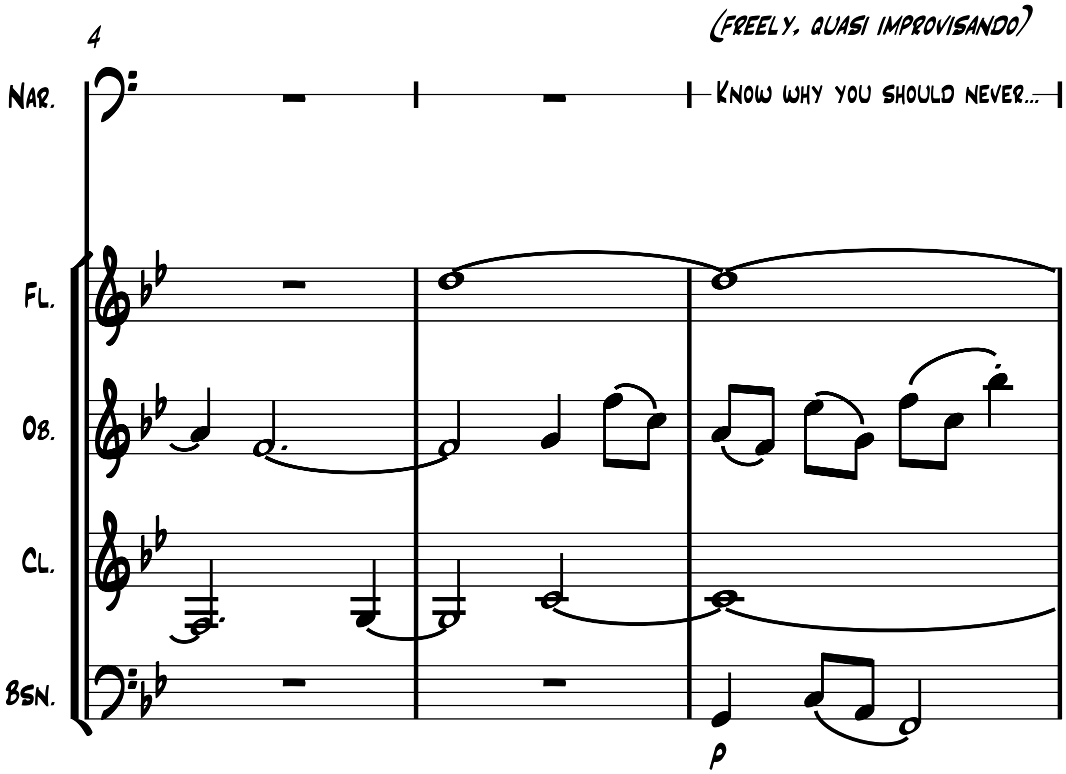
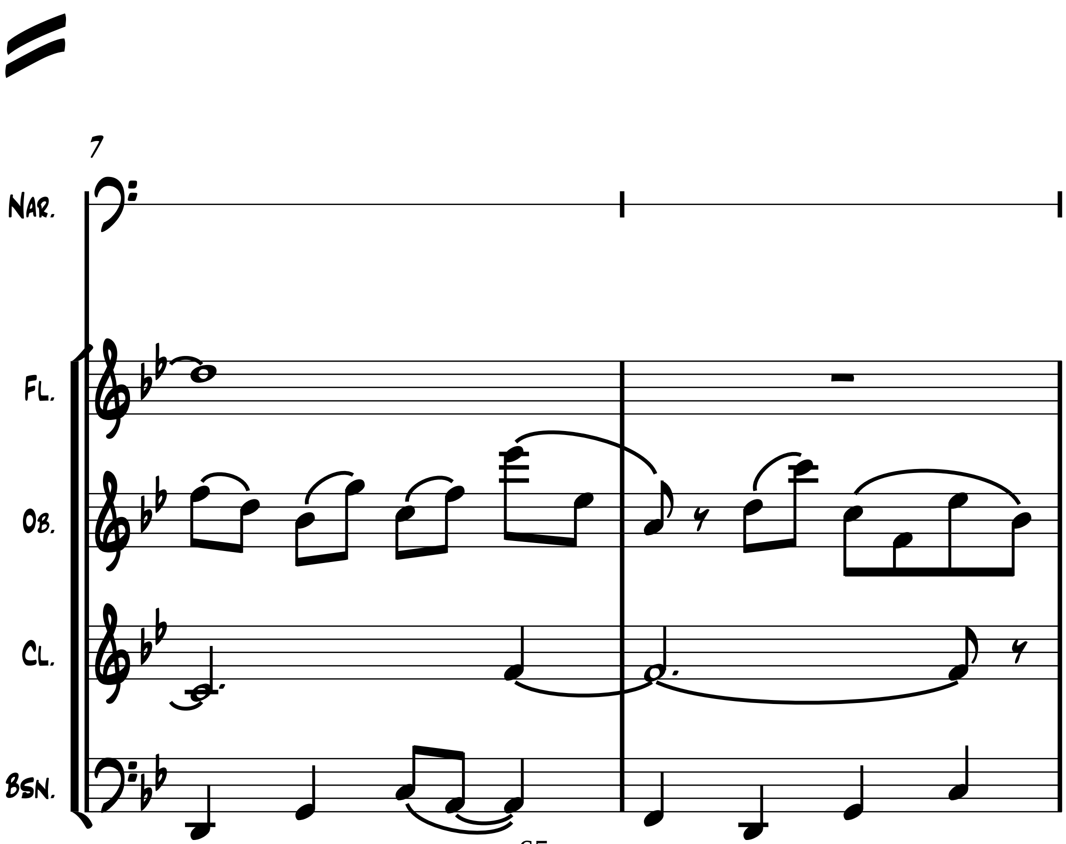
51
Allargando
A.
Fl.
Ob.
Cl.
Bsn.
più f

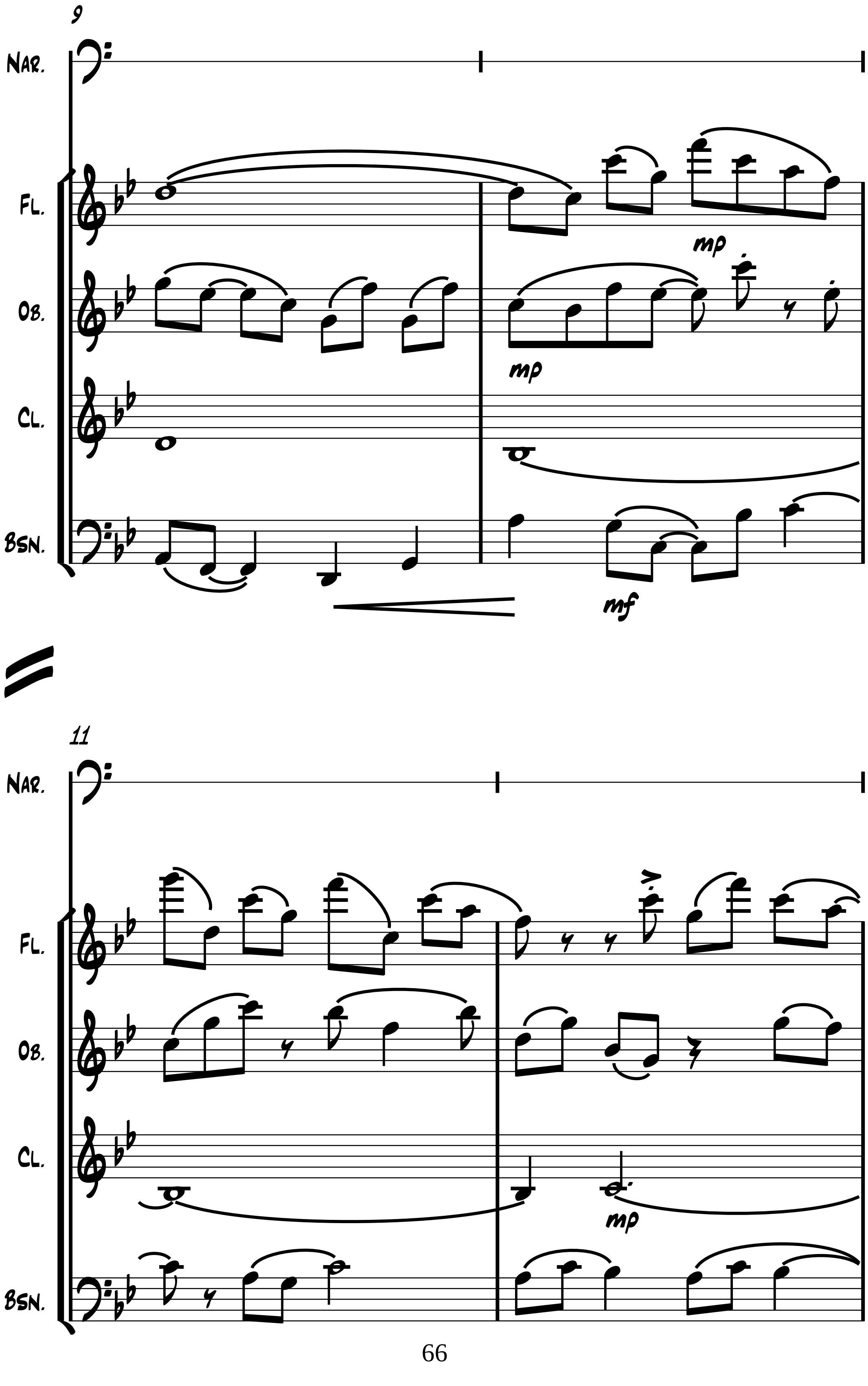
APPENDIX B

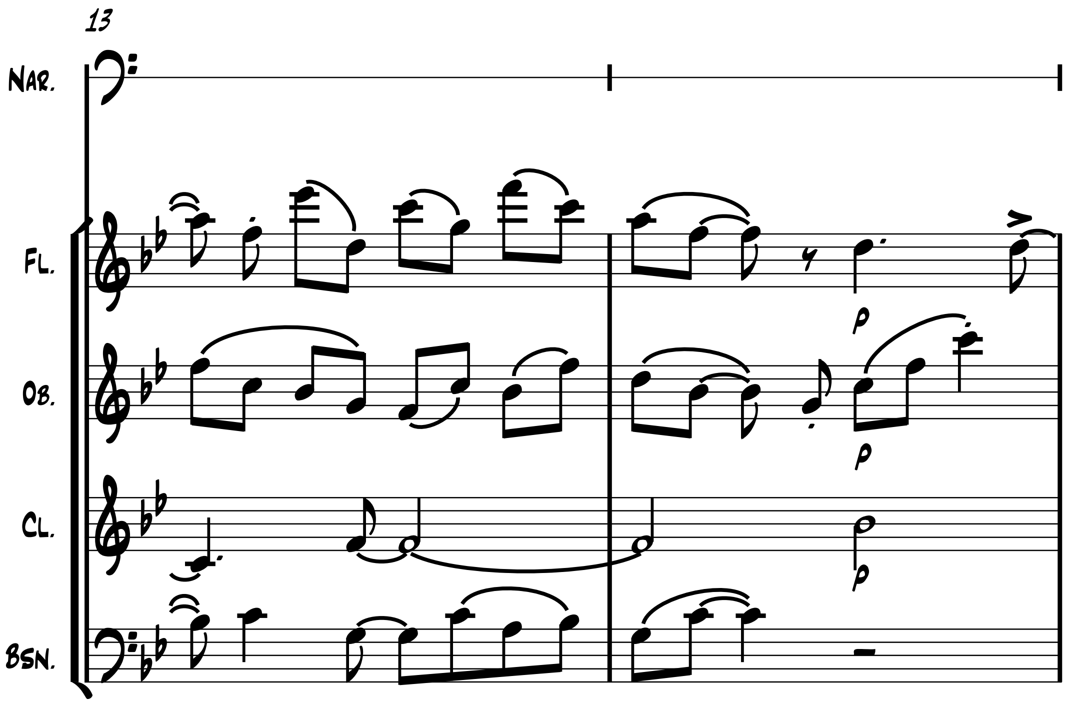
…she turned from Ares her bright eyes, Movement 2

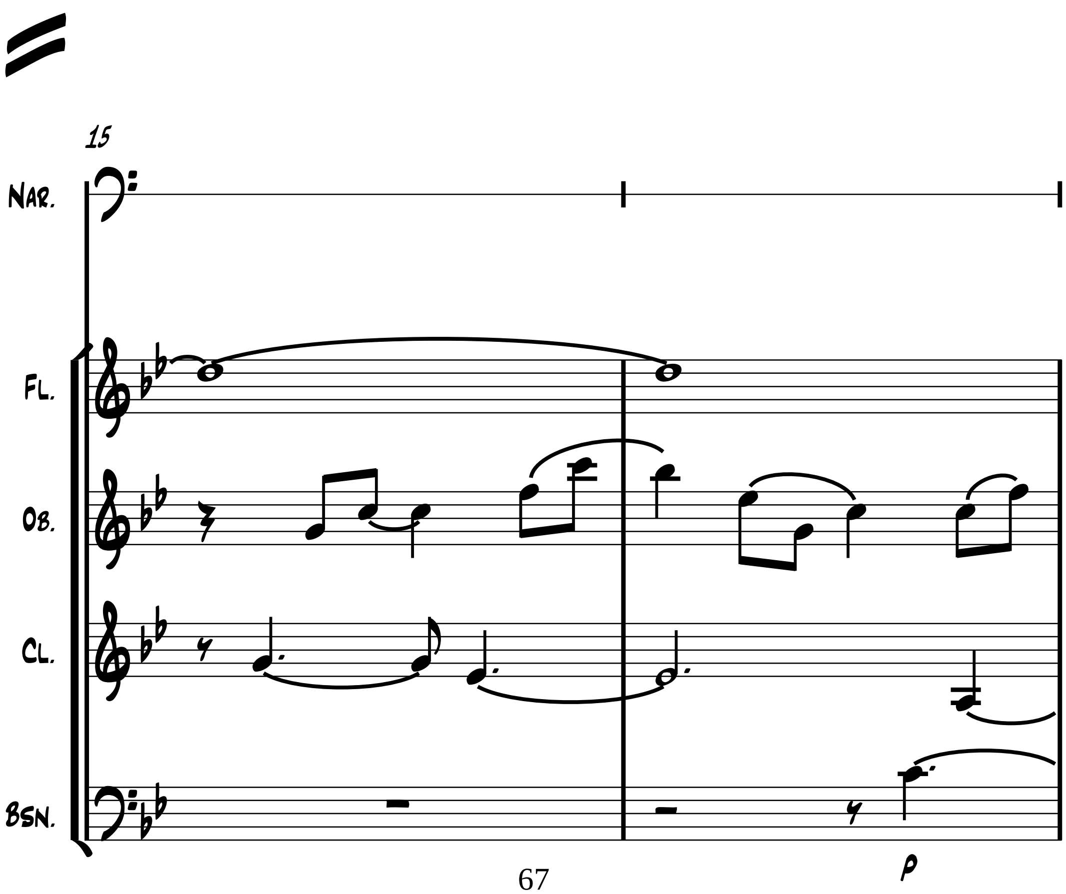
# 2.

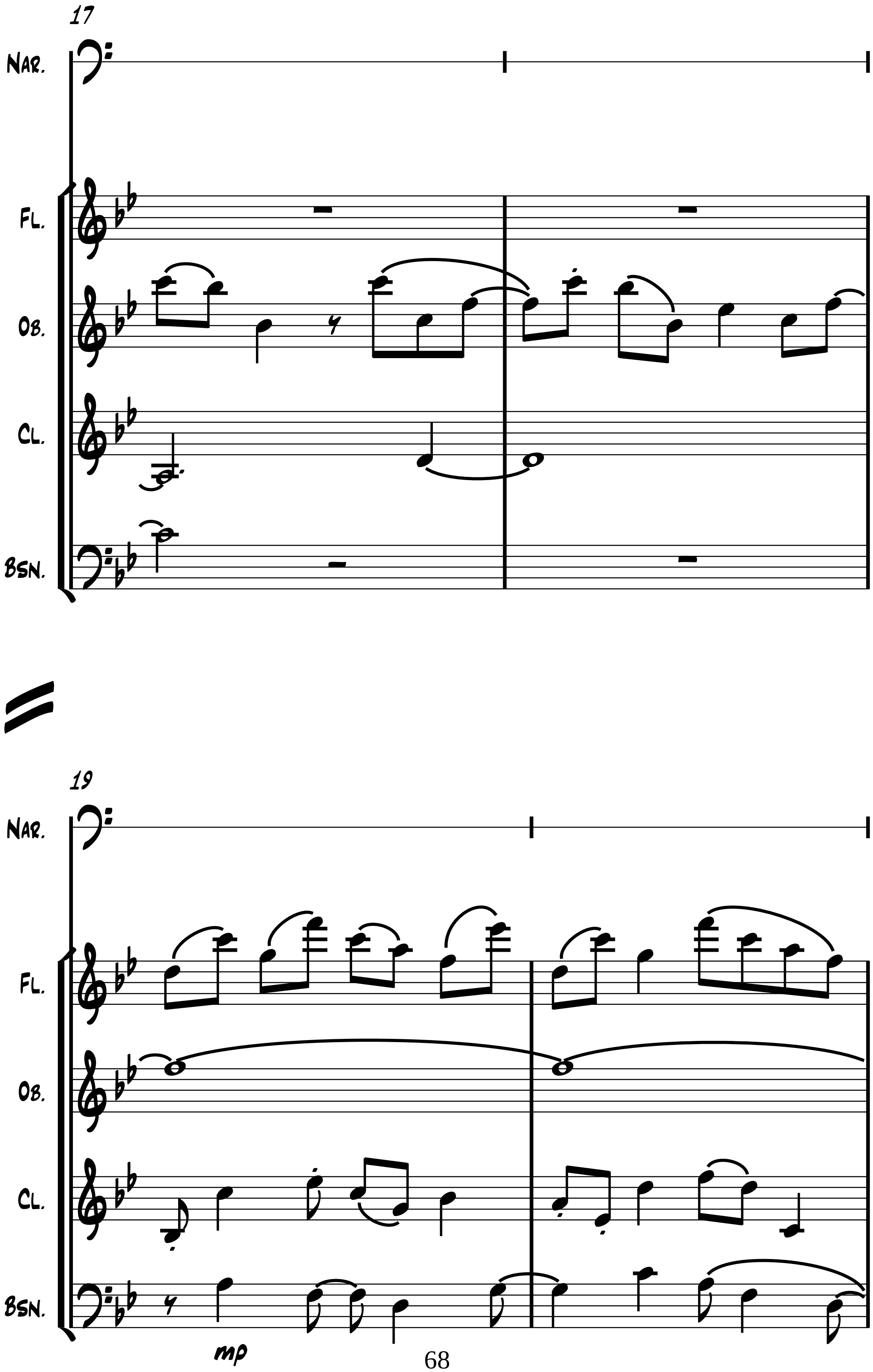


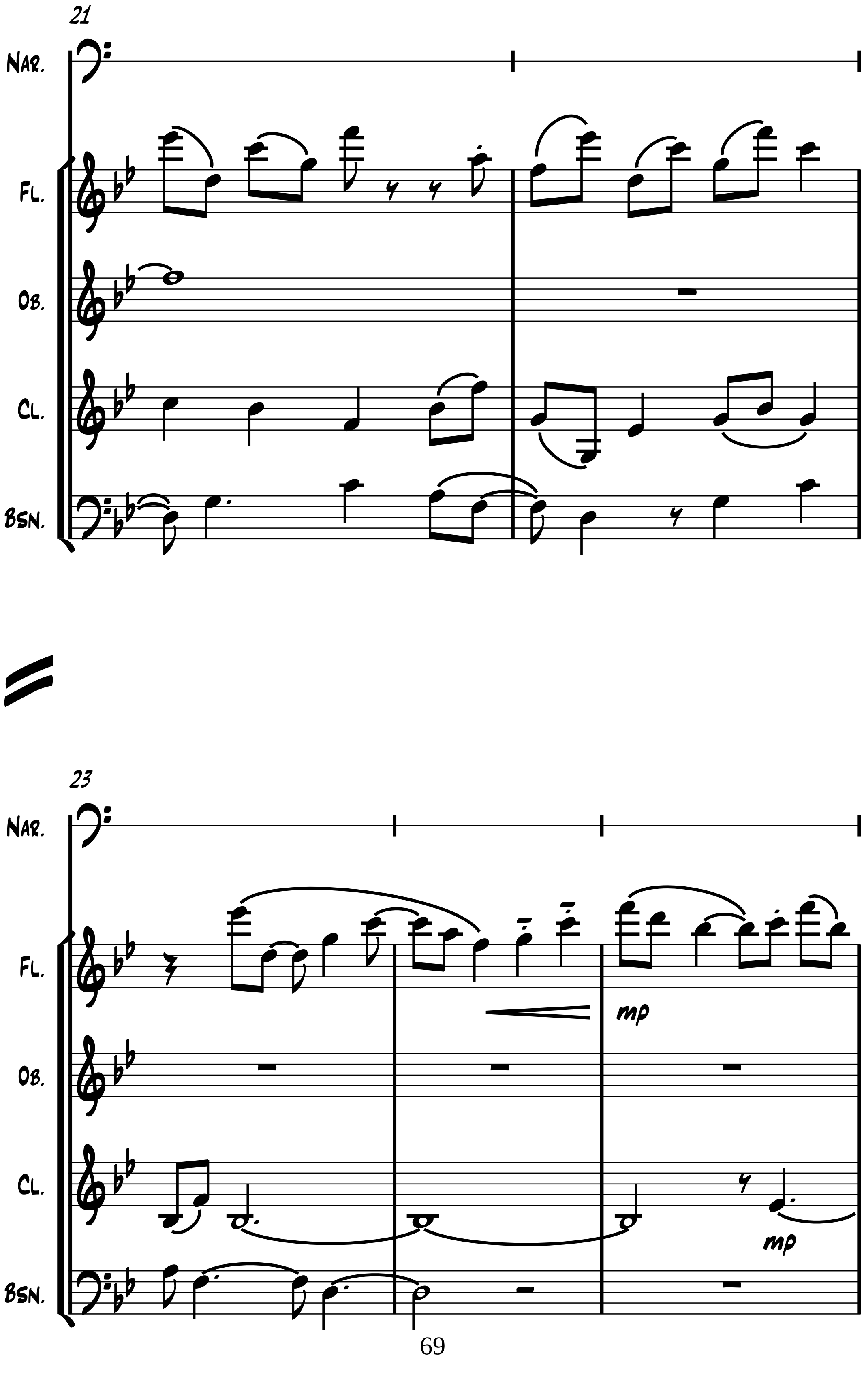

(freely, quasi improvisando)
4
Nar.
Fl.
Ob.
Cl.
Bsn.
Know why you should never...
p


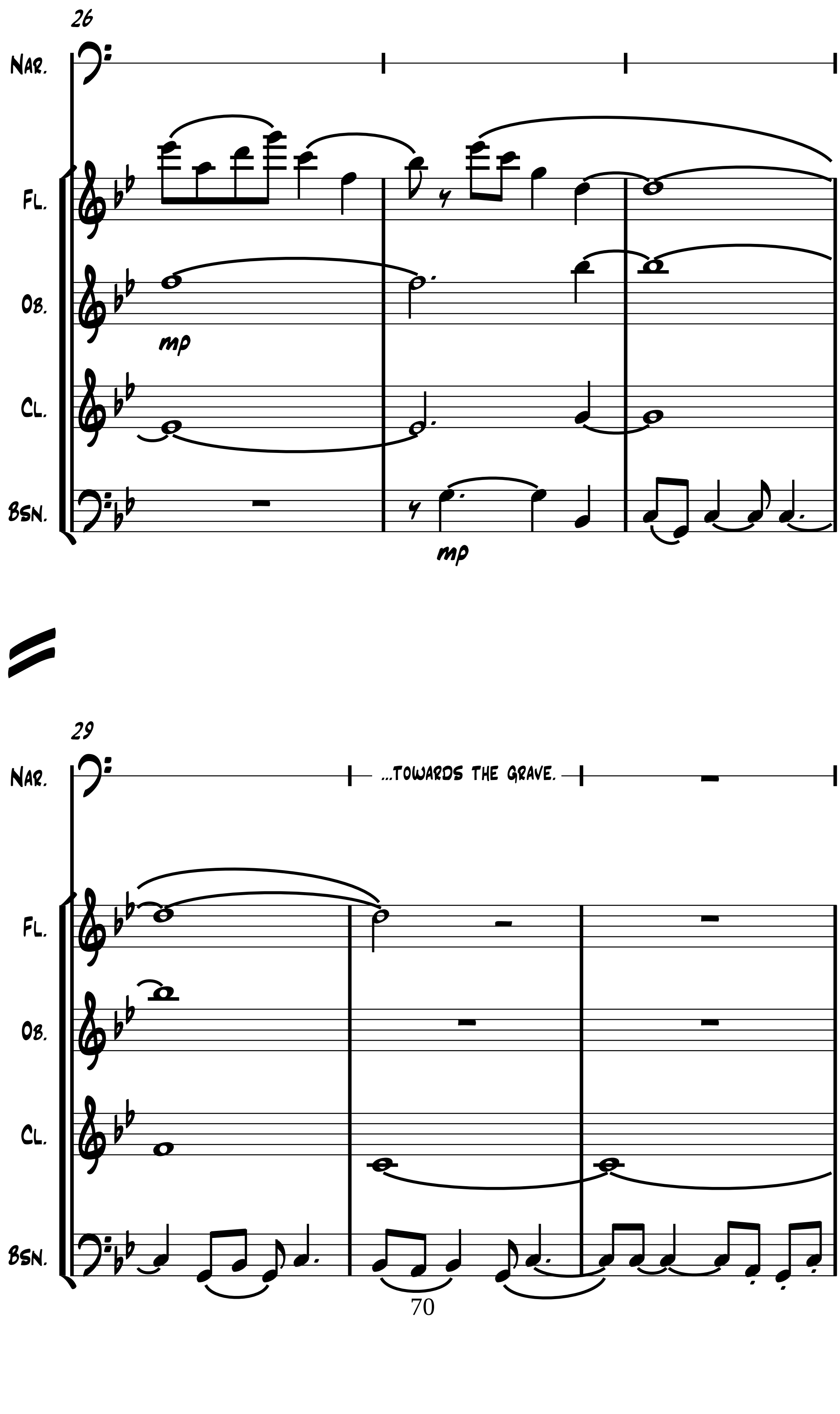
7
Nar.
Fl.
Ob.
Cl.
Bsn.


9
Nar.
Fl.
Ob.
Cl.
Bsn.
mp
mp
mf
11
Nar.
Fl.
Ob.
Cl.
Bsn.
mp


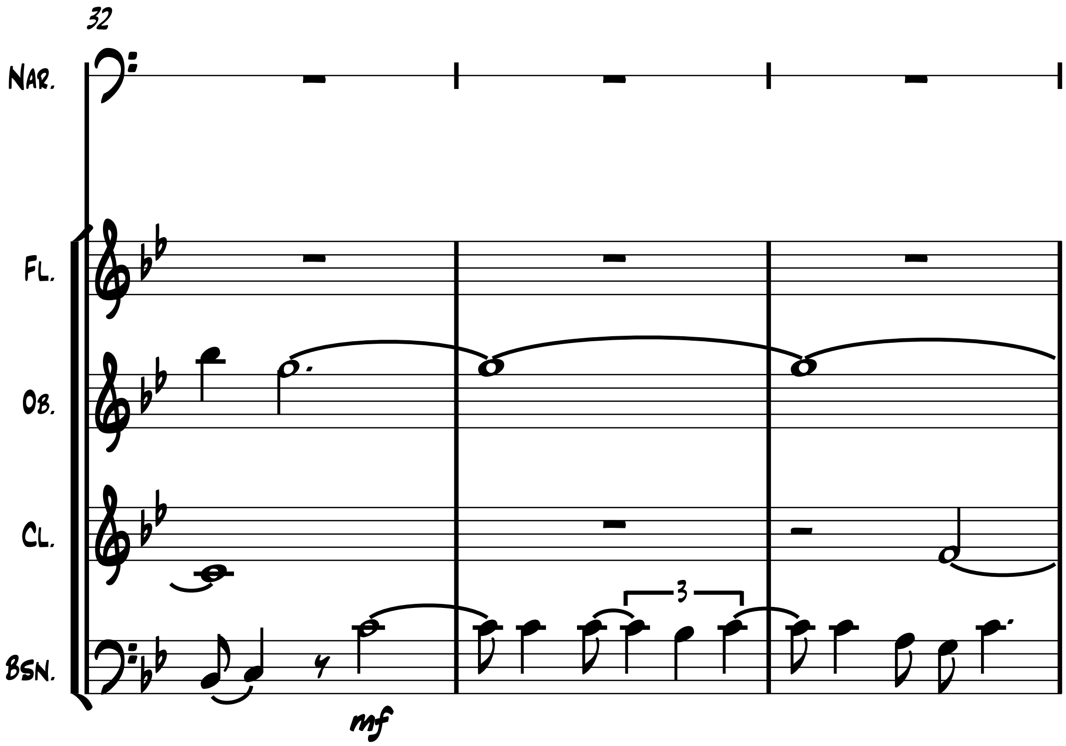
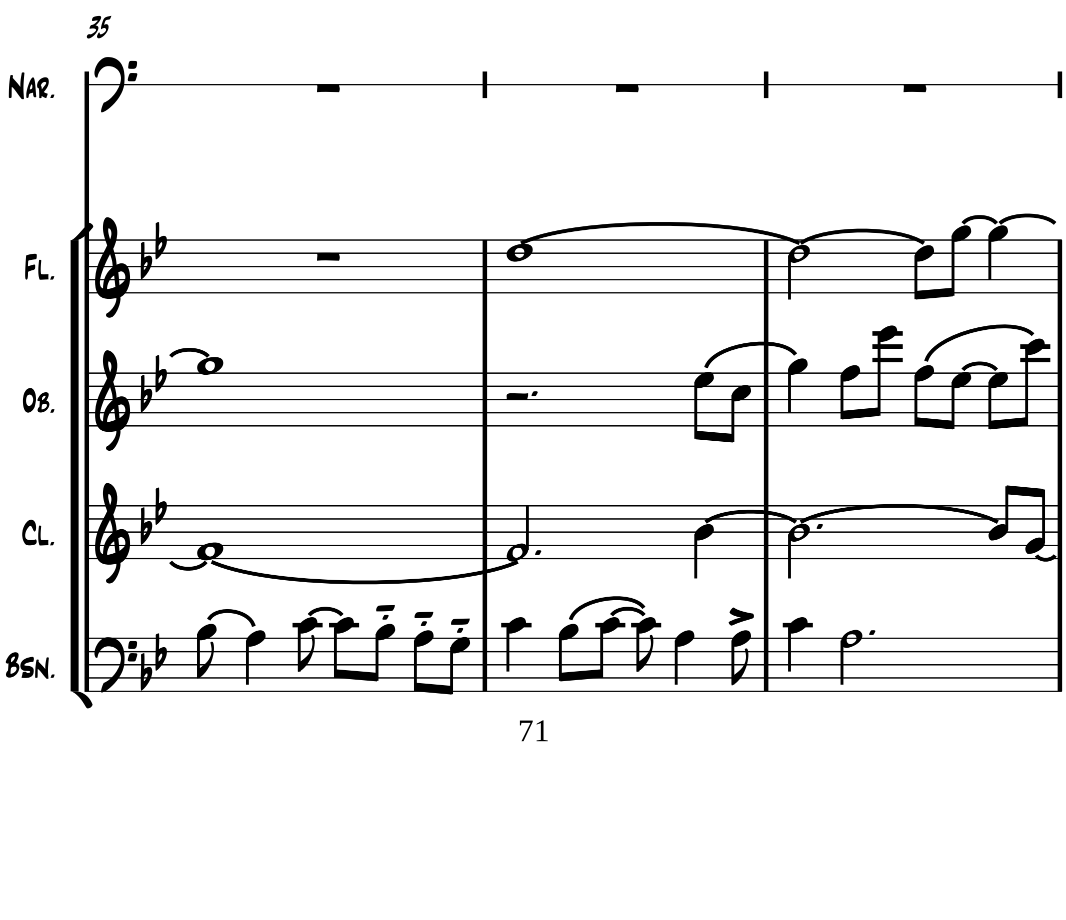
13
Nar.
Fl.
Ob.
Cl.
Bsn.
p
p
p


15
Nar.
Fl.
Ob.
Cl.
Bsn.
p


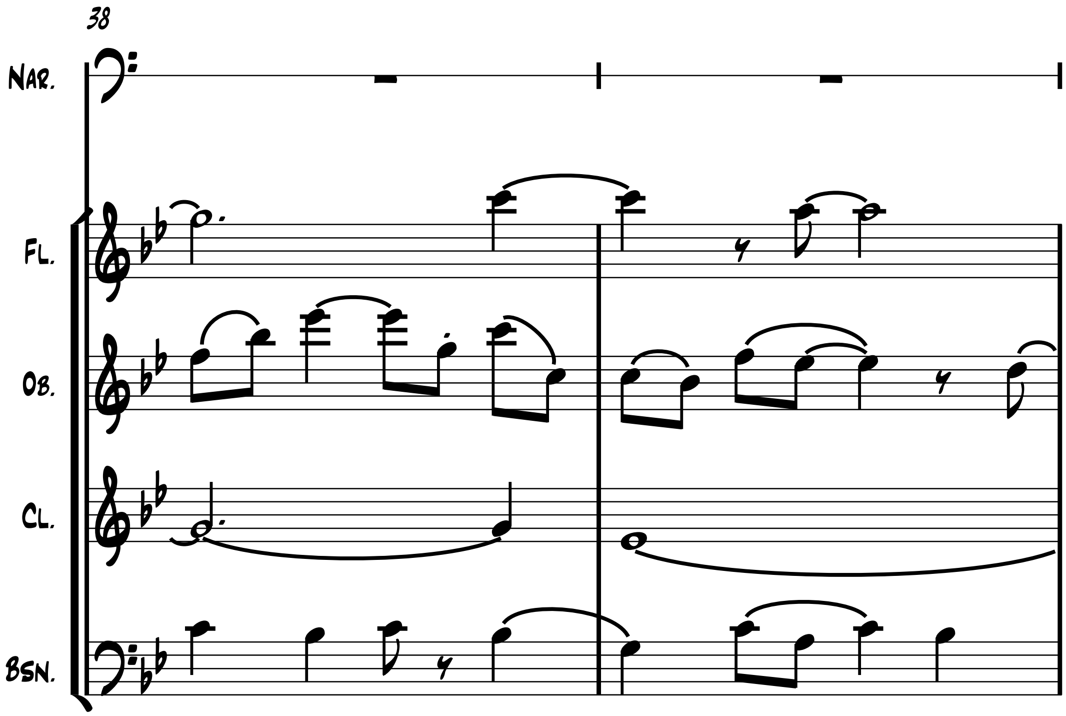
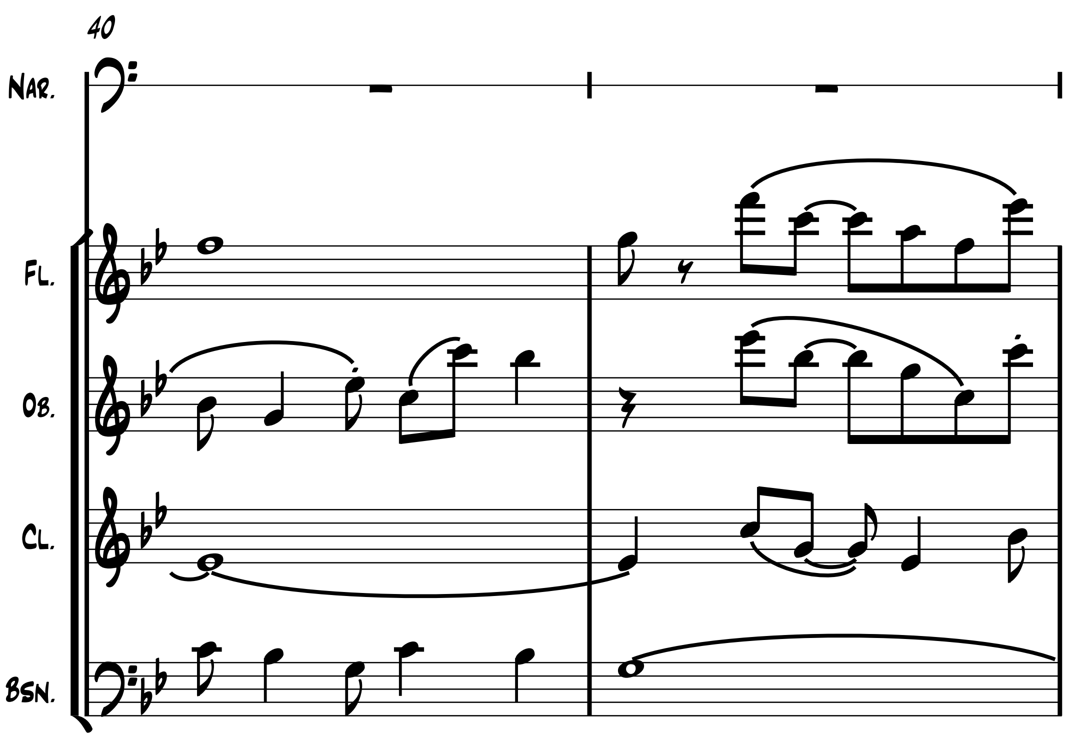
17
Nar.
Fl.
Ob.
Cl.
Bsn.
19
Nar.
Fl.
Ob.
Cl.
Bsn.
mp


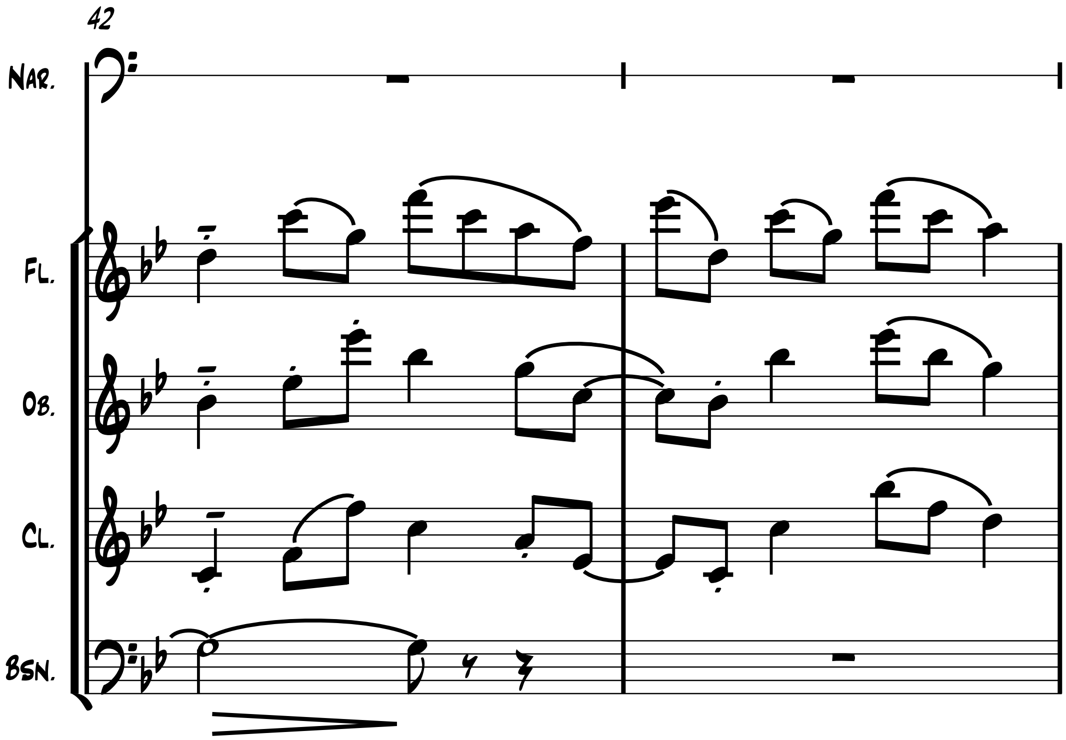
21
Nar.
Fl.
Ob.
Cl.
Bsn.
23
Nar.
Fl.
mp
Ob.
Cl.
mp
Bsn.


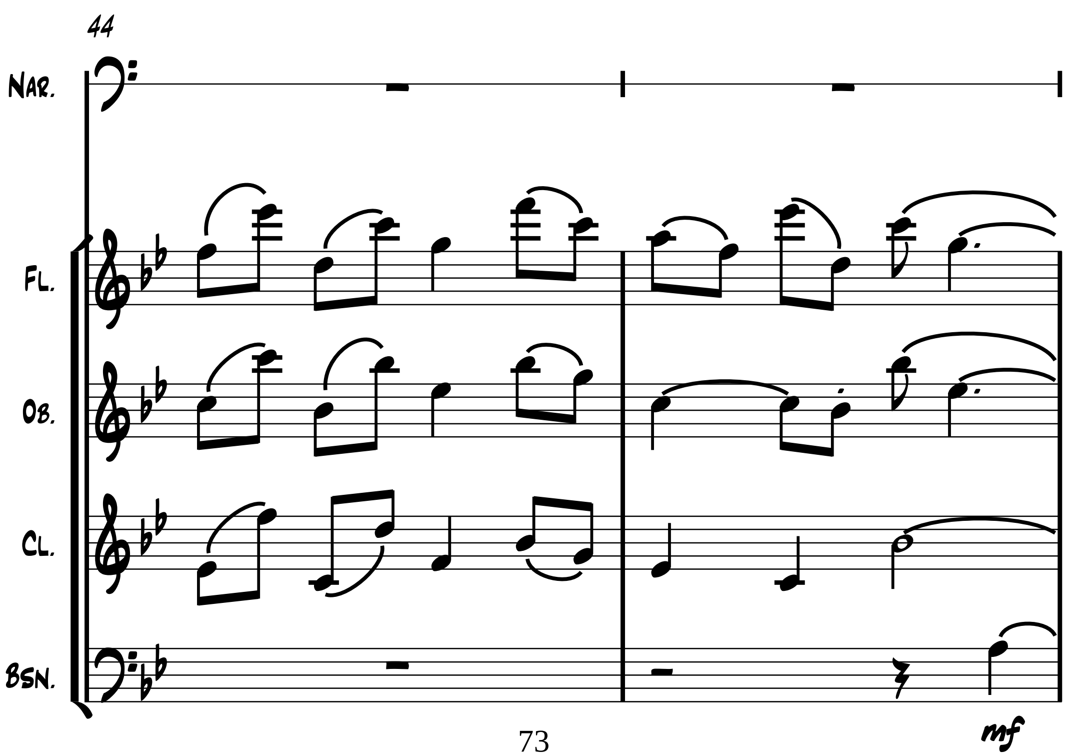
26
Nar.
Fl.
Ob.
mp
Cl.
Bsn.
mp
29
Nar.
...towards the grave.
Fl.
Ob.
Cl.
Bsn.

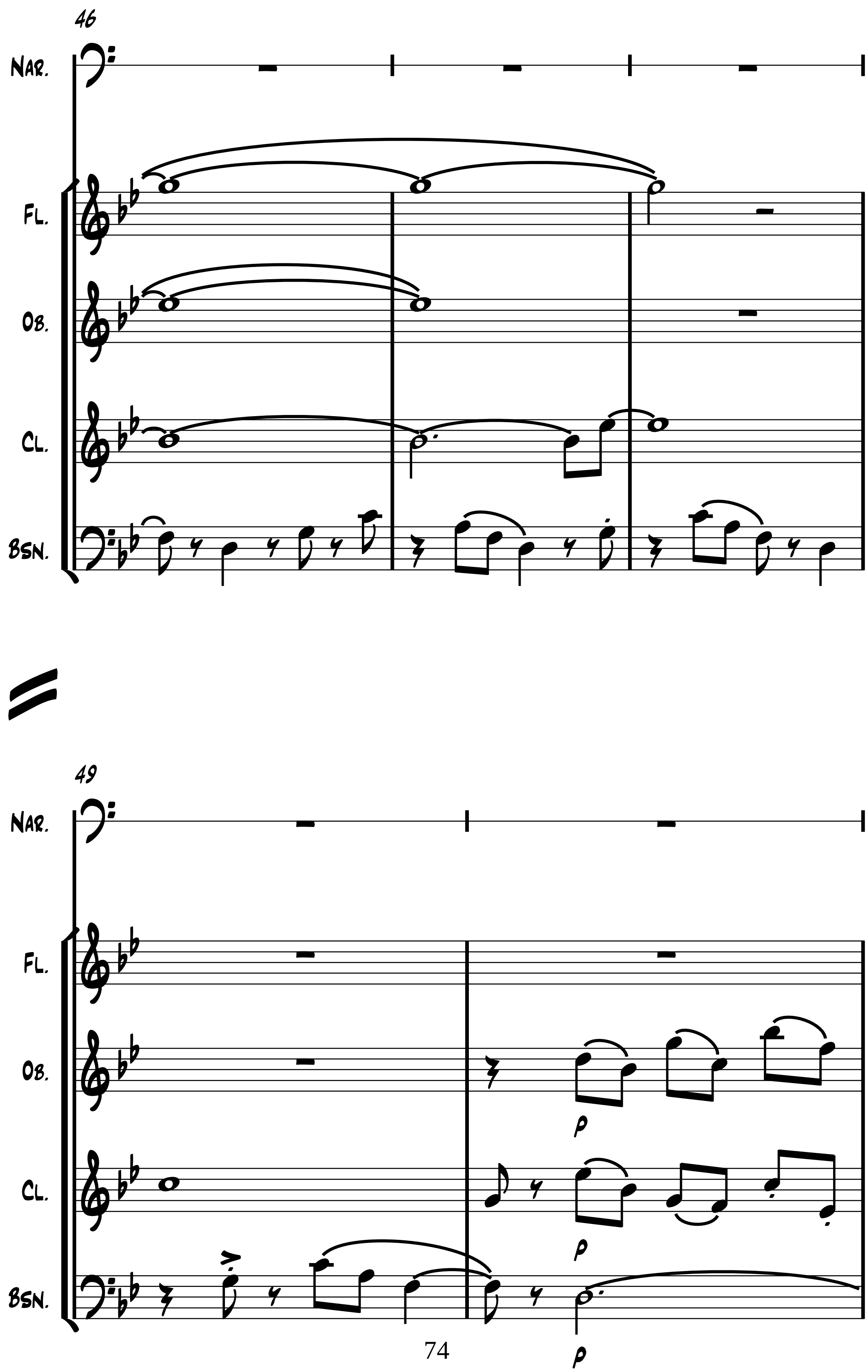

32
Nar.
Fl.
Ob.
Cl.
Bsn.
3
mf


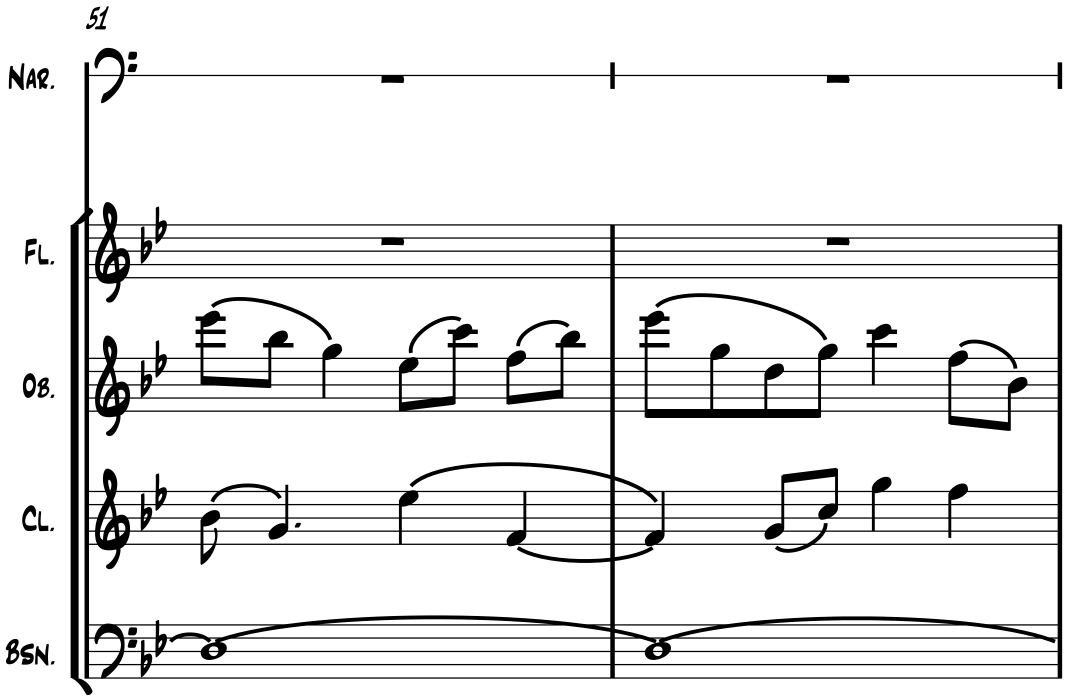
35
Nar.
Fl.
Ob.
Cl.
Bsn.

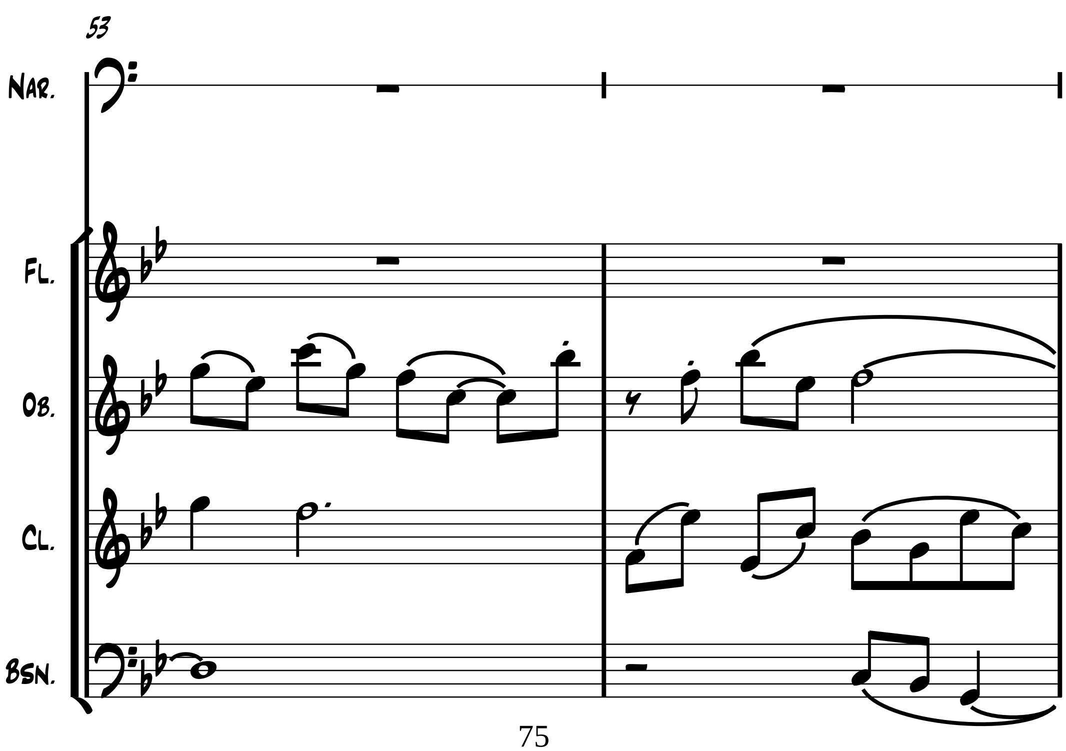

38
Nar.
Fl.
Ob.
Cl.
Bsn.


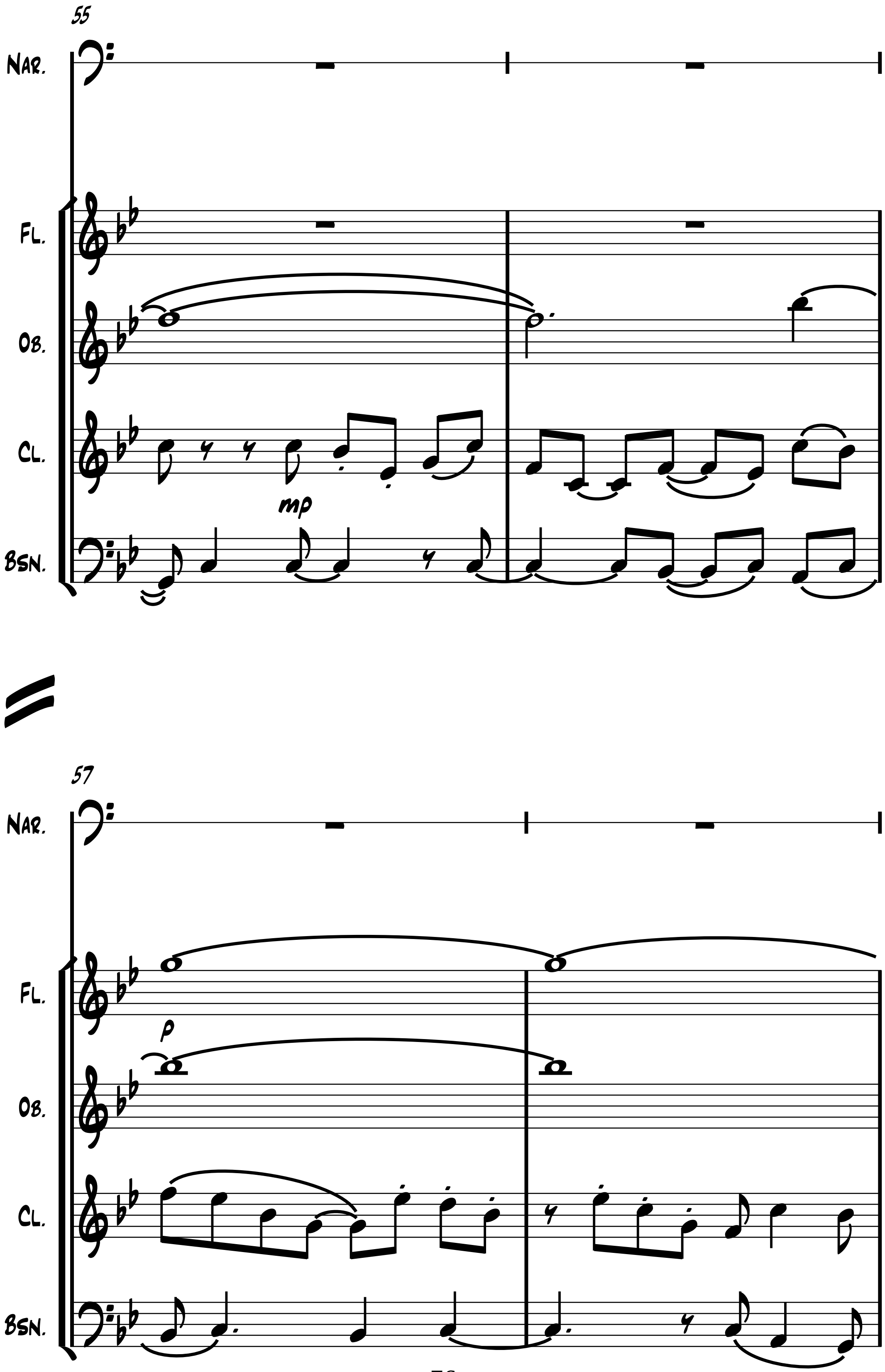
40
Nar.
Fl.
Ob.
Cl.
Bsn.


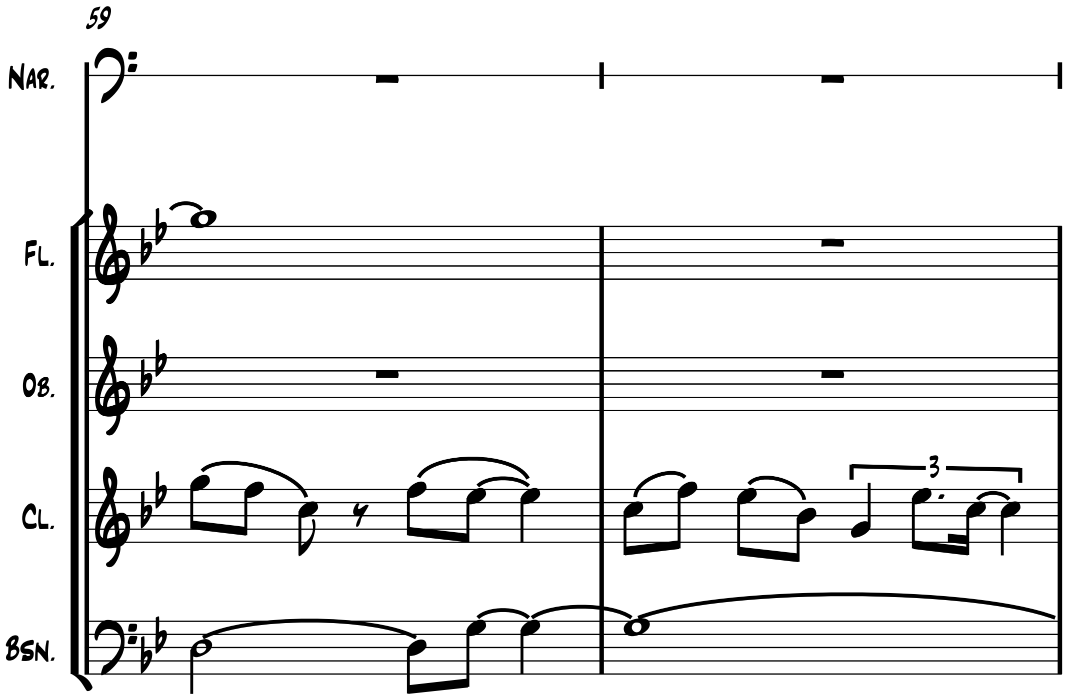
42
Nar.
Fl.
Ob.
Cl.
Bsn.


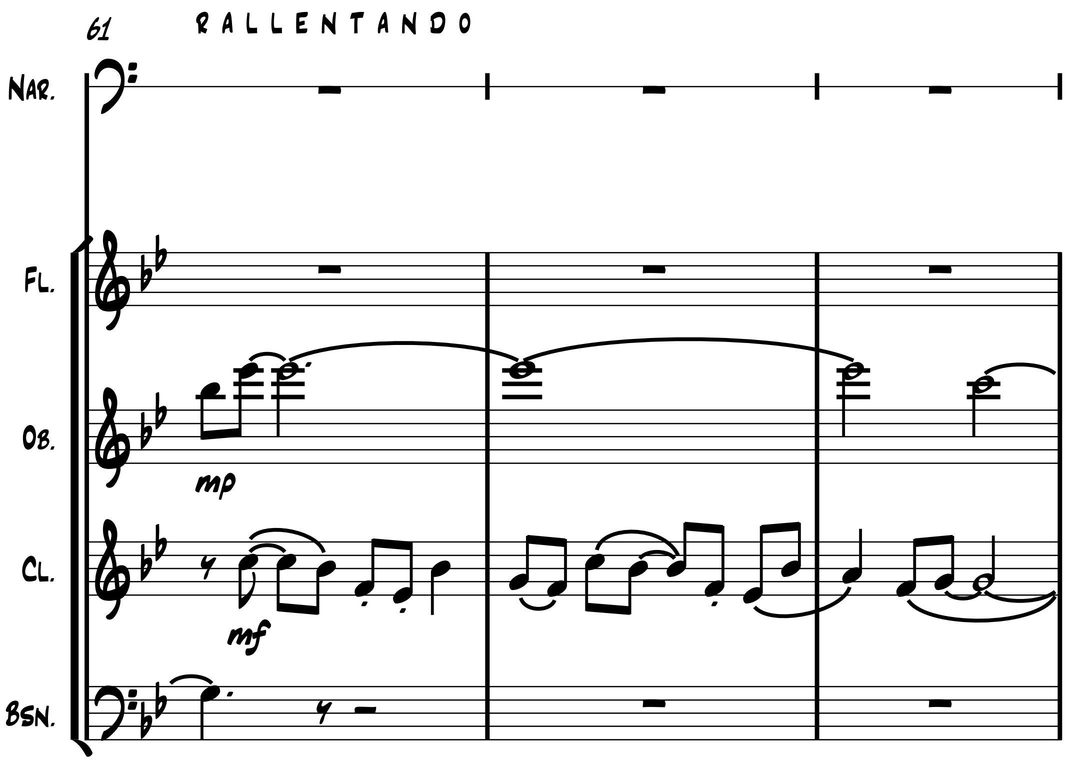
44
Nar.
Fl.
Ob.
Cl.
Bsn.
mf


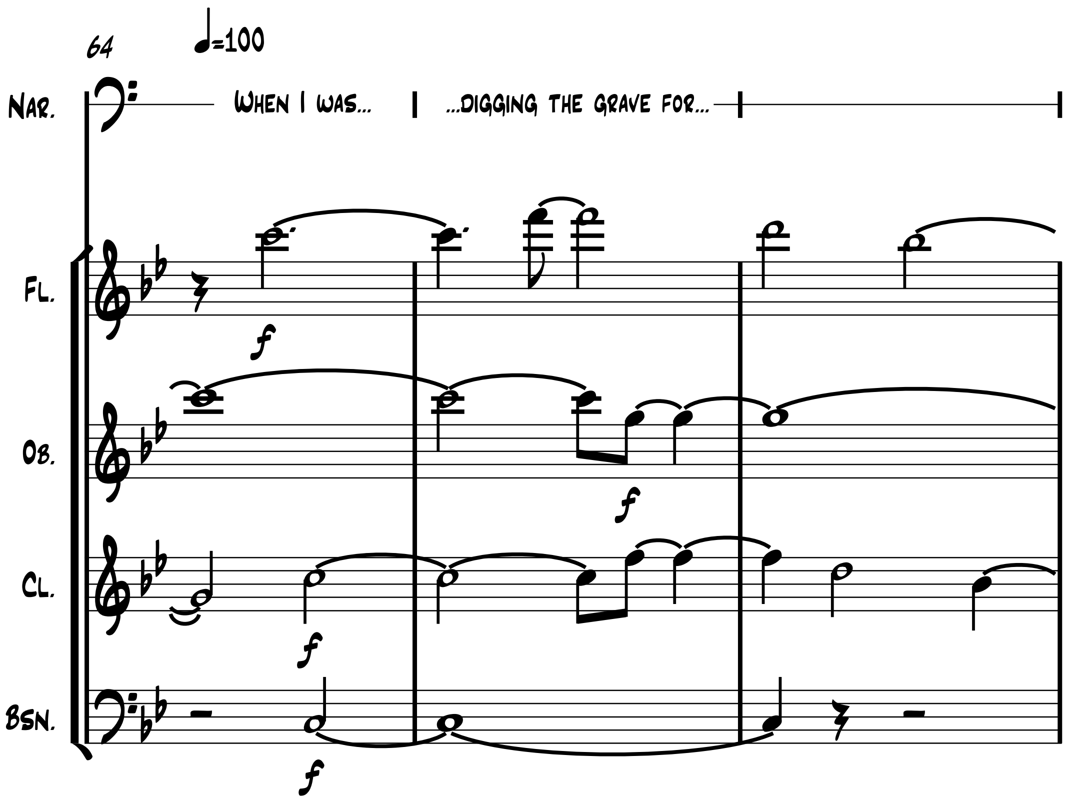
46
Nar.
Fl.
Ob.
Cl.
Bsn.
49
Nar.
Fl.
Ob.
Cl.
Bsn.
p
p
p


51
Nar.
Fl.
Ob.
Cl.
Bsn.


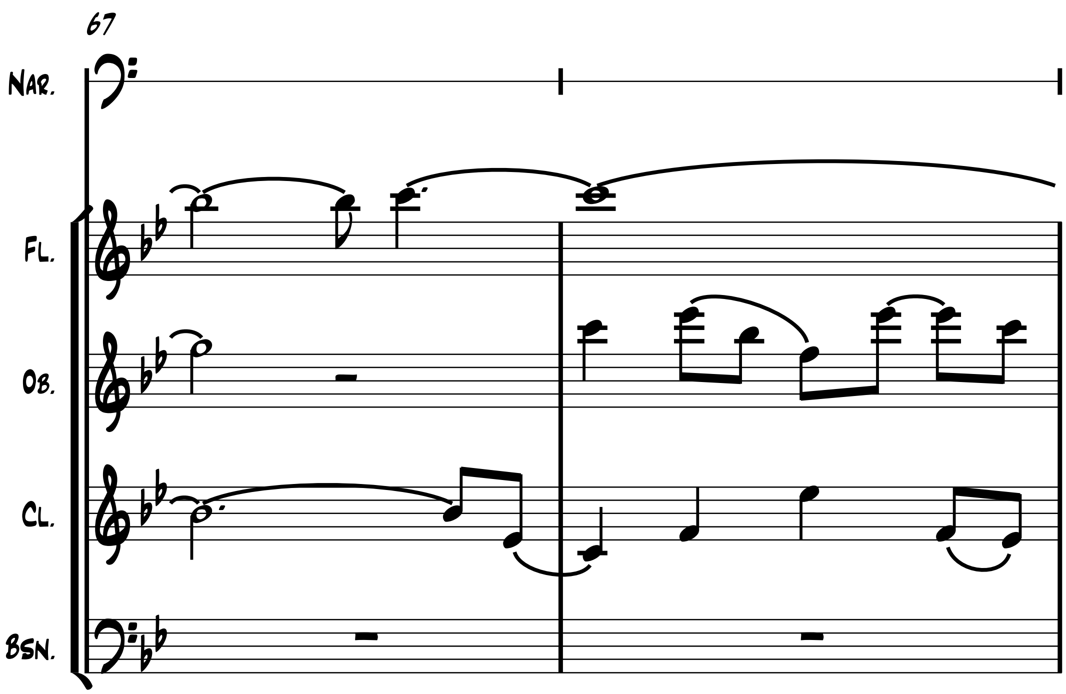
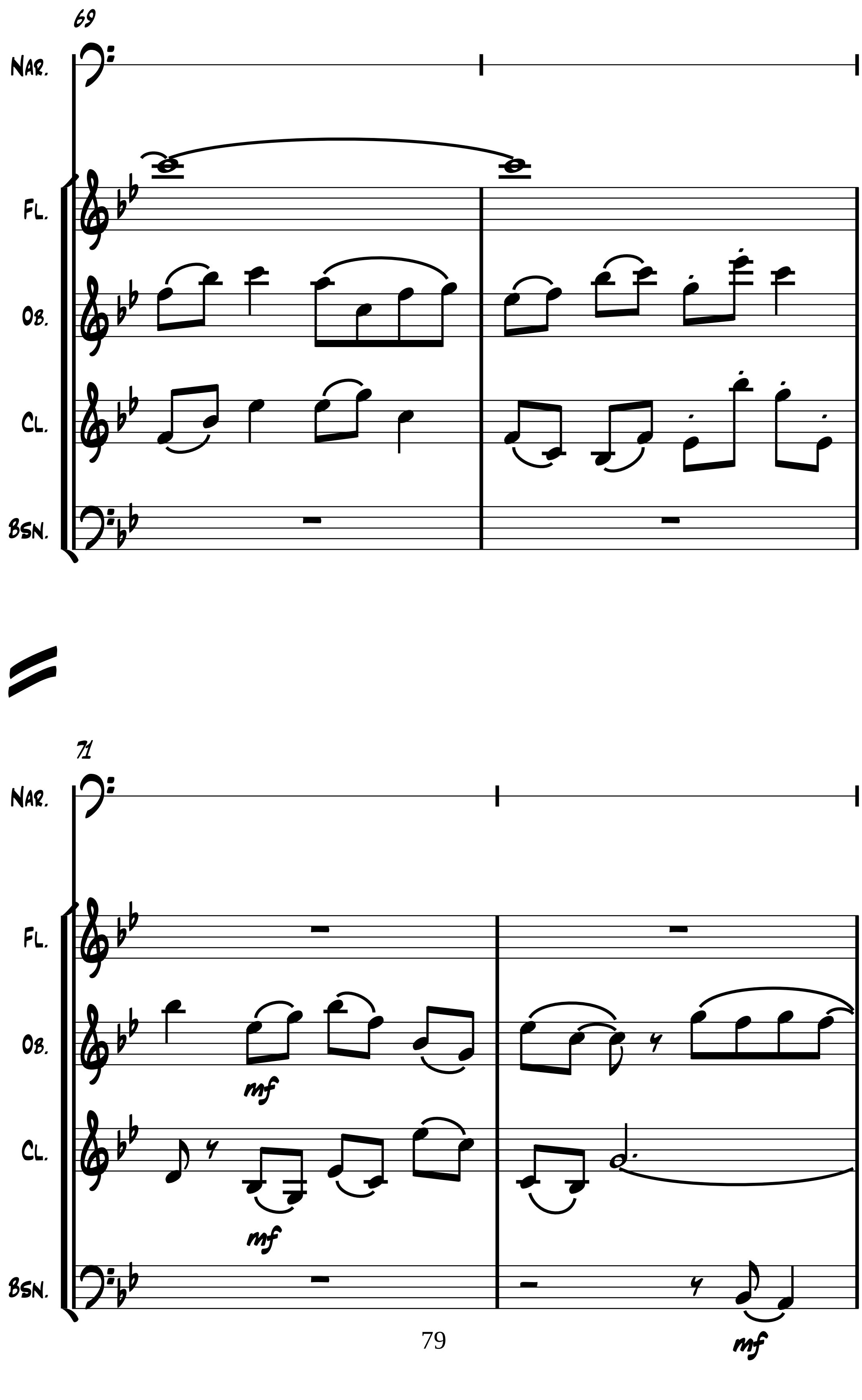
53
Nar.
Fl.
Ob.
Cl.
Bsn.


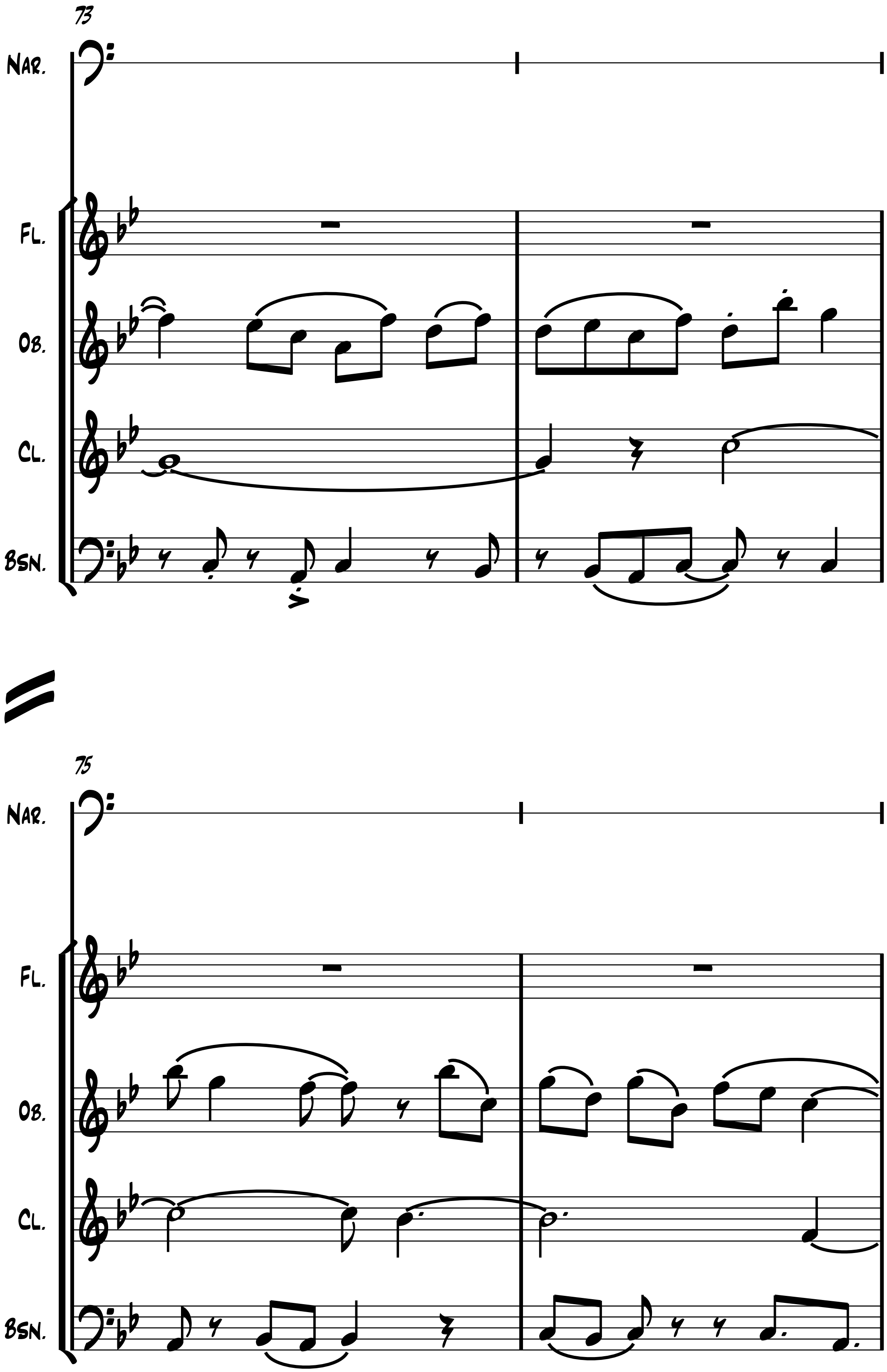
55
Nar.
Fl.
Ob.
Cl.
mp
Bsn.
57
Nar.
Fl.
p
Ob.
Cl.
Bsn.


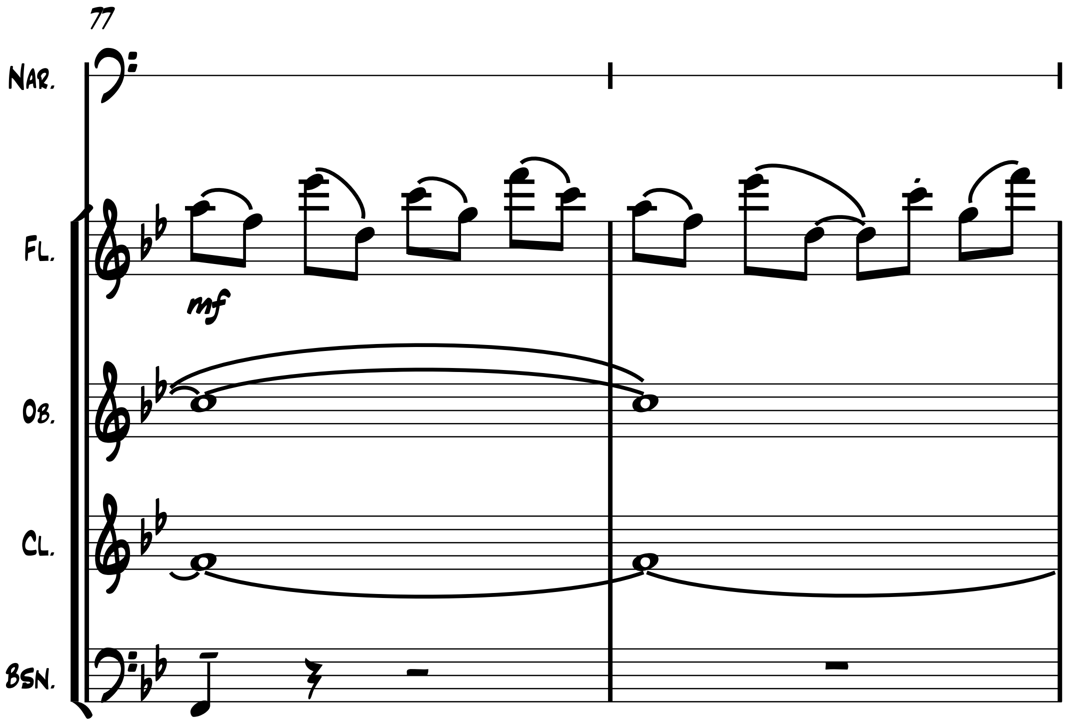
59
Nar.
Fl.
Ob.
Cl.
3
Bsn.


61
RALLENTANDO
Nar.
Fl.
Ob.
mp
Cl.
mf
Bsn.


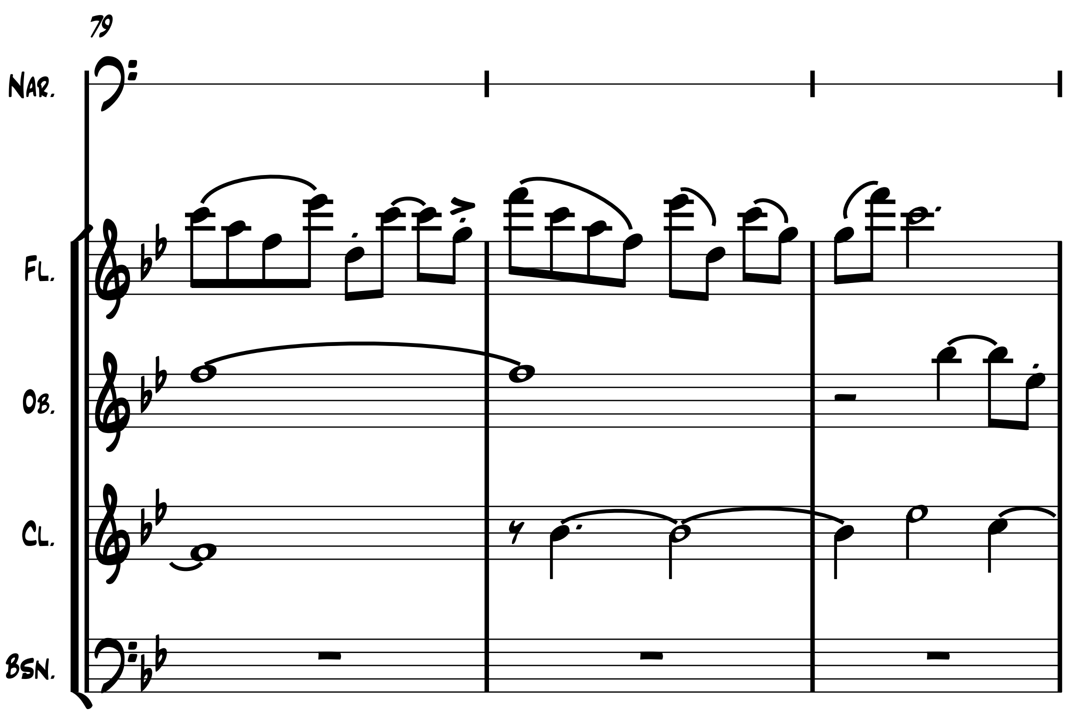
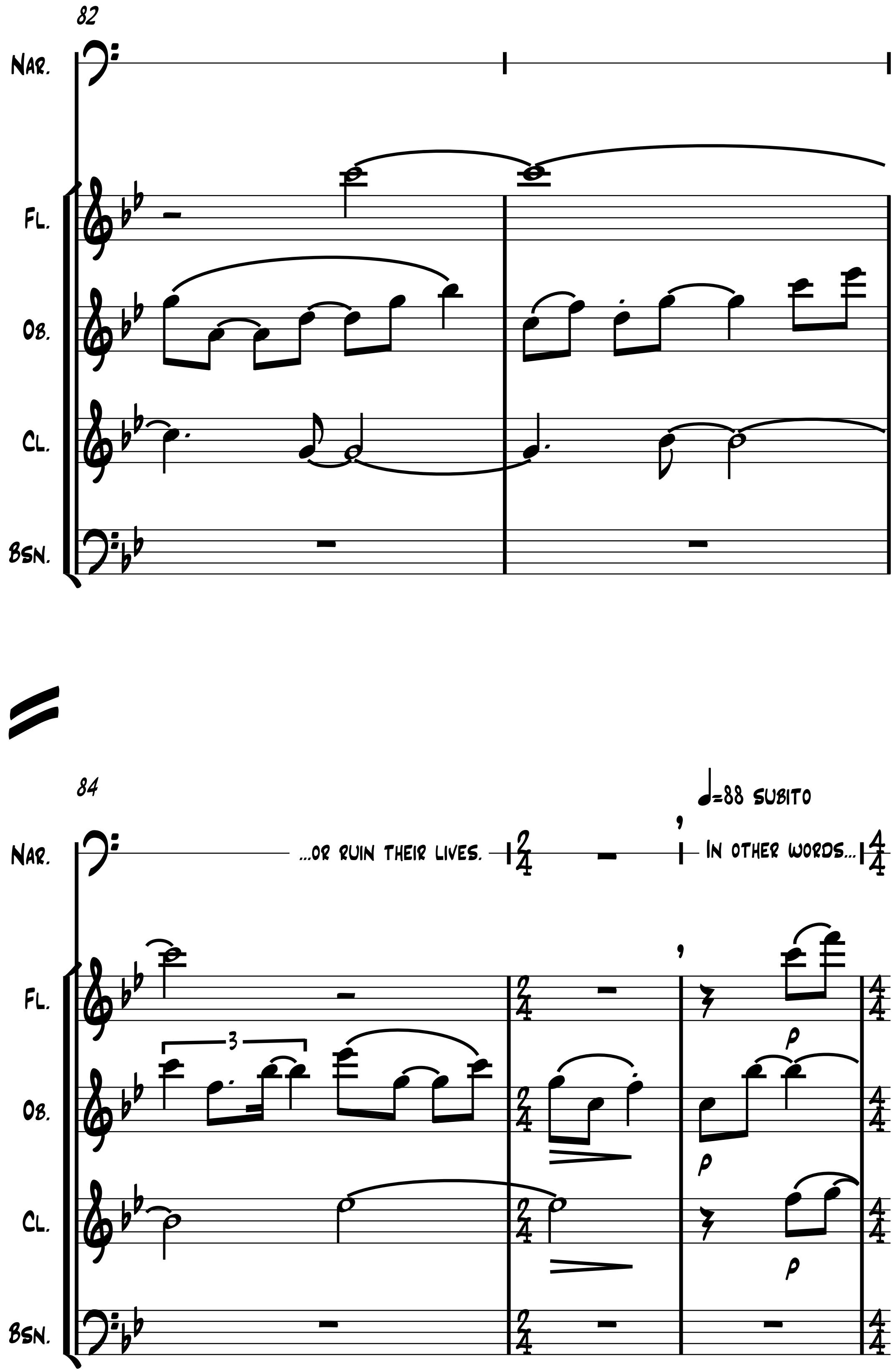
64
♩=100
Nar.
When I was...
...digging the grave for...
Fl.
f
Ob.
f
Cl.
f
Bsn.
f


67
Nar.
Fl.
Ob.
Cl.
Bsn.


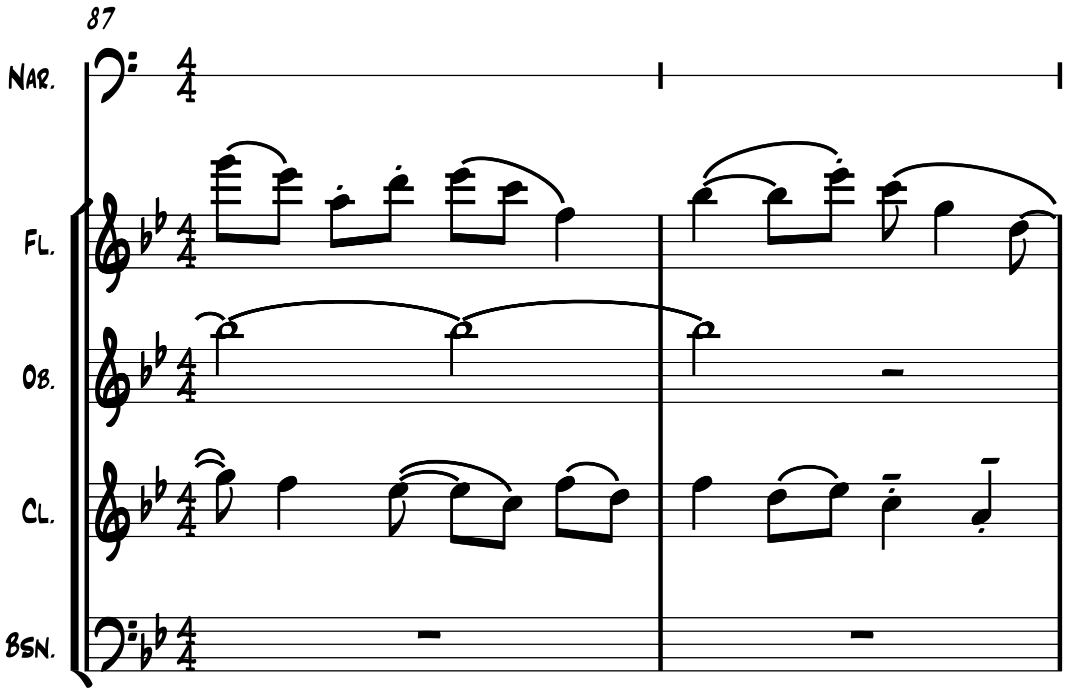
69
Nar.
Fl.
Ob.
Cl.
Bsn.
71
Nar.
Fl.
Ob.
mf
Cl.
mf
Bsn.
mf


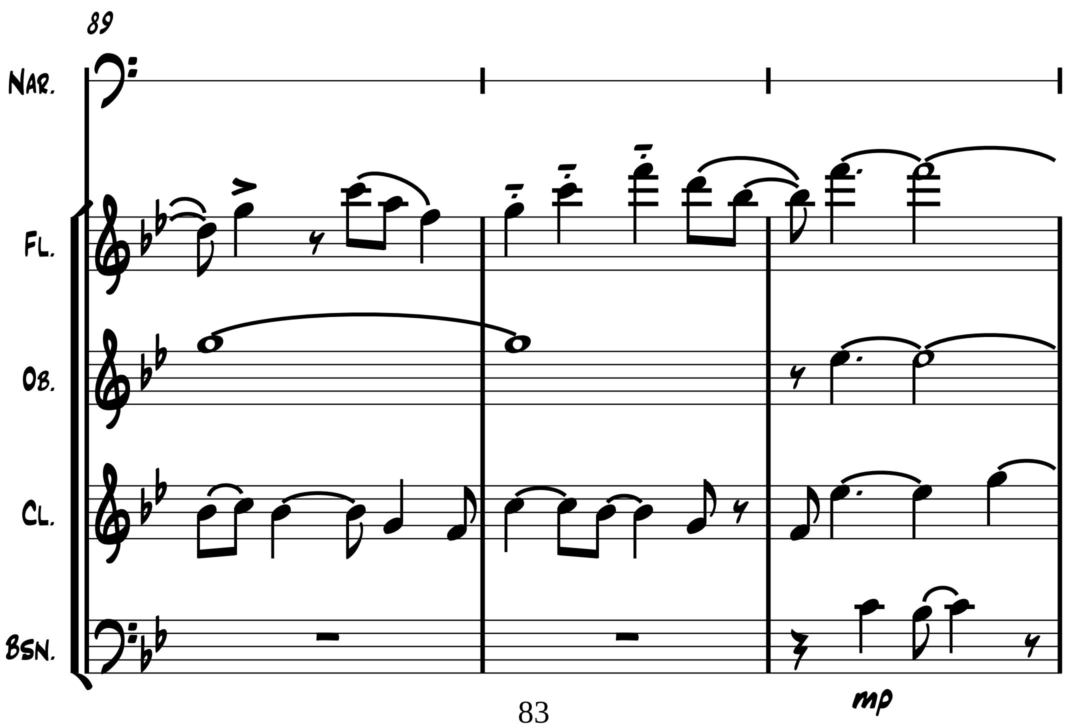
73
Nar.
Fl.
Ob.
Cl.
Bsn.
75
Nar.
Fl.
Ob.
Cl.
Bsn.


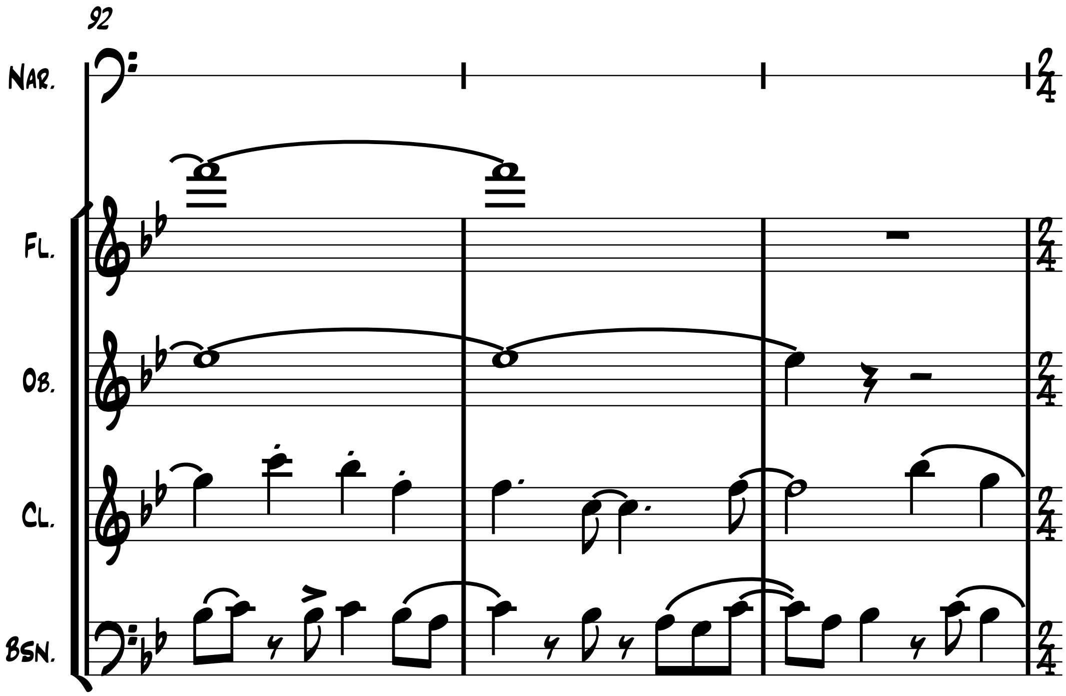
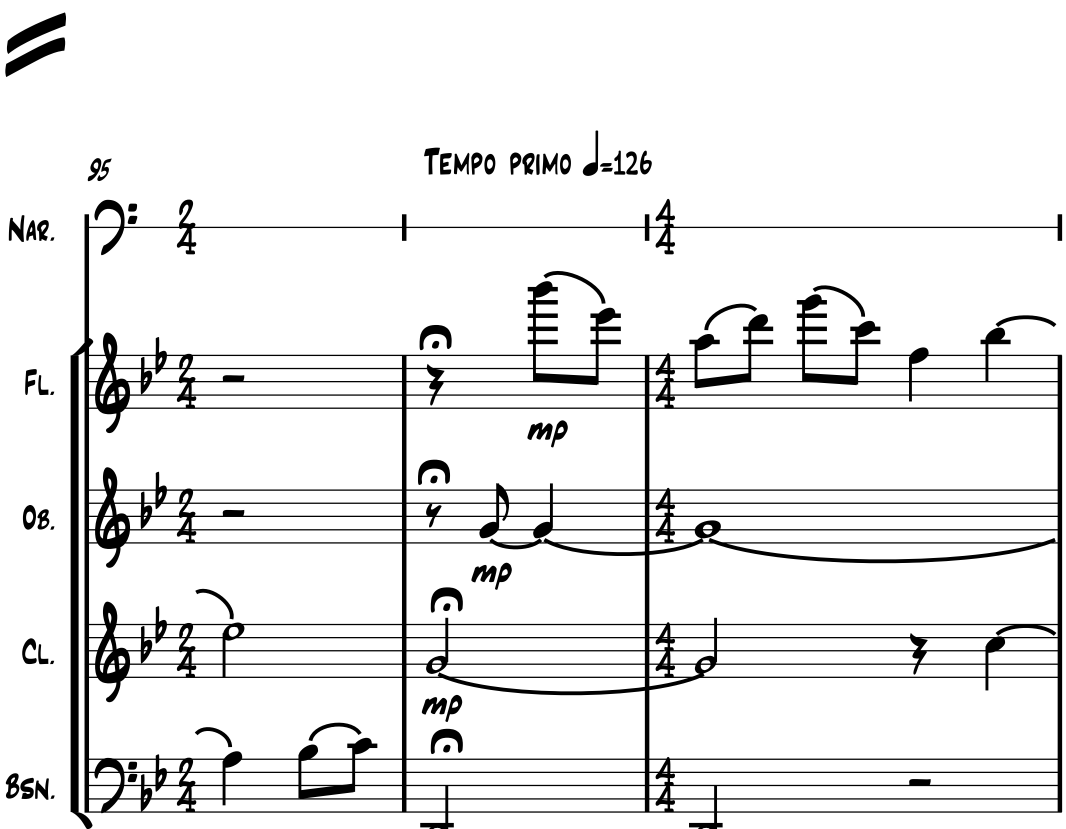
77
Nar.
Fl.
mf
Ob.
Cl.
Bsn.


79
Nar.
Fl.
Ob.
Cl.
Bsn.


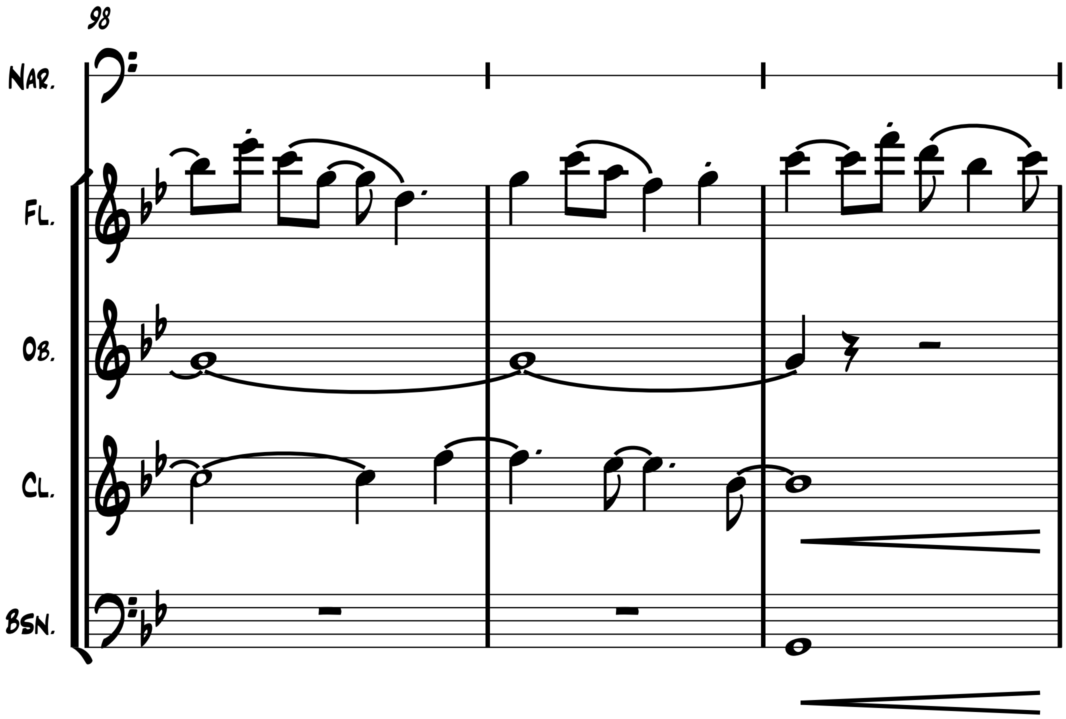
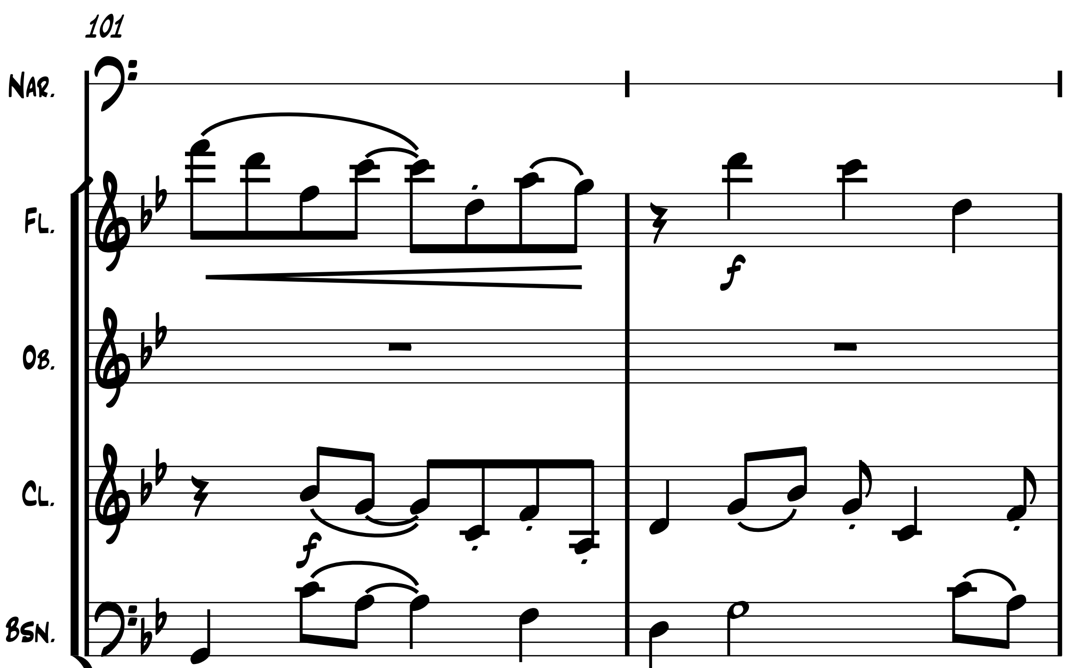
82
Nar.
Fl.
Ob.
Cl.
Bsn.
84
♩=88 subito
...or ruin their lives.
In other words...
3
p
p
p

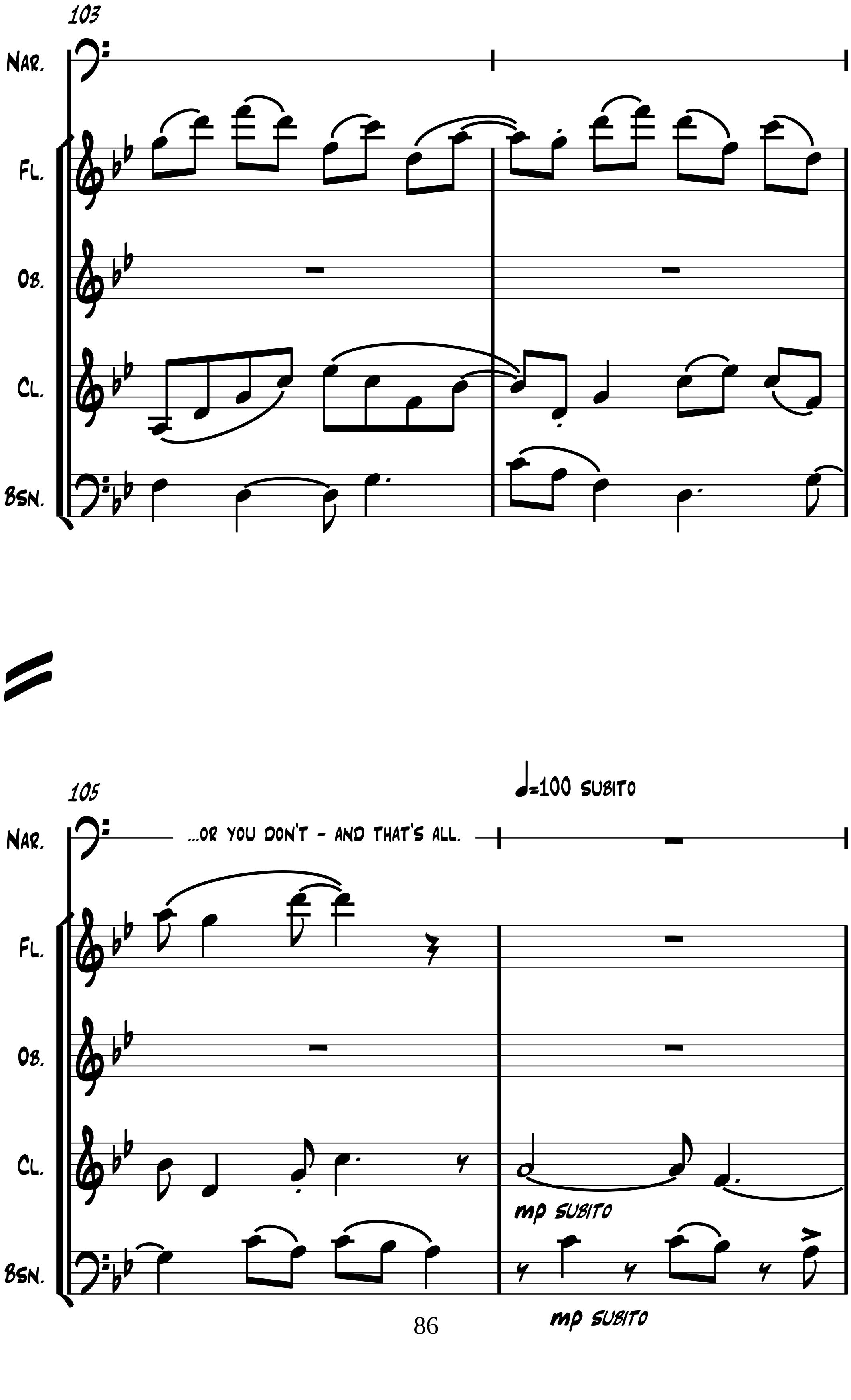

87
Nar.
Fl.
Ob.
Cl.
Bsn.

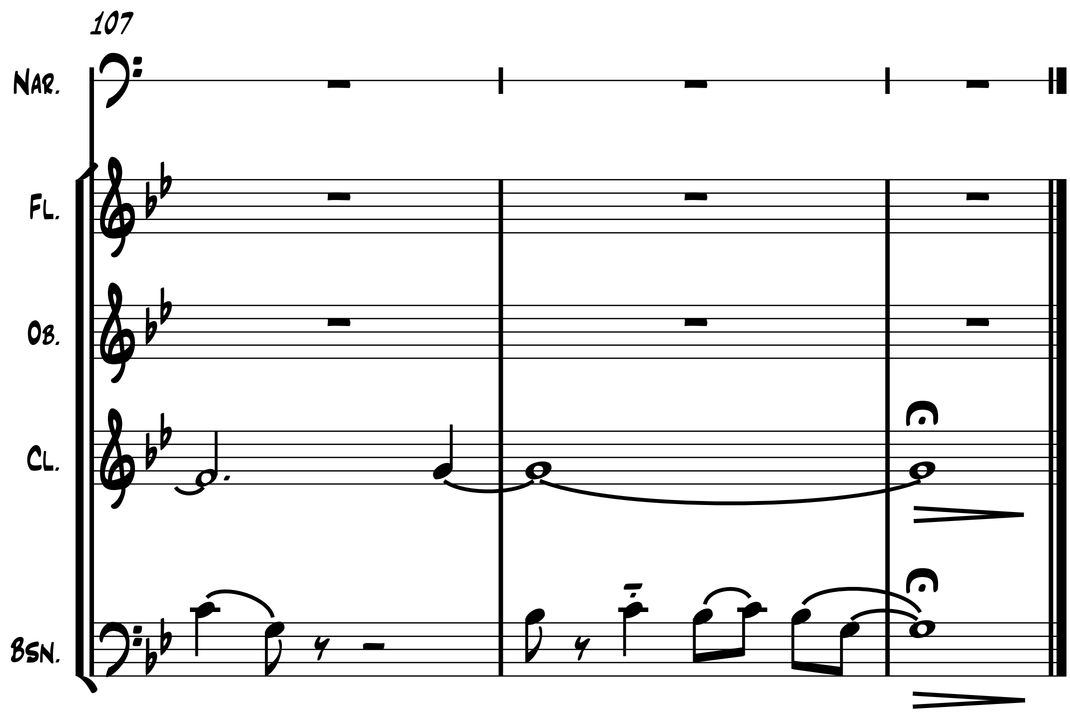

89
Nar.
Fl.
Ob.
Cl.
Bsn.
mp


92
Nar.
Fl.
Ob.
Cl.
Bsn.


95
Tempo primo ♩=126
Nar.
Fl.
Ob.
Cl.
Bsn.
mp
mp
mp


98
Nar.
Fl.
Ob.
Cl.
Bsn.


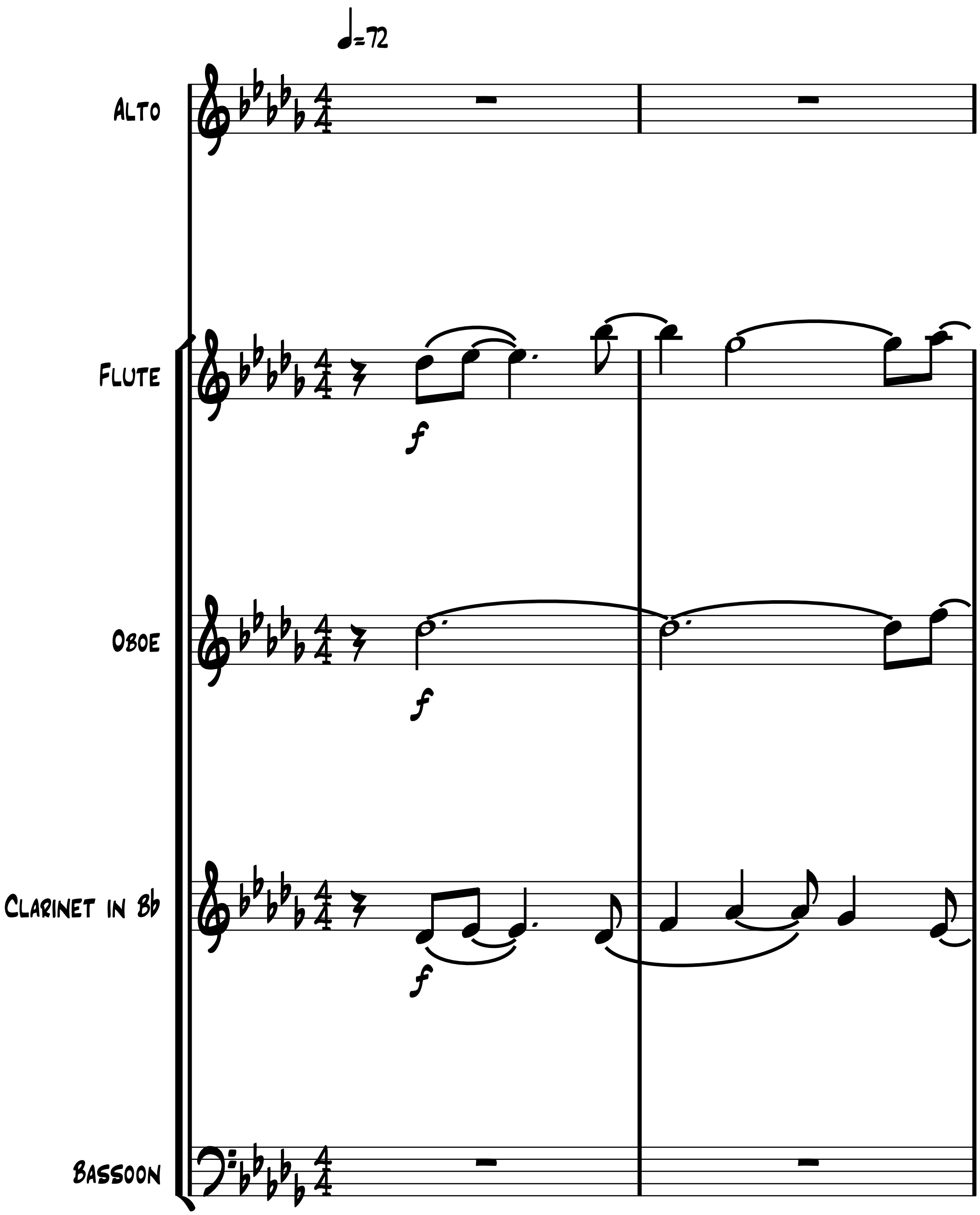
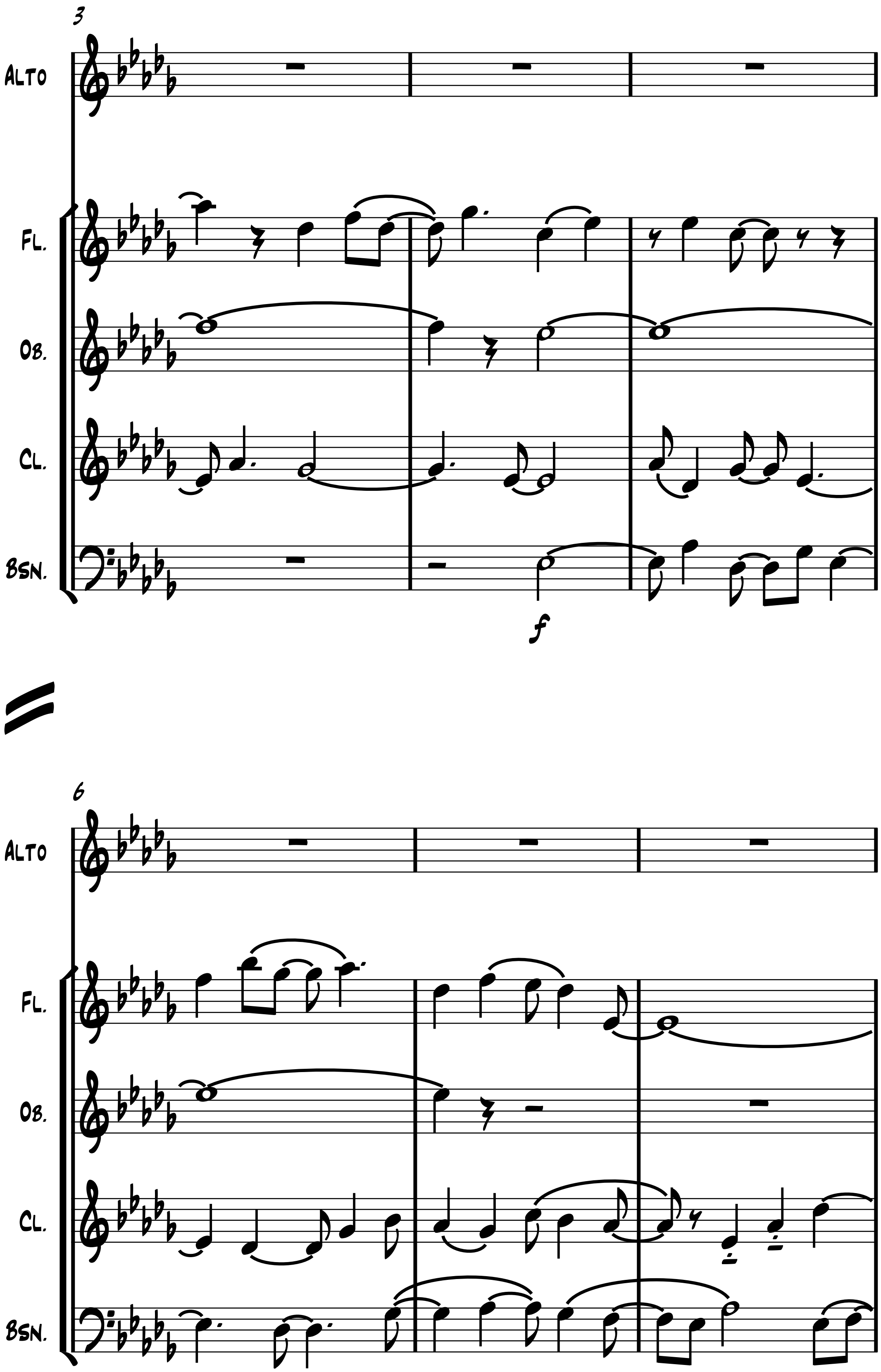
101
Nar.
Fl.
Ob.
Cl.
Bsn.
f
f
f


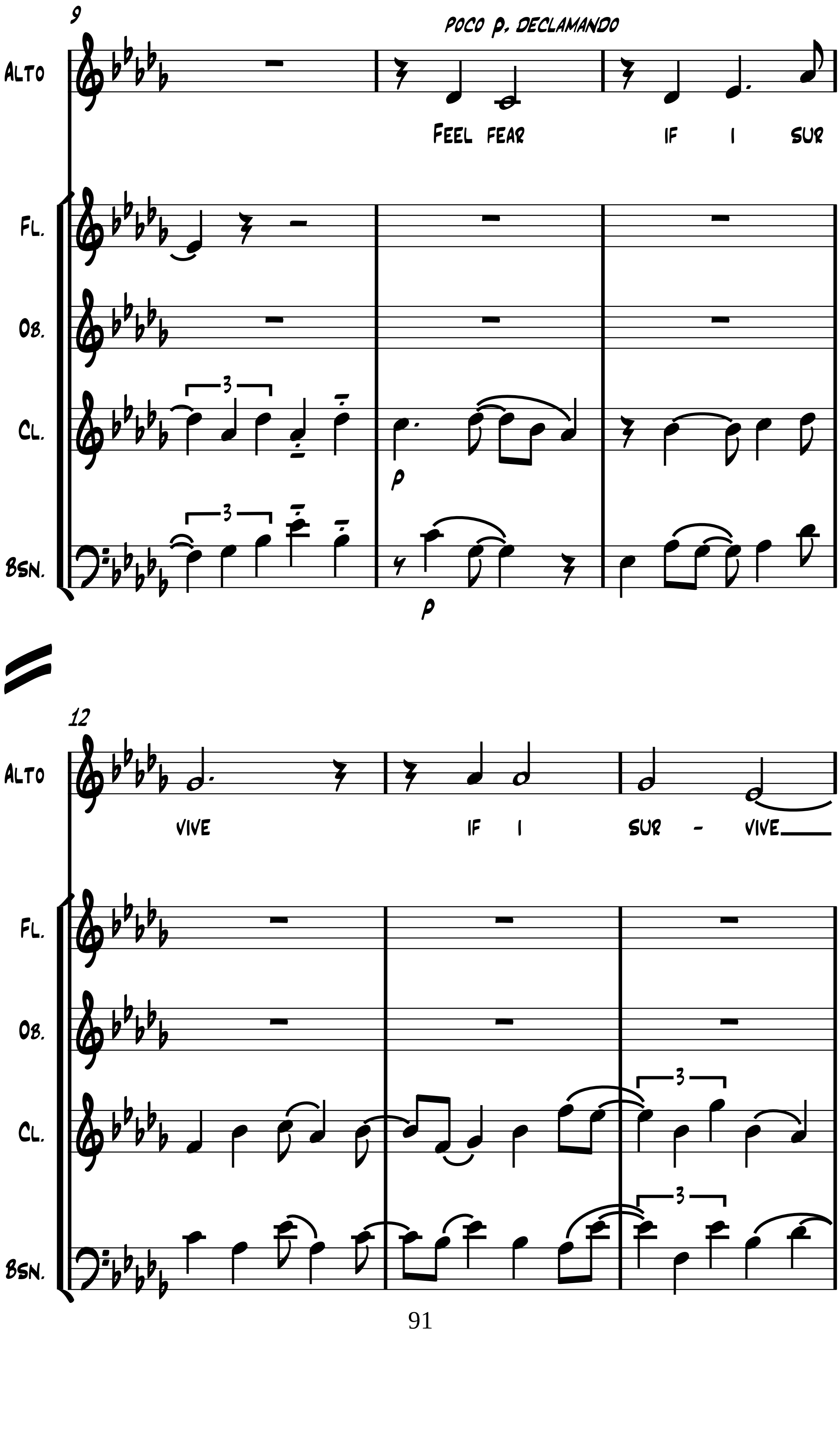
103
Nar.
Fl.
Ob.
Cl.
Bsn.
105
♩=100 subito
...or you don't - and that's all.
mp subito
mp subito


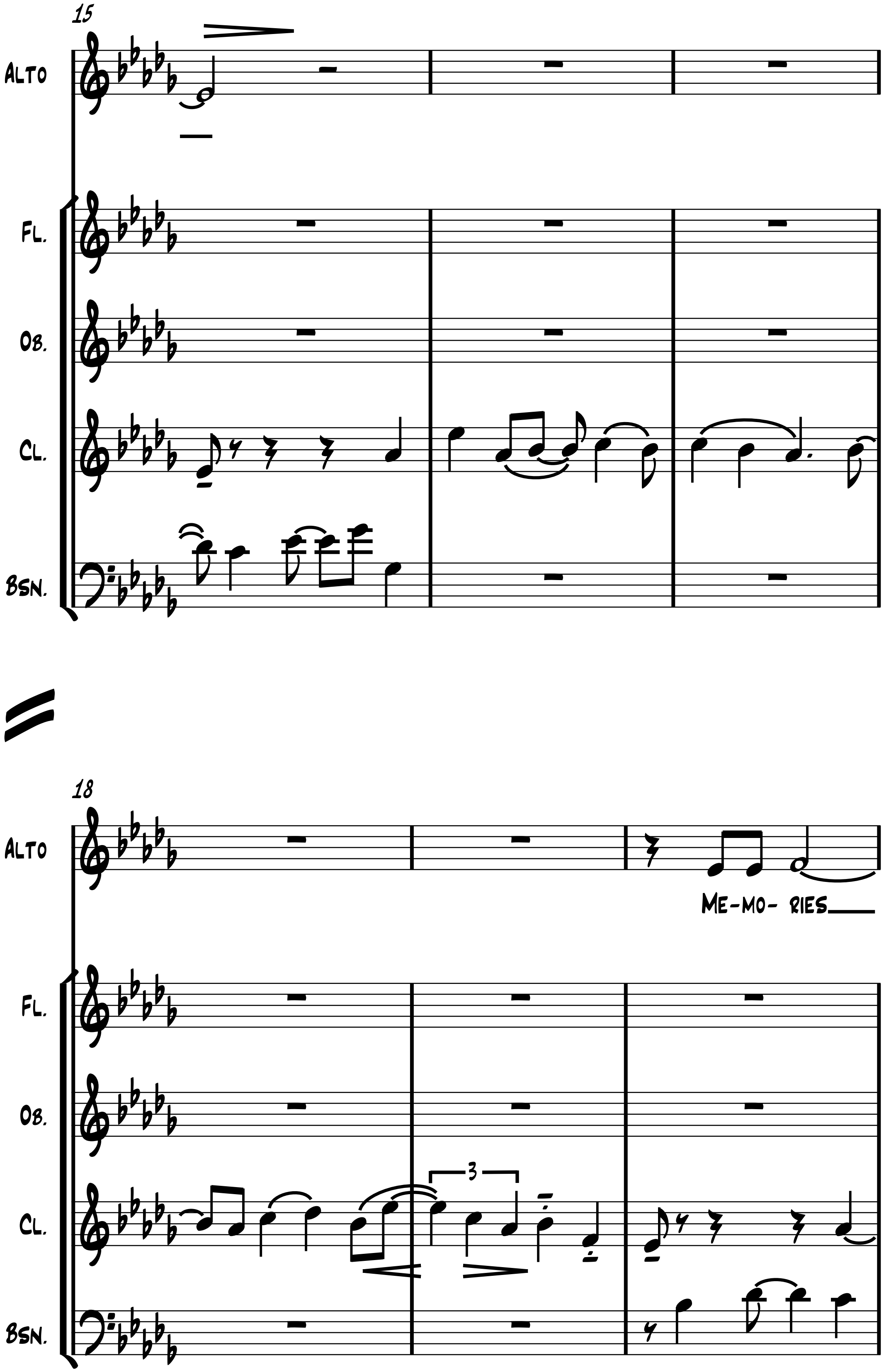
107
Nar.
Fl.
Ob.
Cl.
Bsn.

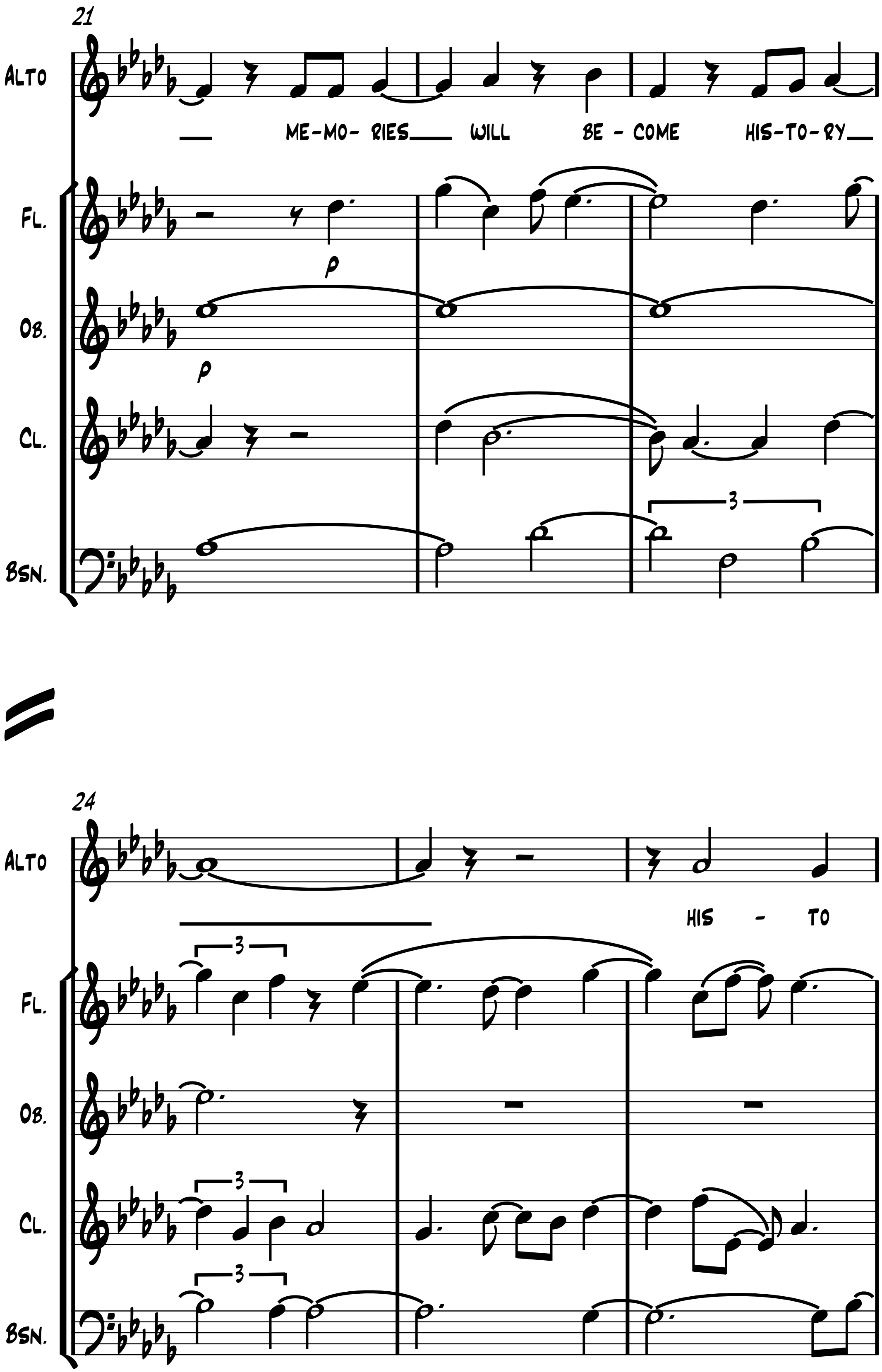
APPENDIX C

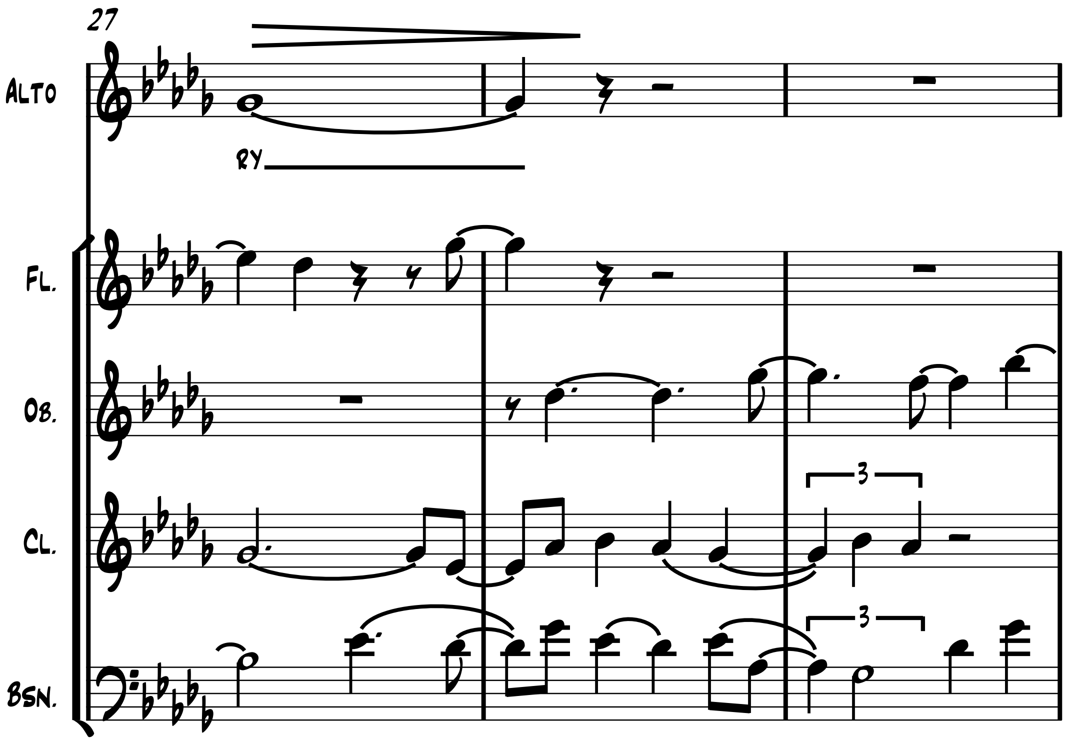
…she turned from Ares her bright eyes, Movement 3

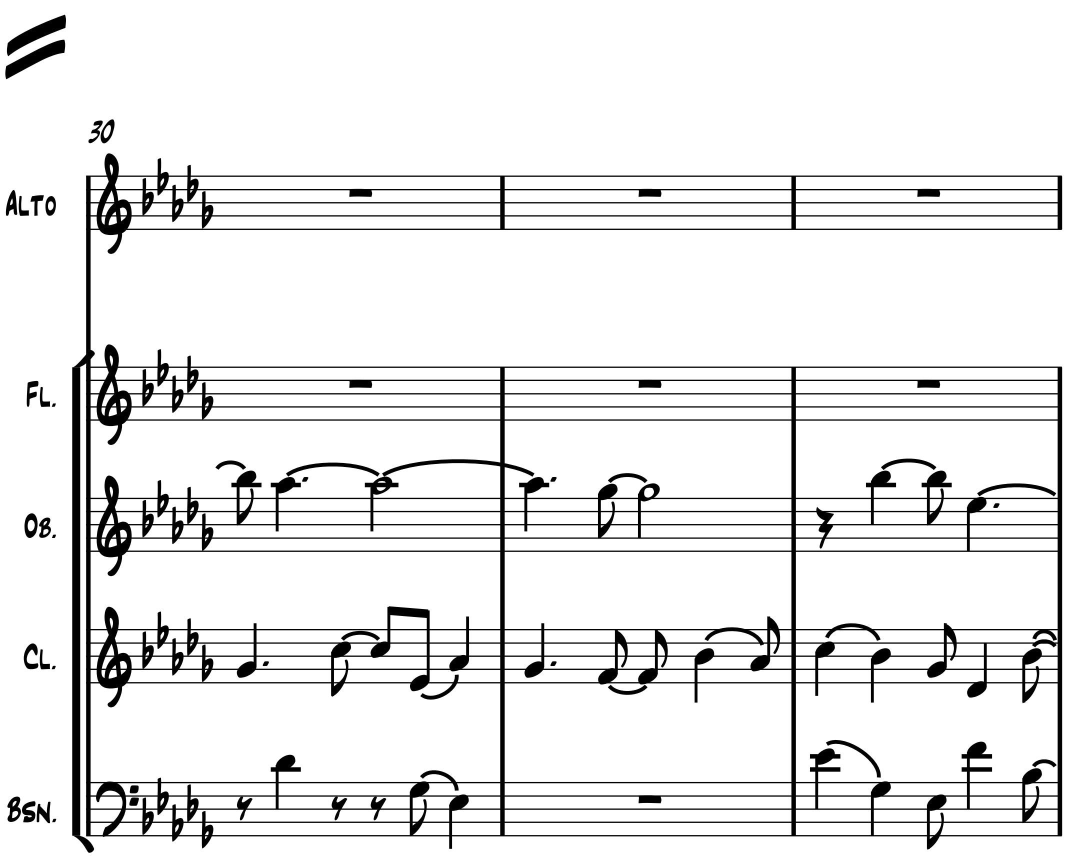
3.


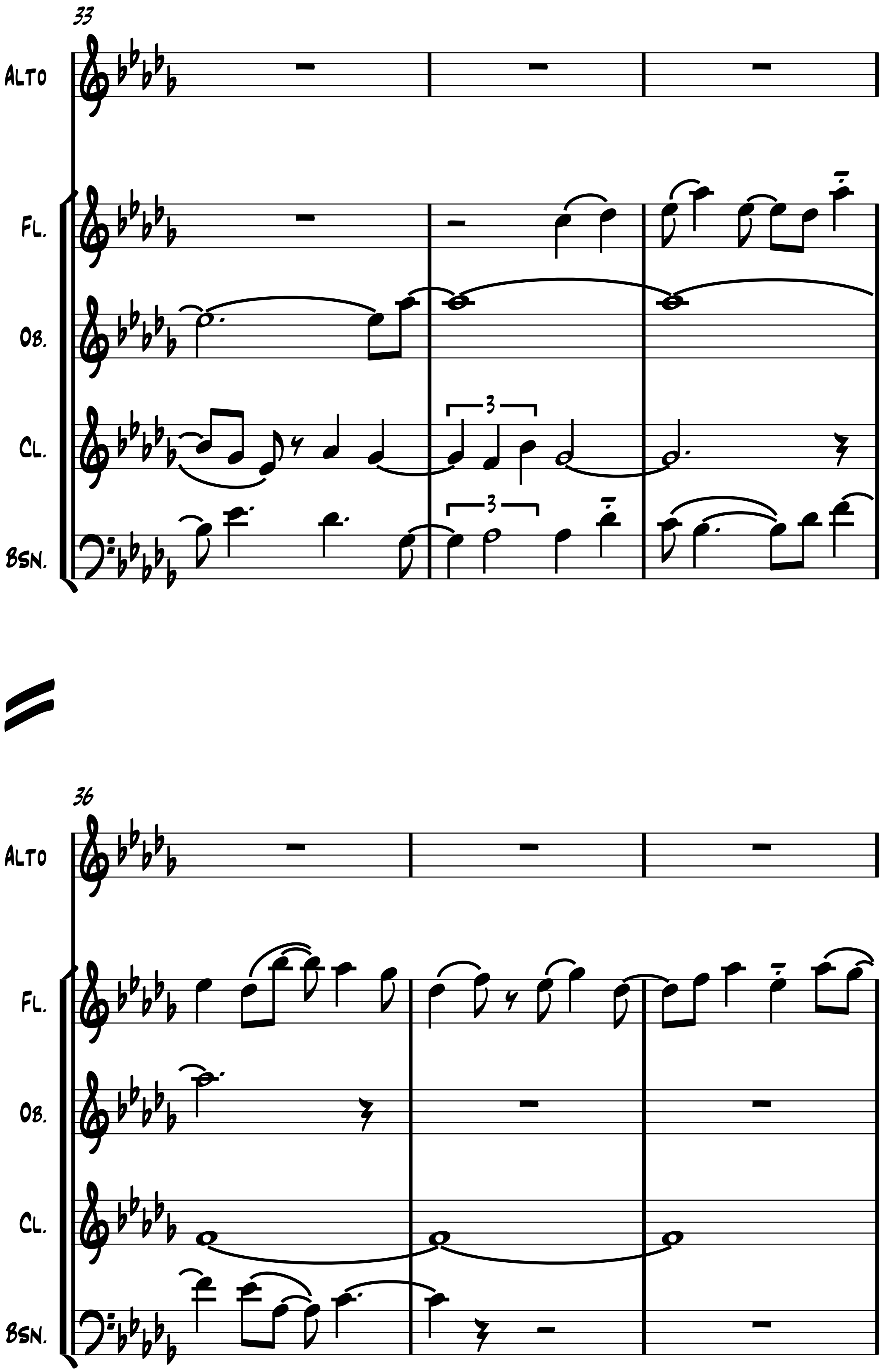
3
Alto
Fl.
Ob.
Cl.
Bsn.
f
6
Alto
Fl.
Ob.
Cl.
Bsn.


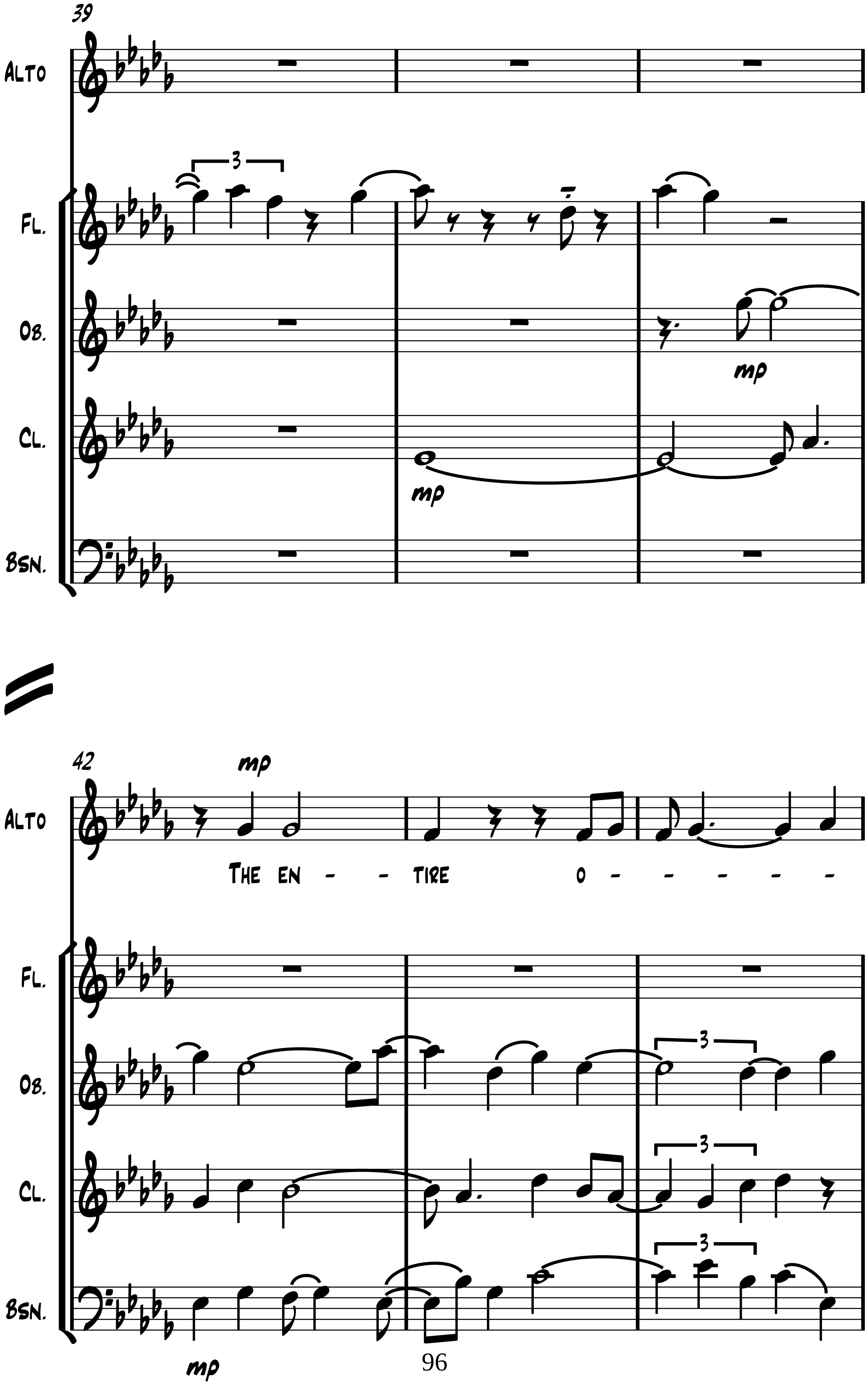
9
poco p. declamando
Alto
Feel fear
if I sur
Fl.
Ob.
Cl.
Bsn.
12
vive
if I
sur - vive


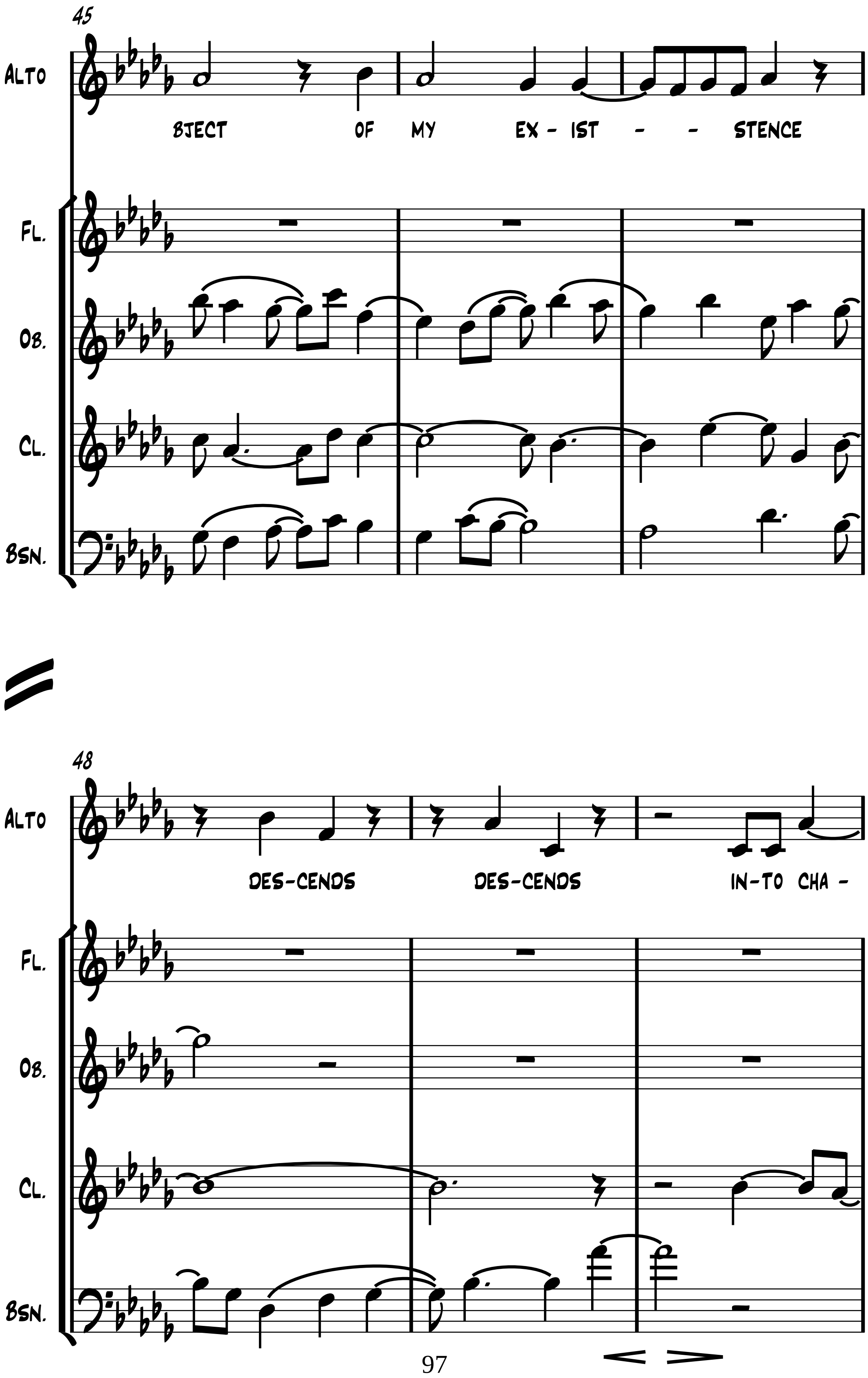
15
Alto
Fl.
Ob.
Cl.
Bsn.
18
Alto
Me-mo- ries
Fl.
Ob.
3
Cl.
Bsn.


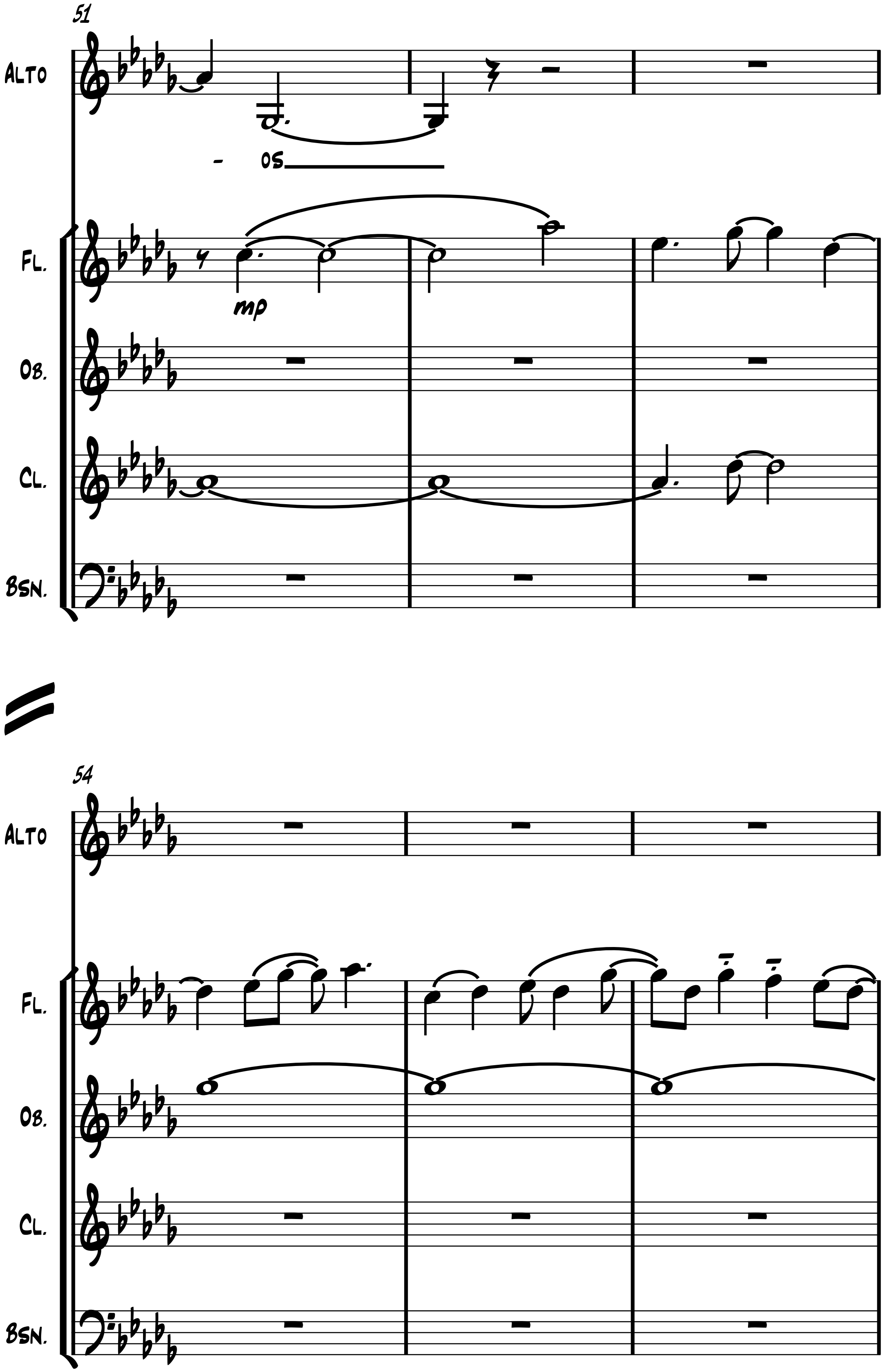
21
Alto
Fl.
Ob.
Cl.
Bsn.
ME-MO- RIES WILL BE - COME HIS-TO-RY
p
p
3
24
3
3
3
HIS - TO


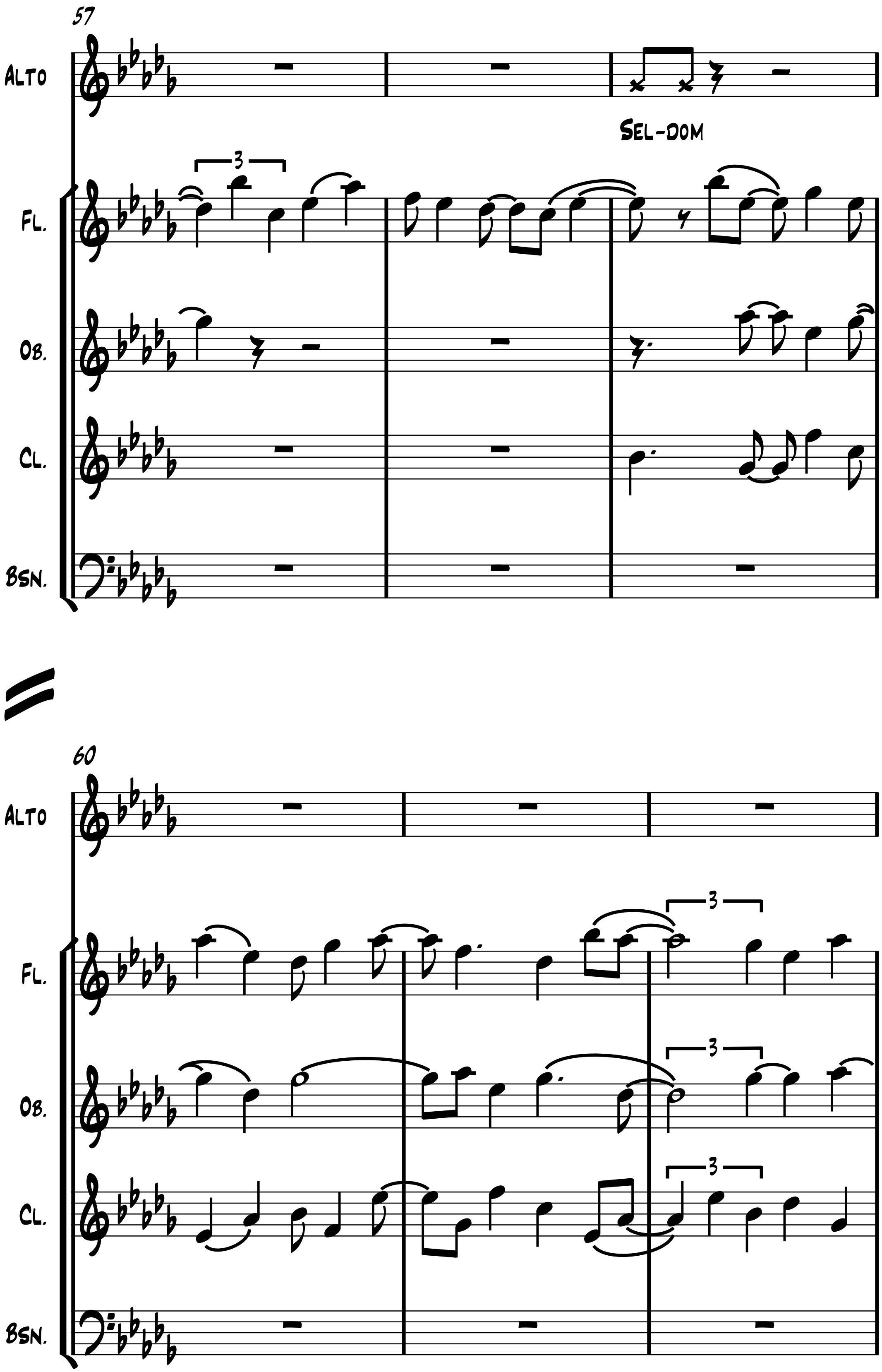
27
Alto
RY
Fl.
Ob.
Cl.
3
Bsn.
3


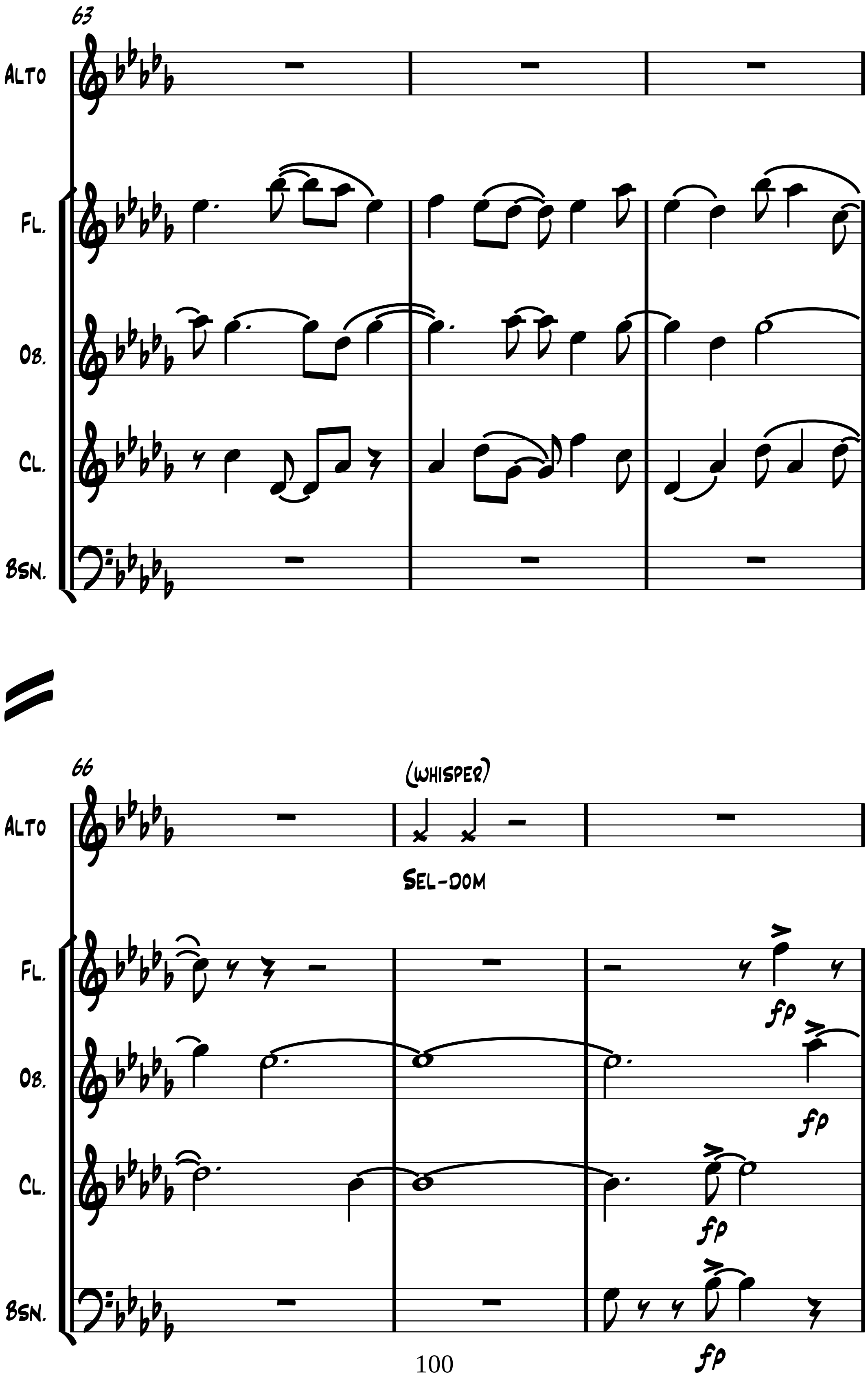
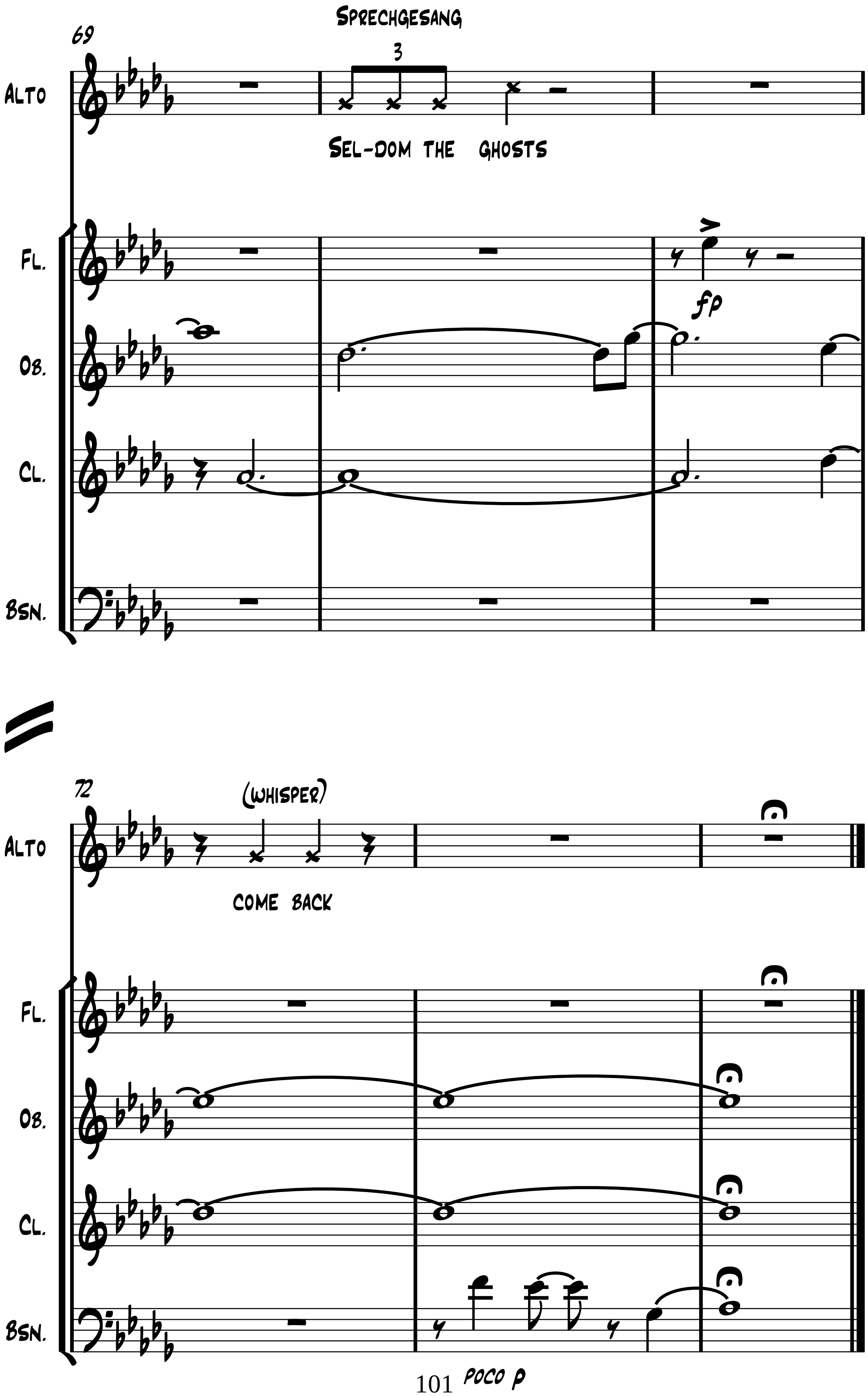
30
Alto
Fl.
Ob.
Cl.
Bsn.


33
Alto
Fl.
Ob.
Cl.
Bsn.
3
3
36
Alto
Fl.
Ob.
Cl.
Bsn.


39
Alto
Fl.
Ob.
Cl.
Bsn.
3
mp
mp
42
mp
The en - - tire o - - - - -
3
3
3
mp


45
Alto
Fl.
Ob.
Cl.
Bsn.
BJECT OF MY EX - IST - - STENCE
48
Alto
Fl.
Ob.
Cl.
Bsn.
DES-CENDS DES-CENDS IN-TO CHA -


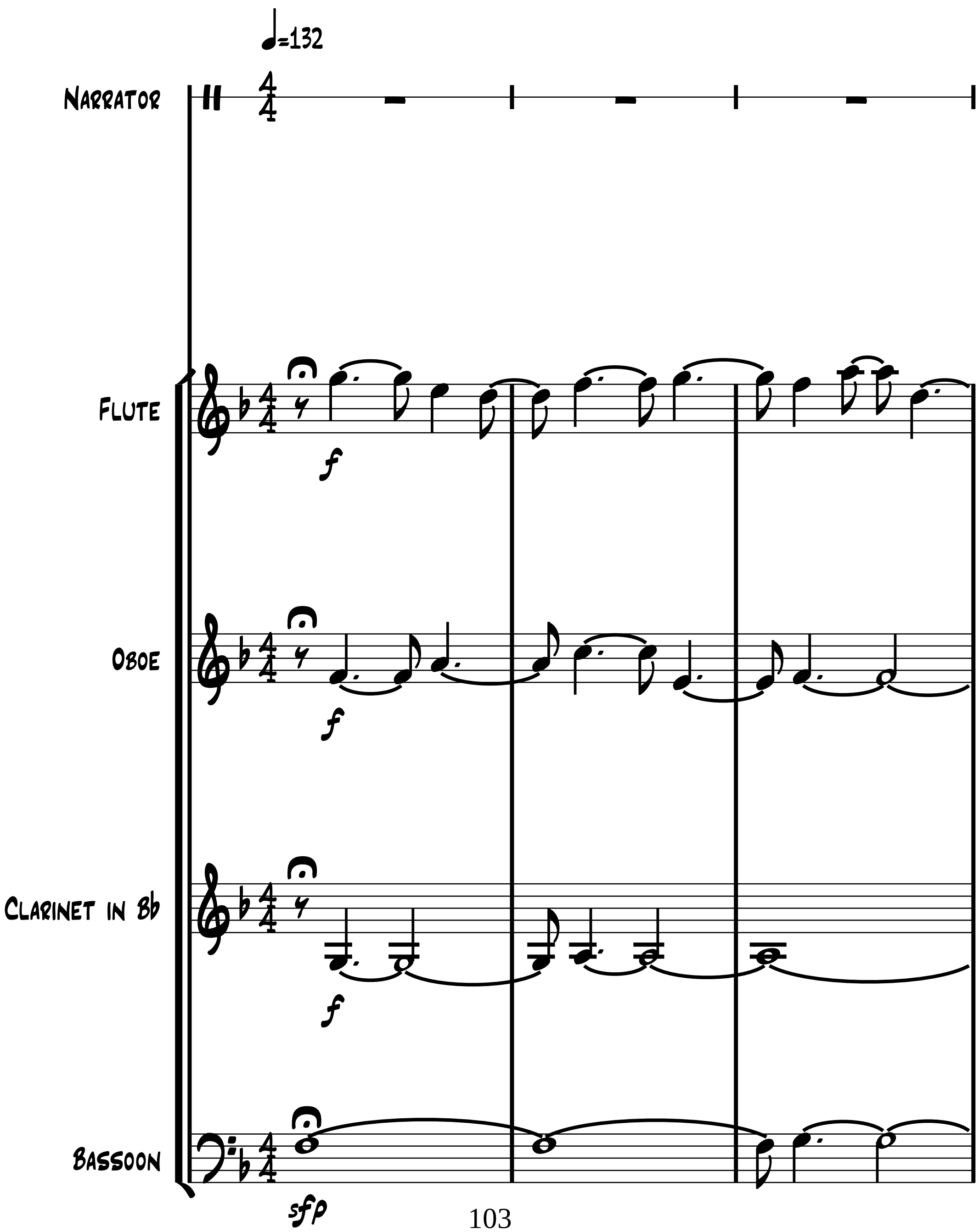
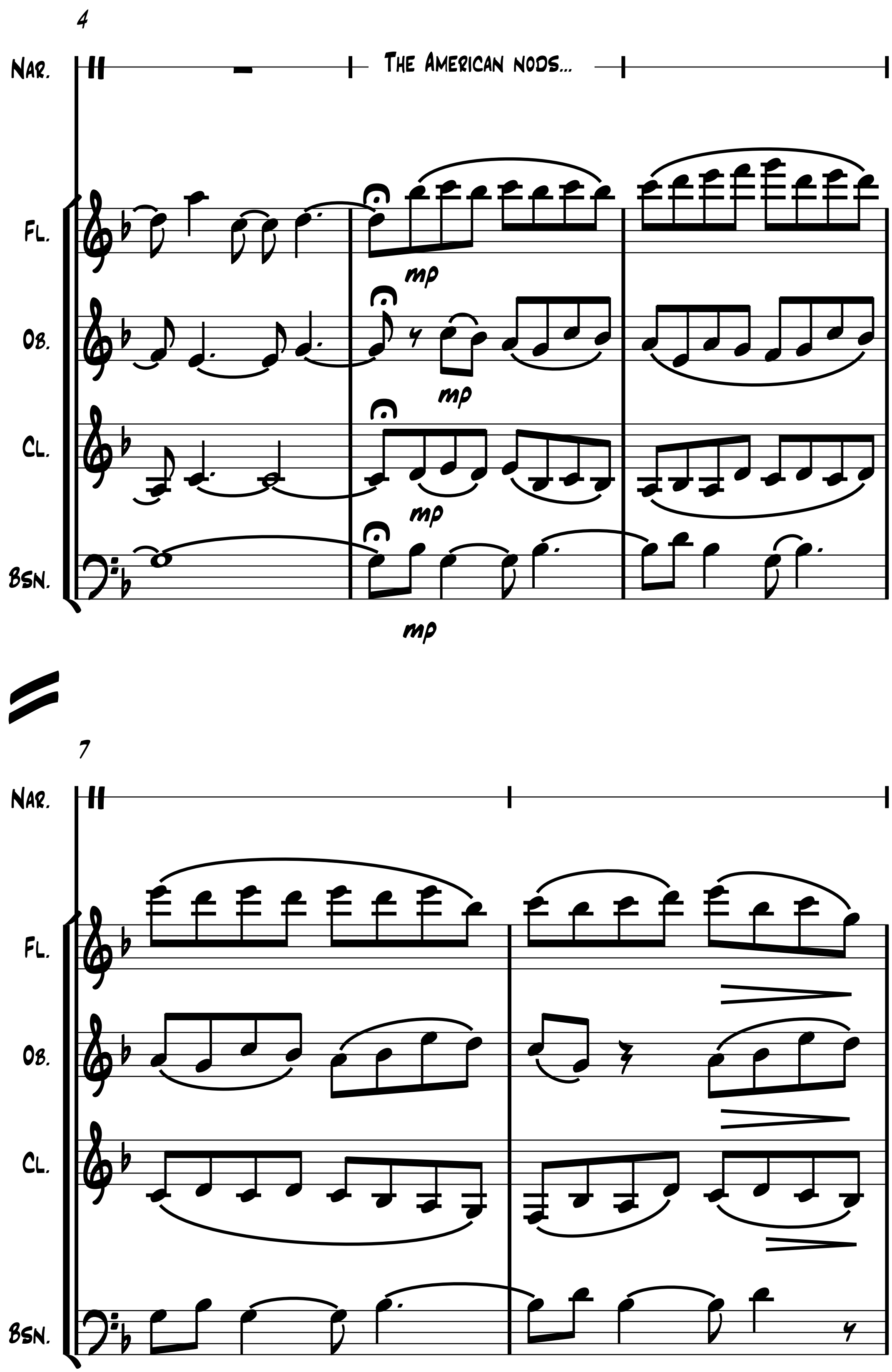
51
Alto
- os
Fl.
mp
Ob.
Cl.
Bsn.
54
Alto
Fl.
Ob.
Cl.
Bsn.


57
Alto
Sel-dom
Fl.
Ob.
Cl.
Bsn.
60
Alto
Fl.
Ob.
Cl.
Bsn.

63
Alto
Fl.
Ob.
Cl.
Bsn.
66
(whisper)
Sel-dom
fp


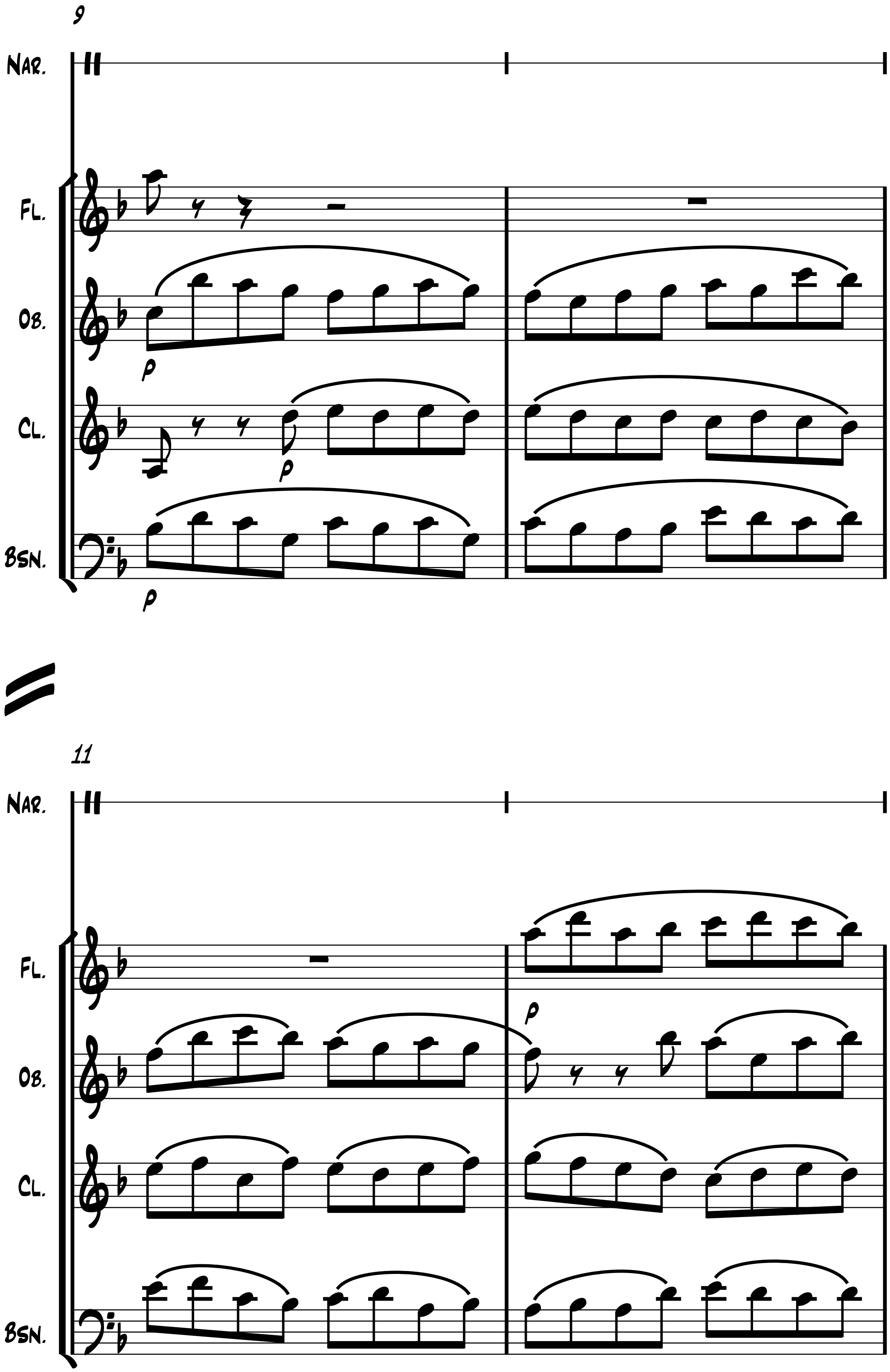
69
Sprechgesang
3
Alto
Sel-dom the ghosts
Fl.
fp
Ob.
Cl.
Bsn.
72
(whisper)
Alto
come back
Fl.
Ob.
Cl.
Bsn.
poco p

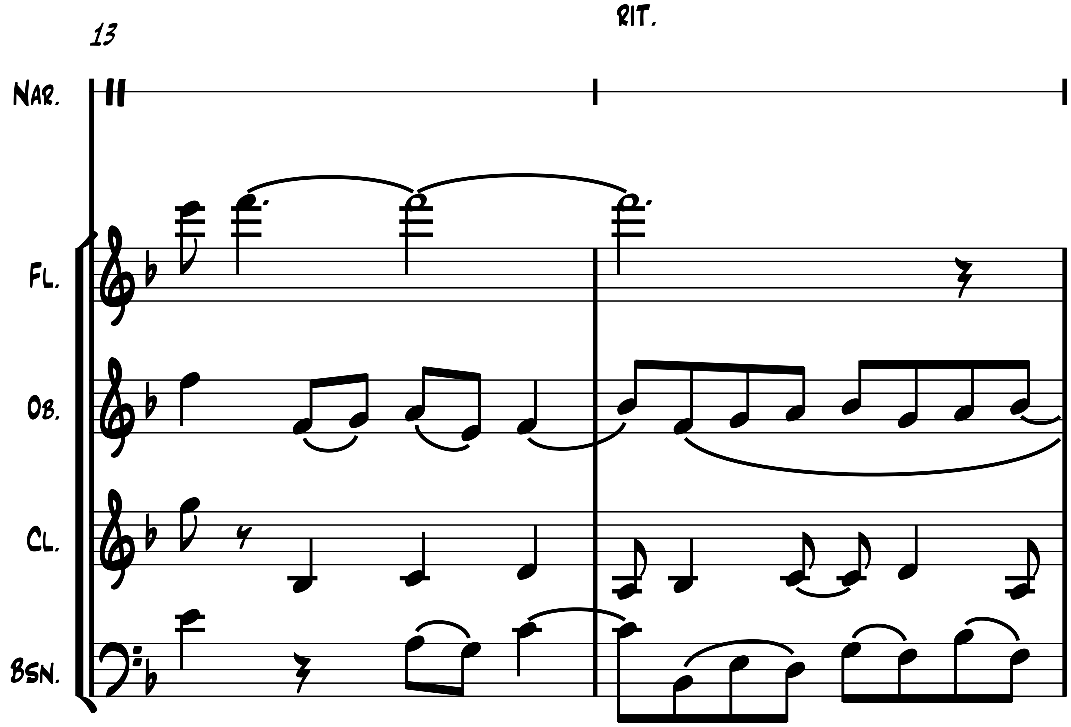
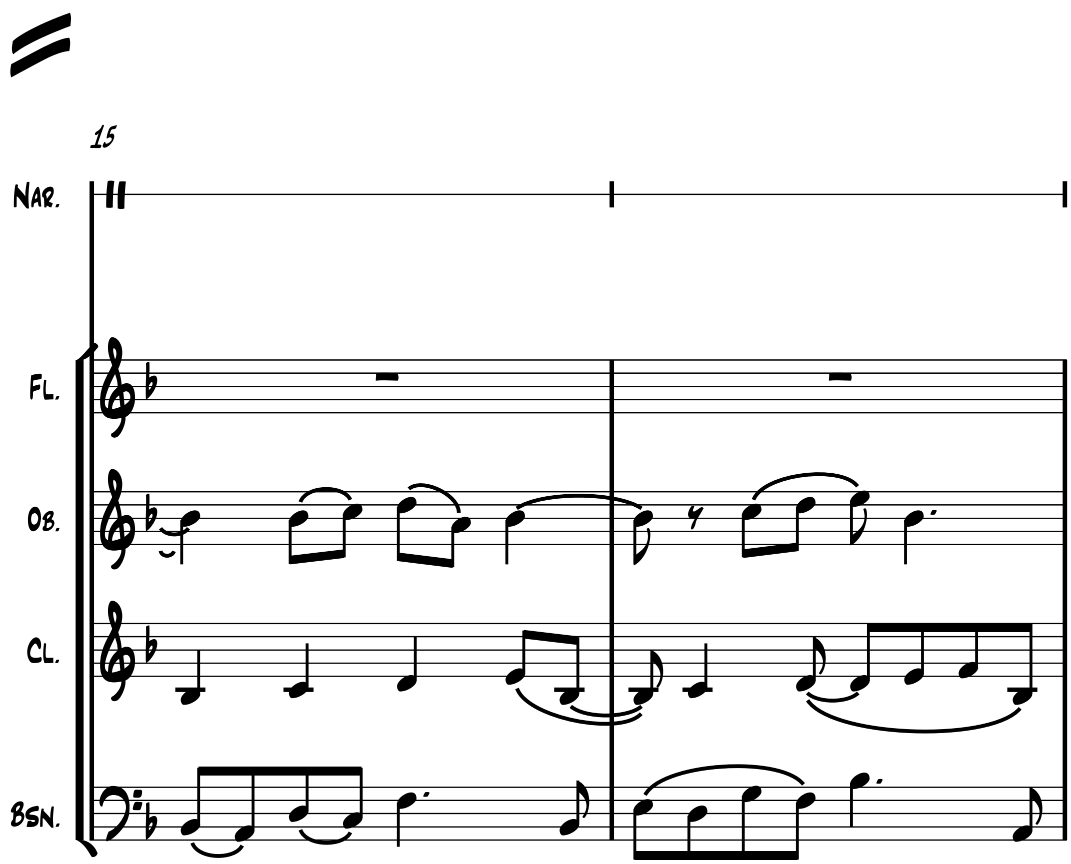
# APPENDIX D

…she turned from Ares her bright eyes, Movement 4

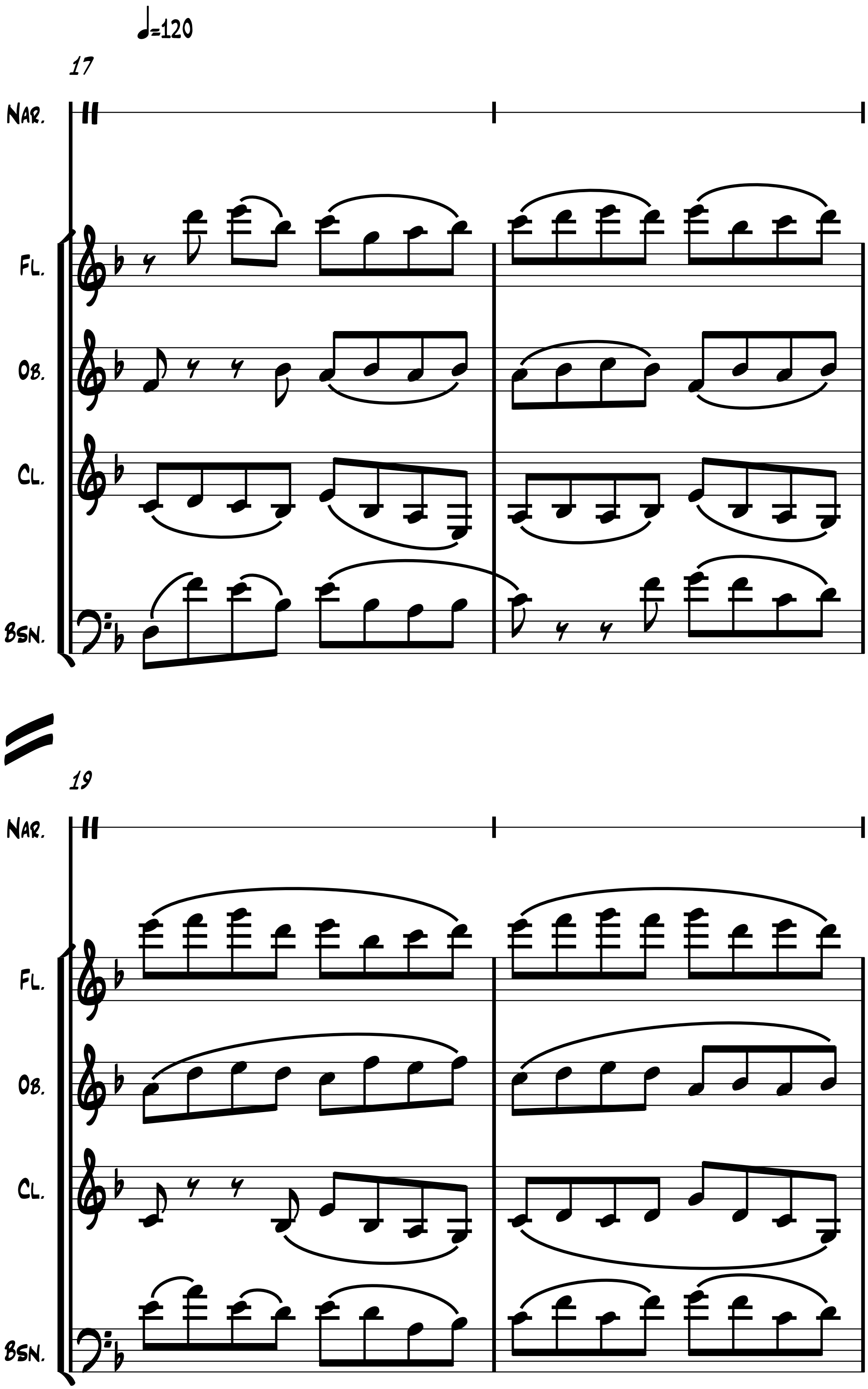
4.


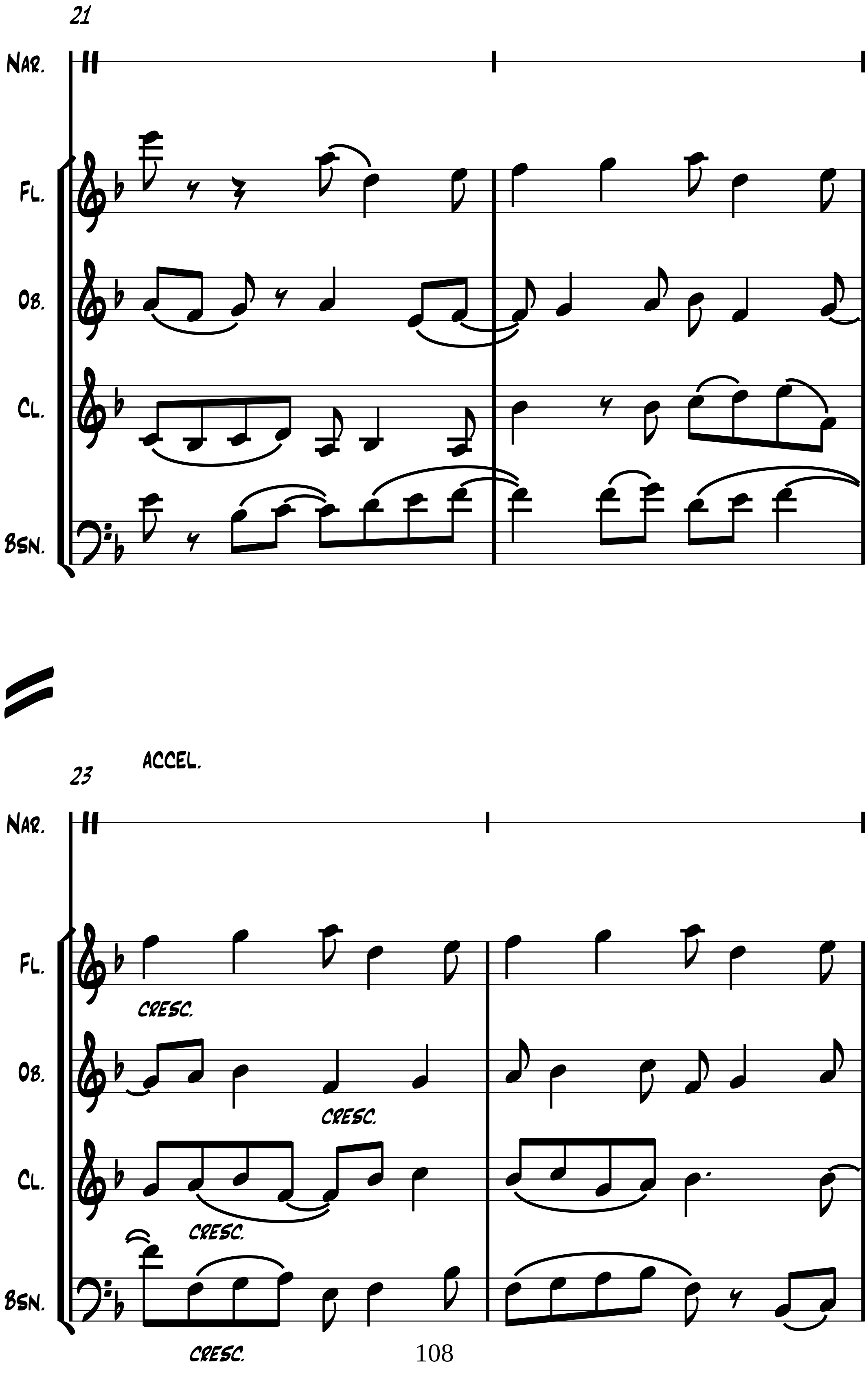
4
Nar.
The American nods...
Fl.
Ob.
Cl.
Bsn.
mp
mp
mp
mp
7
Nar.
Fl.
Ob.
Cl.
Bsn.


9
Nar.
Fl.
Ob.
Cl.
Bsn.
p
p
p
11
Nar.
Fl.
Ob.
Cl.
Bsn.
p

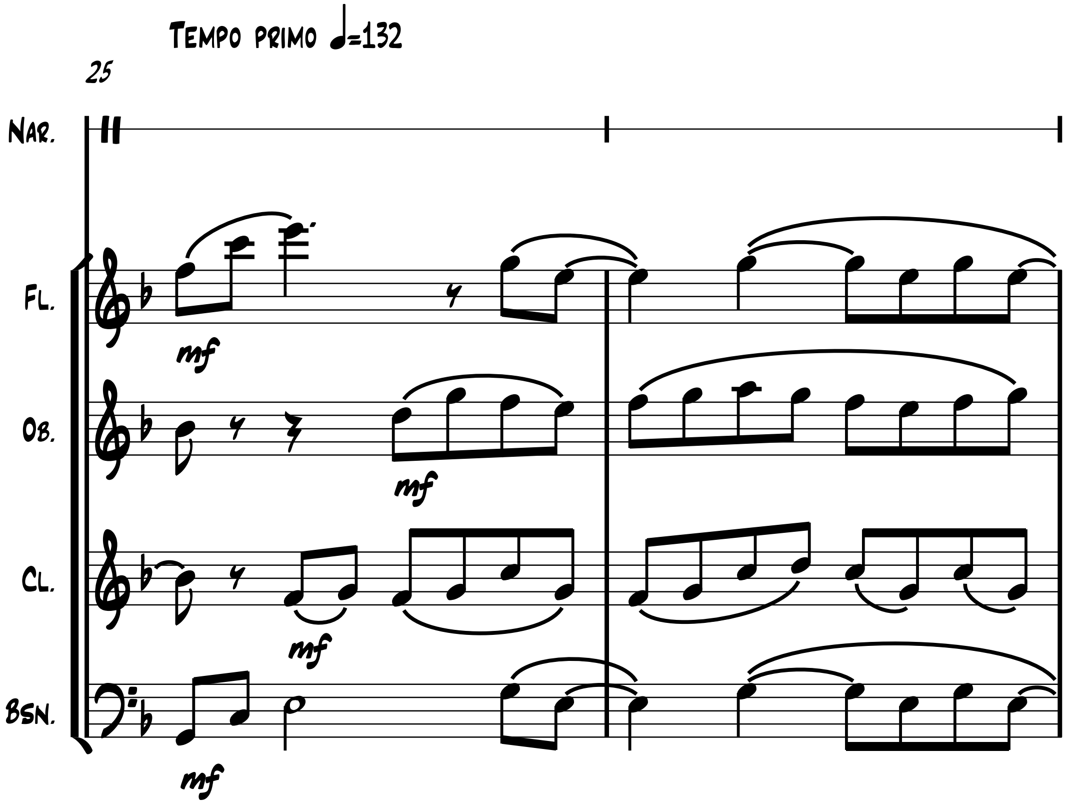
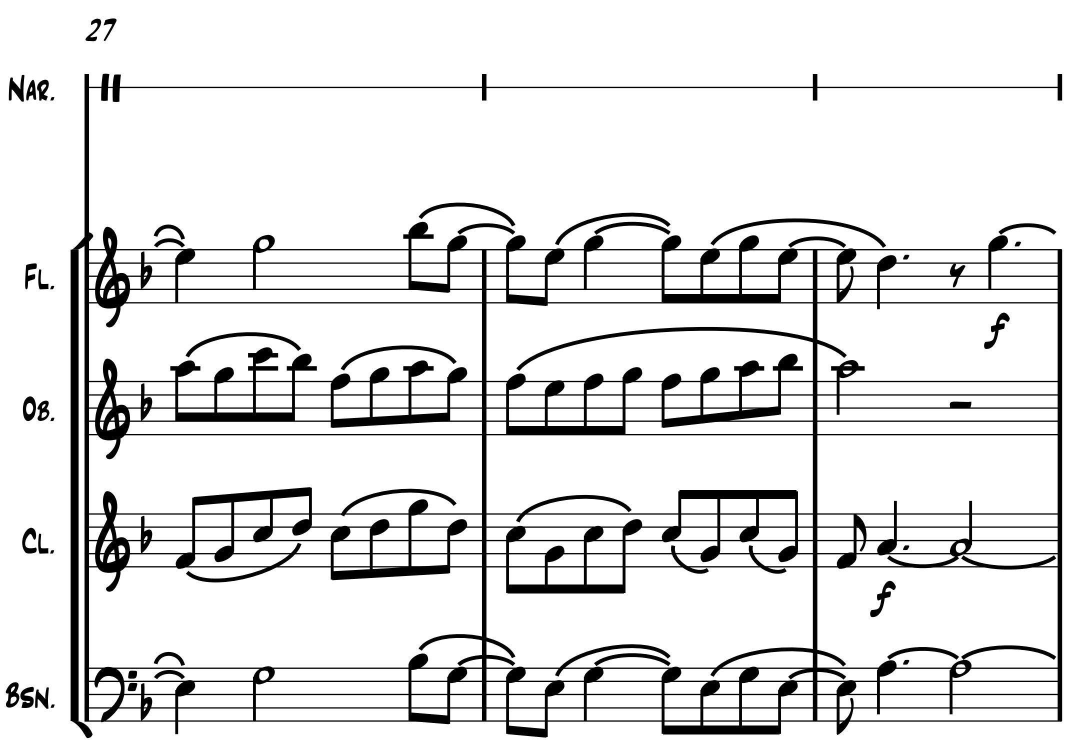

13
rit.
Nar.
Fl.
Ob.
Cl.
Bsn.


15
Nar.
Fl.
Ob.
Cl.
Bsn.

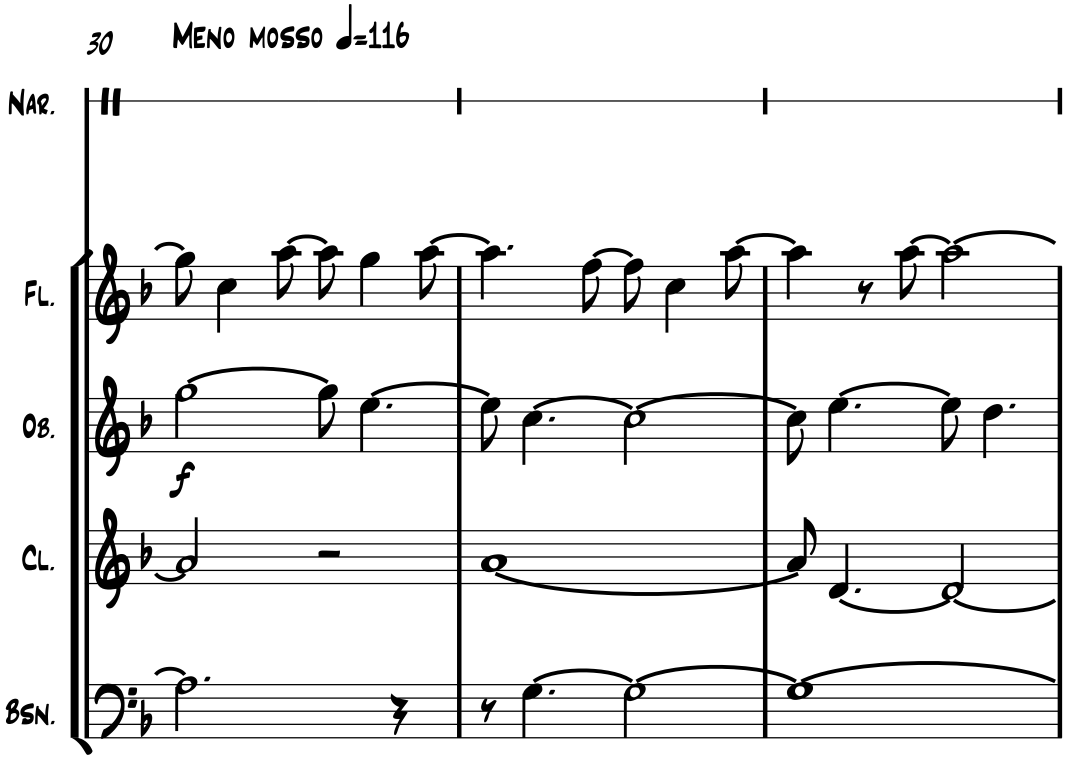


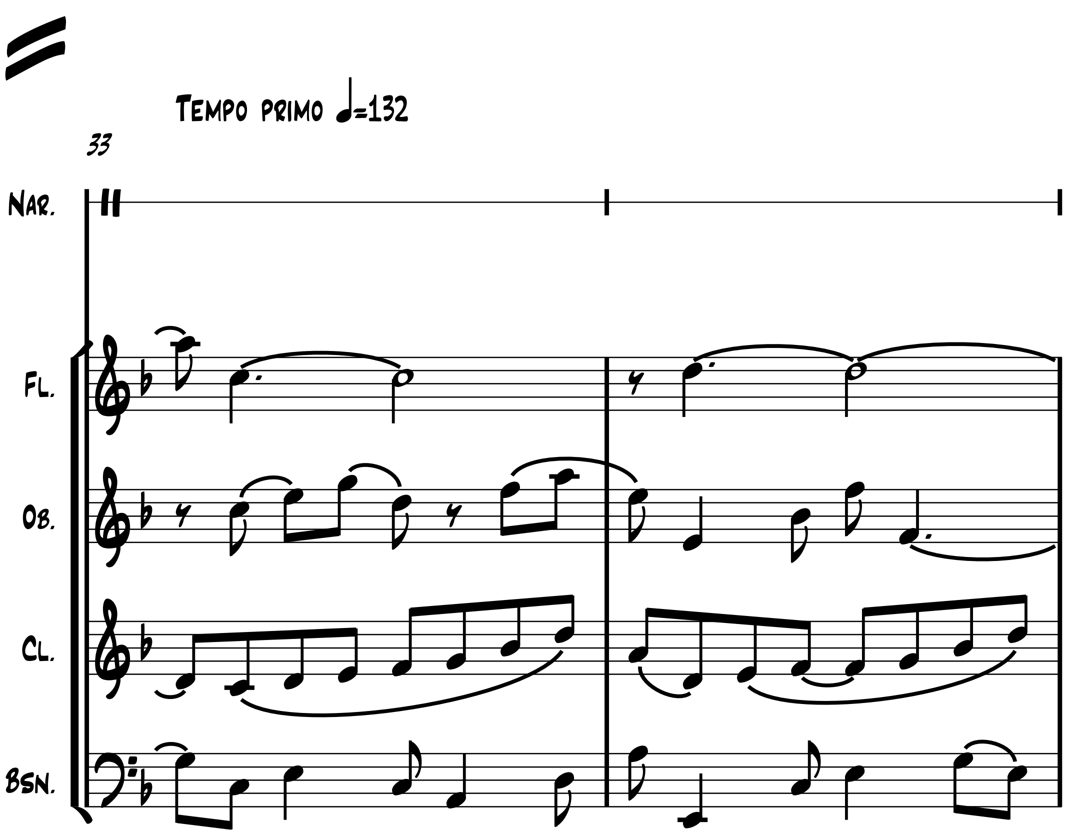
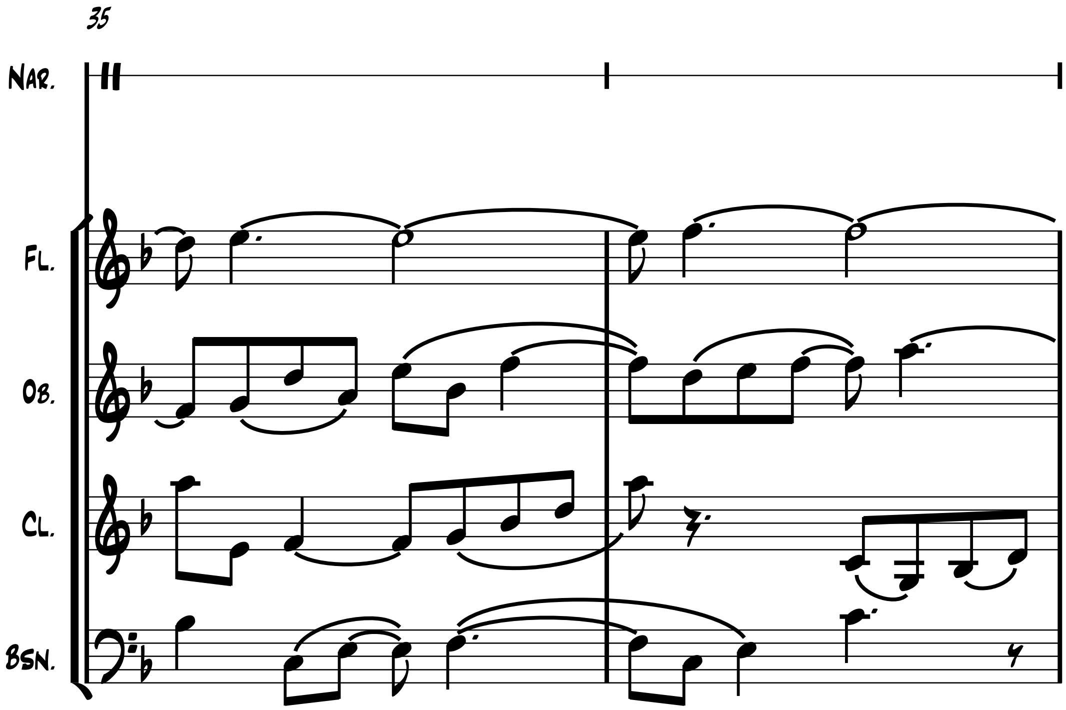
=120
17
Nar.
Fl.
Ob.
Cl.
Bsn.
19
Nar.
Fl.
Ob.
Cl.
Bsn.


21
Nar.
Fl.
Ob.
Cl.
Bsn.
accel.
23
Nar.
Fl.
cresc.
Ob.
cresc.
Cl.
cresc.
Bsn.
cresc.

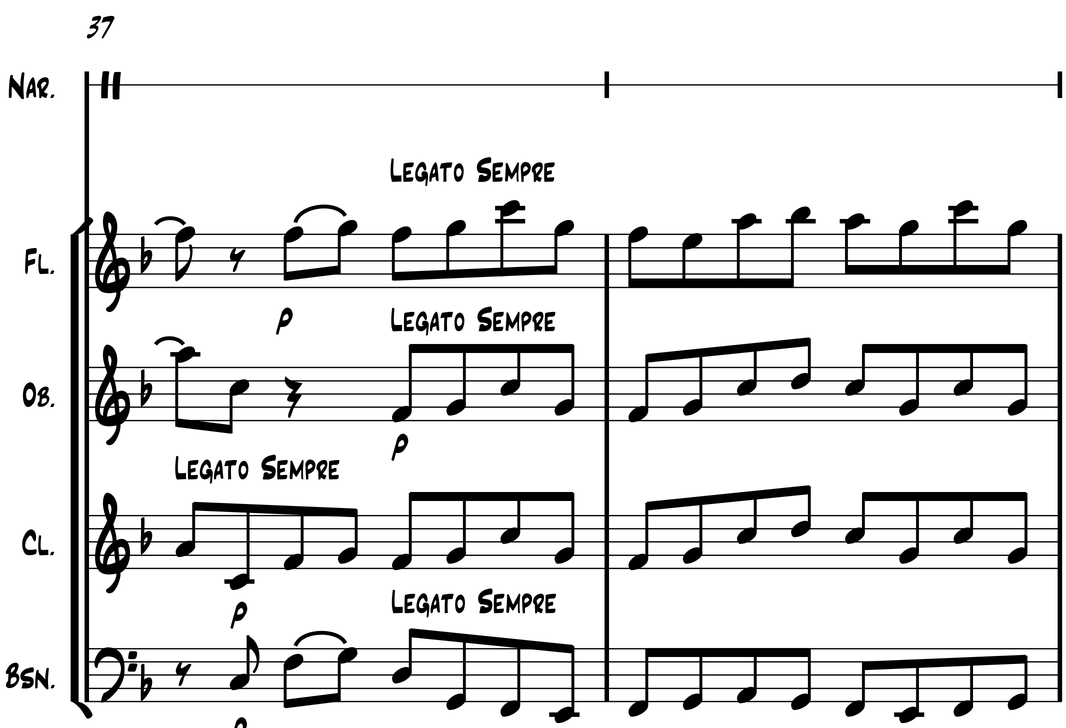
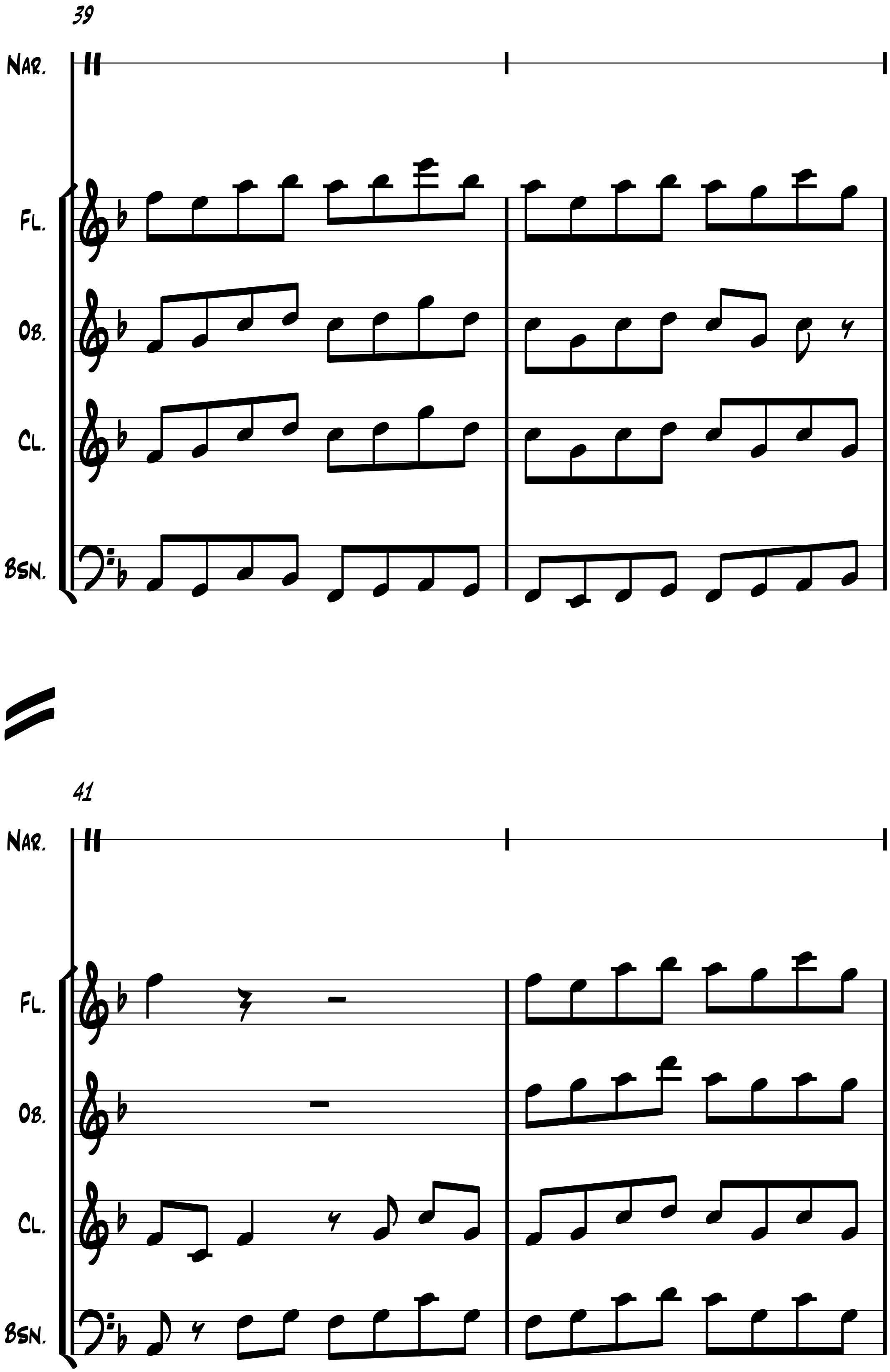

Tempo primo ♩=132
25
Nar.
Fl.
Ob.
Cl.
Bsn.
mf
mf
mf
mf


27
Nar.
Fl.
Ob.
Cl.
Bsn.
f
f
f

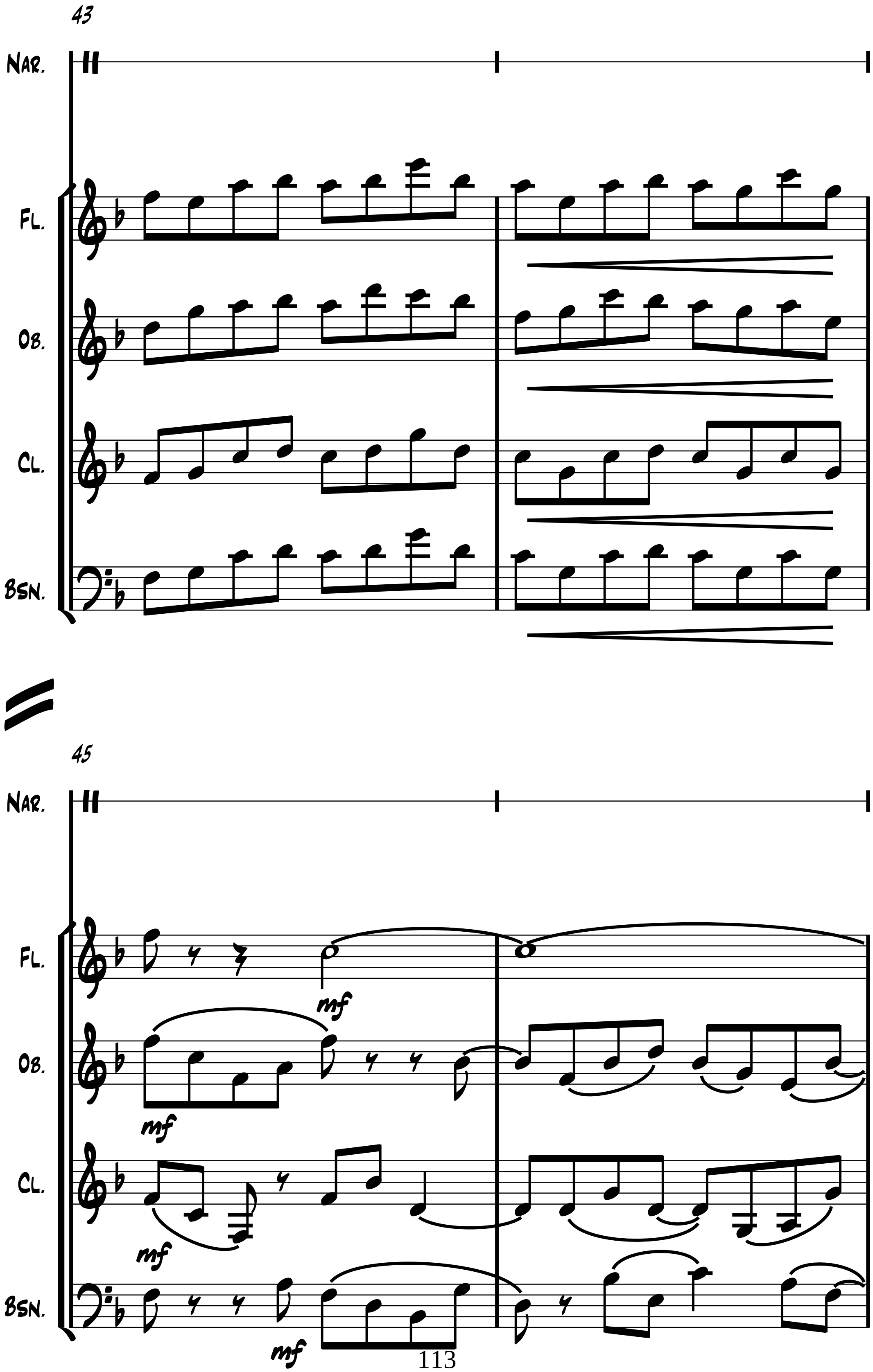
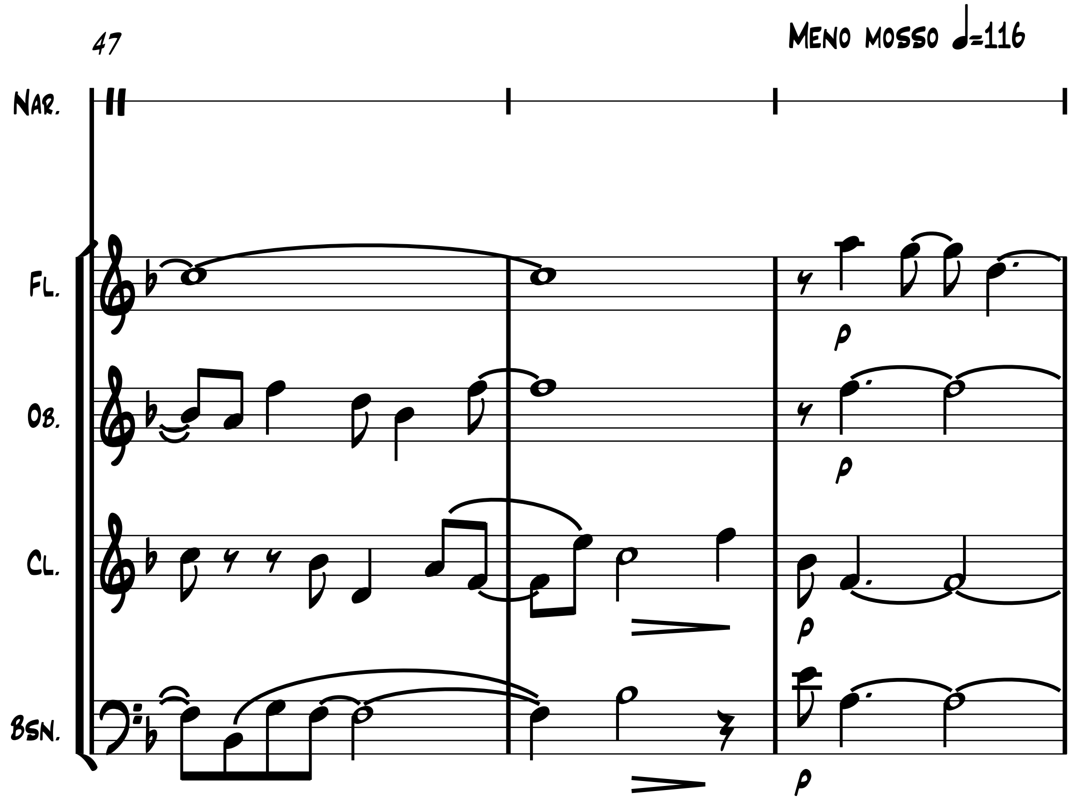

30
Meno mosso ♩=116
Nar.
Fl.
Ob.
f
Cl.
Bsn.

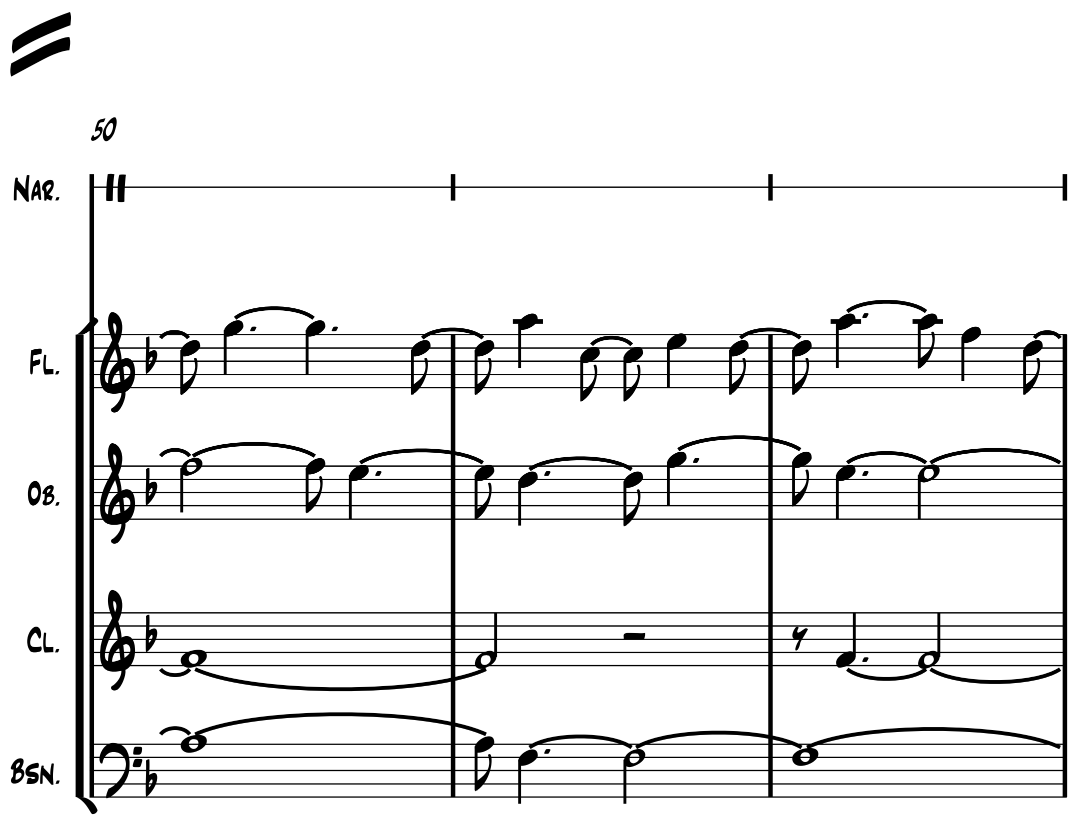

Tempo primo ♩=132
33
Nar.
Fl.
Ob.
Cl.
Bsn.


35
Nar.
Fl.
Ob.
Cl.
Bsn.

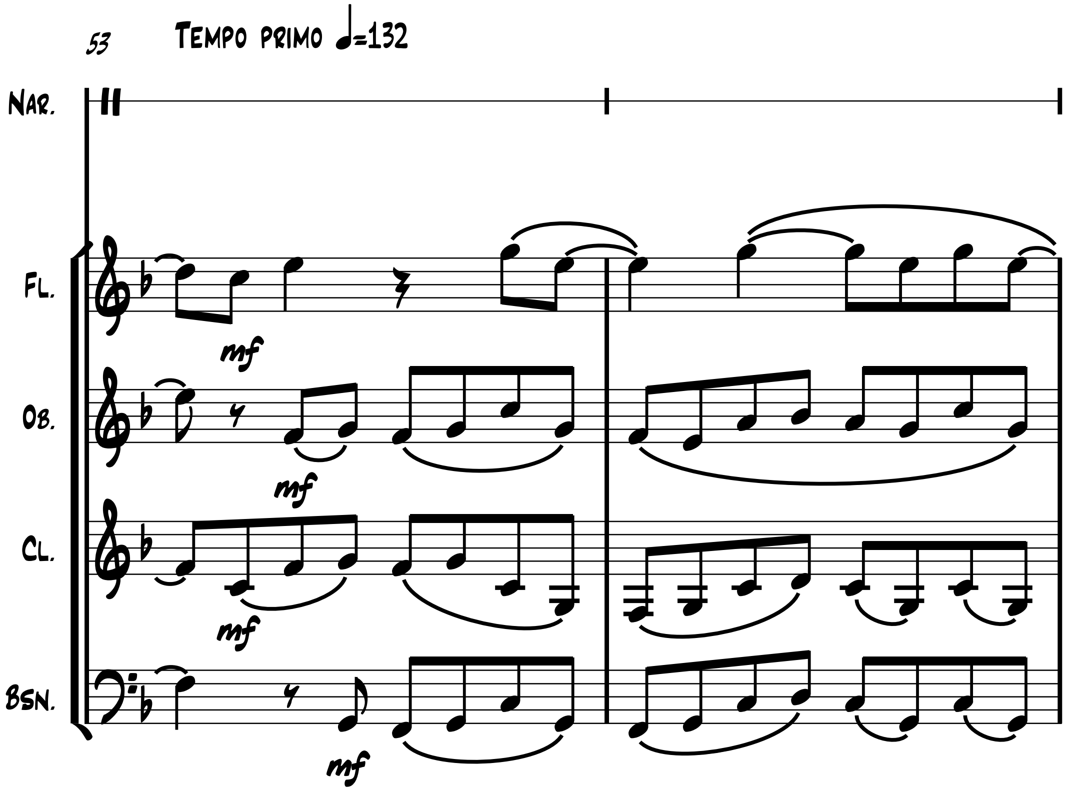

37
Nar.
Legato Sempre
Fl.
p
Legato Sempre
Ob.
p
Legato Sempre
Cl.
p
Legato Sempre
Bsn.
p


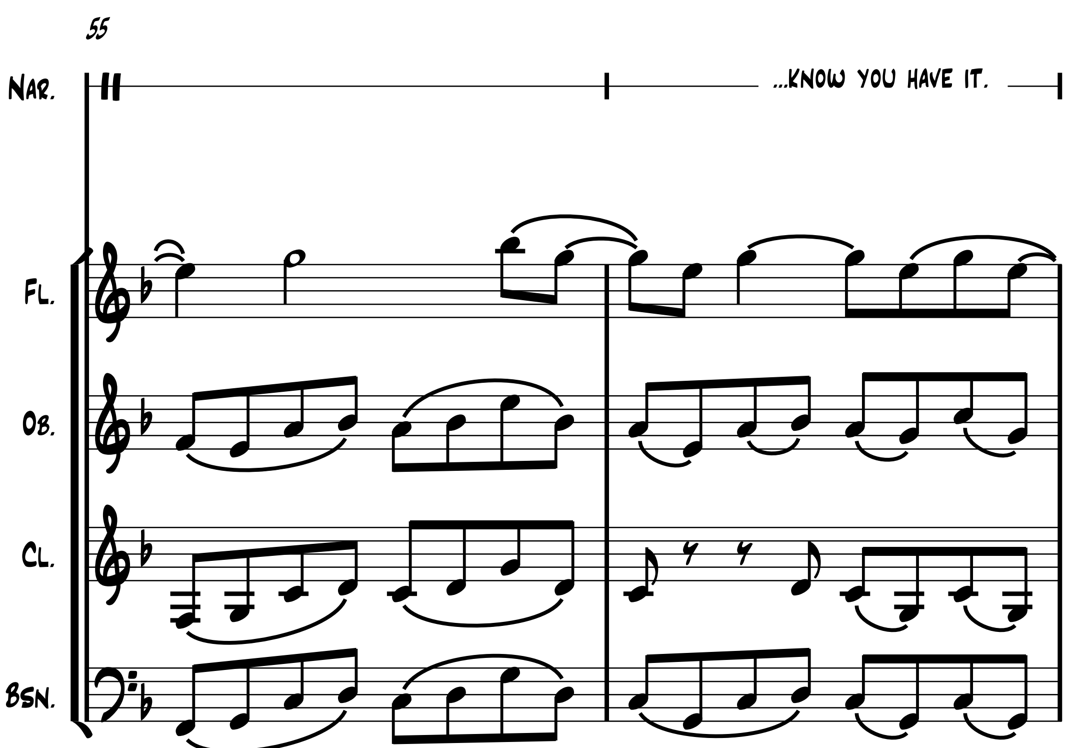
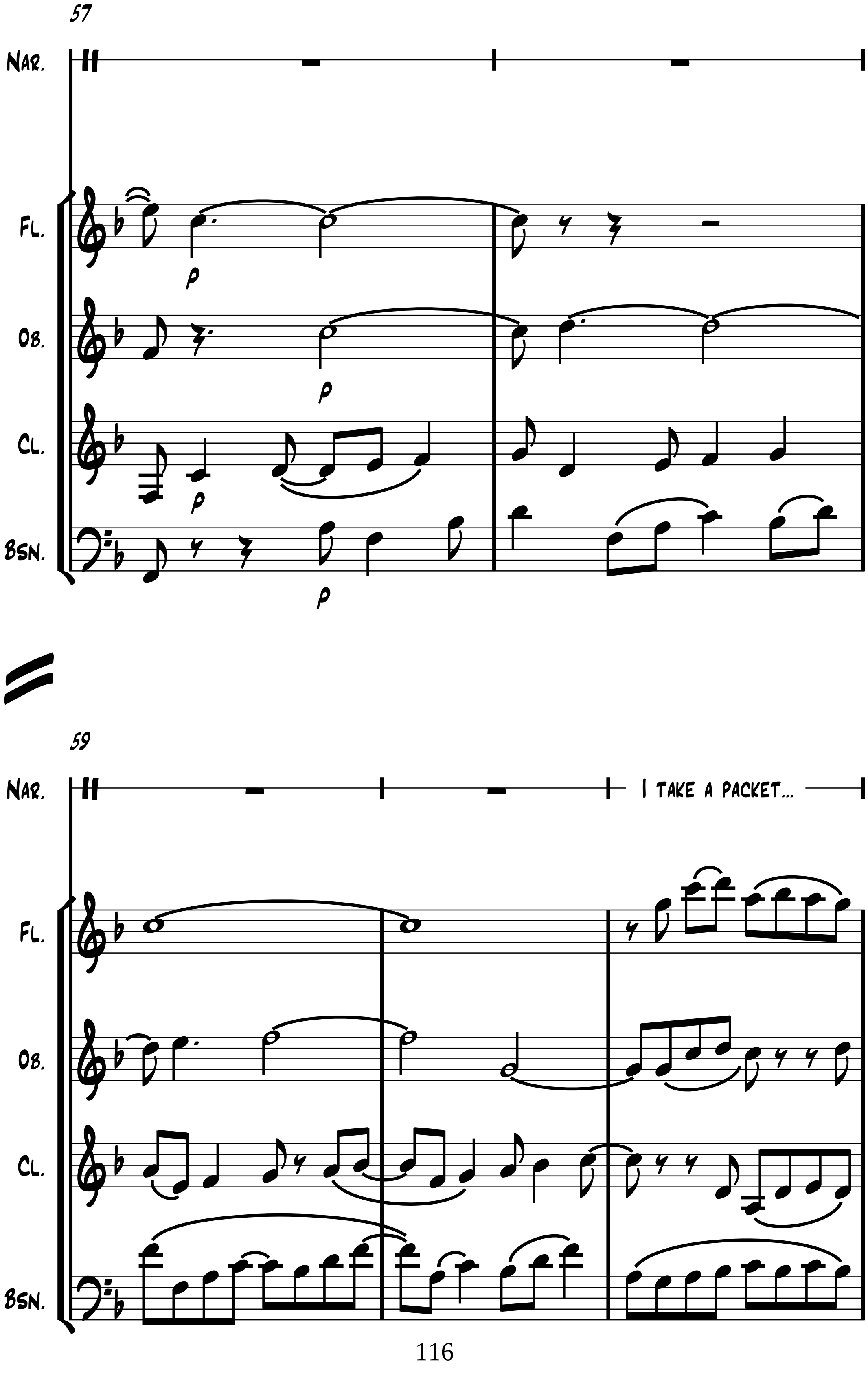
39
Nar.
Fl.
Ob.
Cl.
Bsn.
41
Nar.
Fl.
Ob.
Cl.
Bsn.

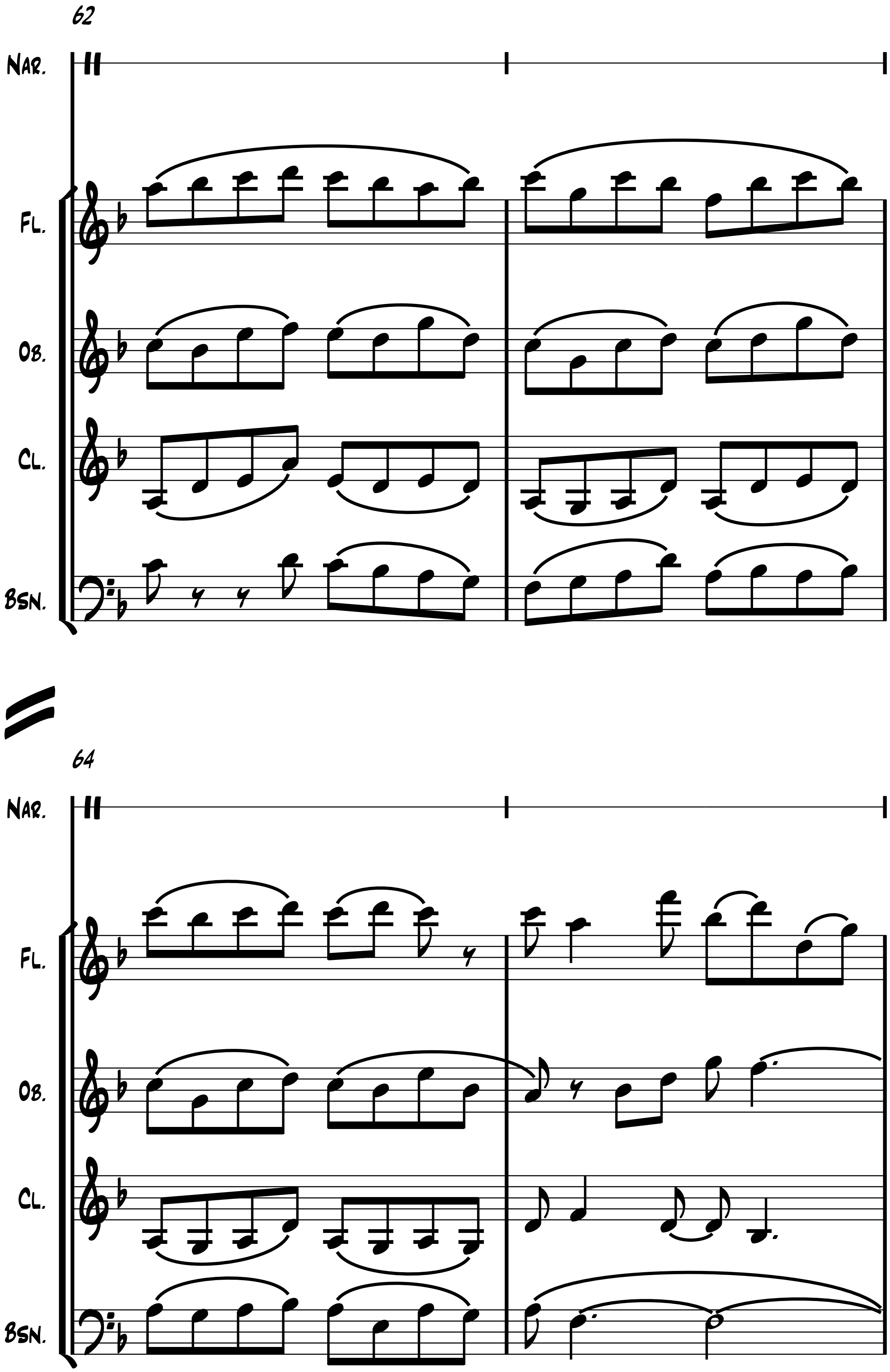

43
Nar.
Fl.
Ob.
Cl.
Bsn.
45
Nar.
Fl.
mf
Ob.
mf
Cl.
mf
Bsn.
mf


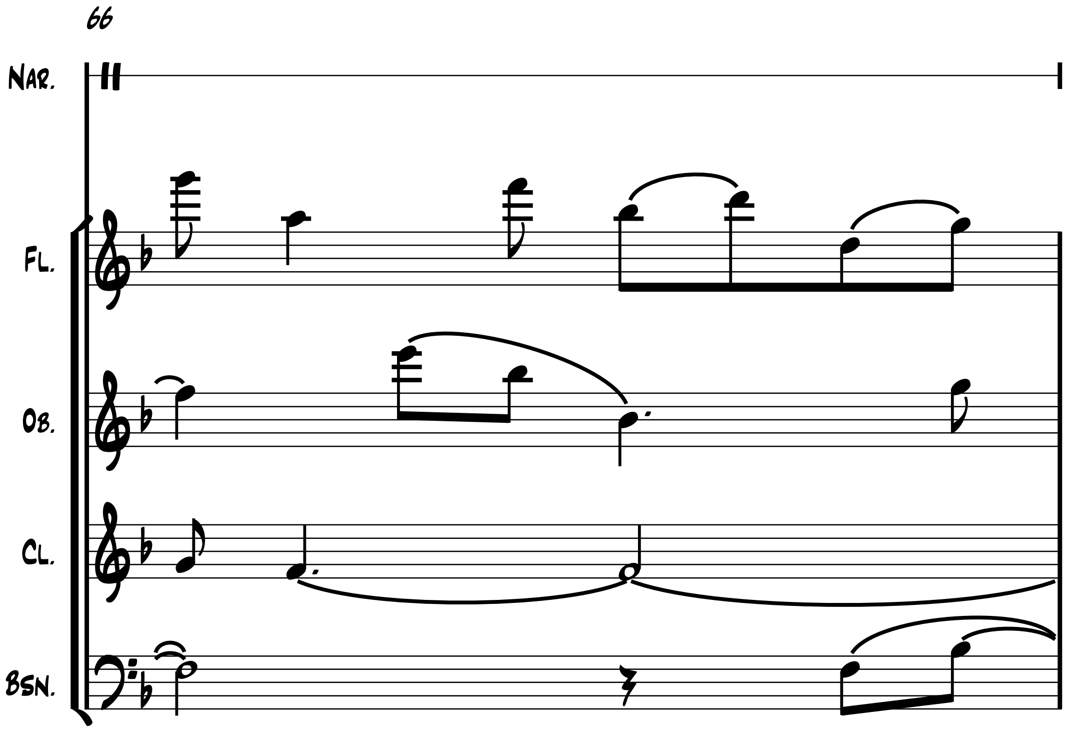
47
Meno mosso ♩=116
Nar.
Fl.
Ob.
Cl.
Bsn.
p
p
p
p


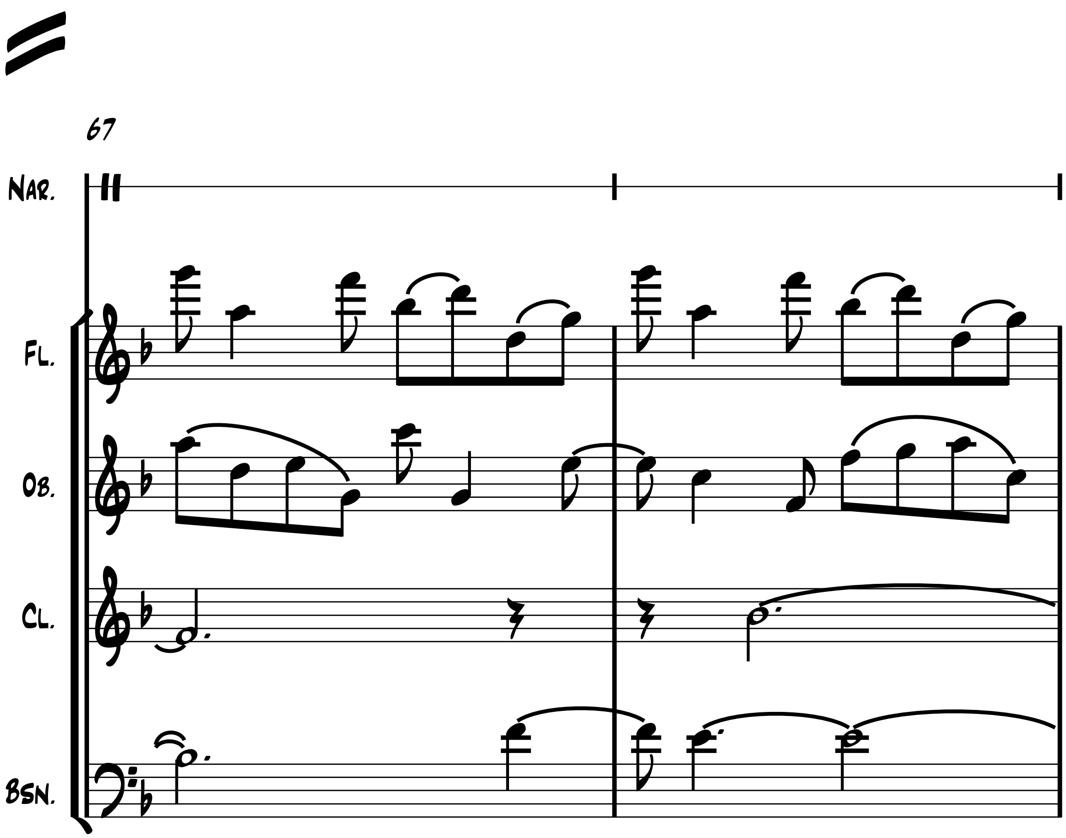
50
Nar.
Fl.
Ob.
Cl.
Bsn.

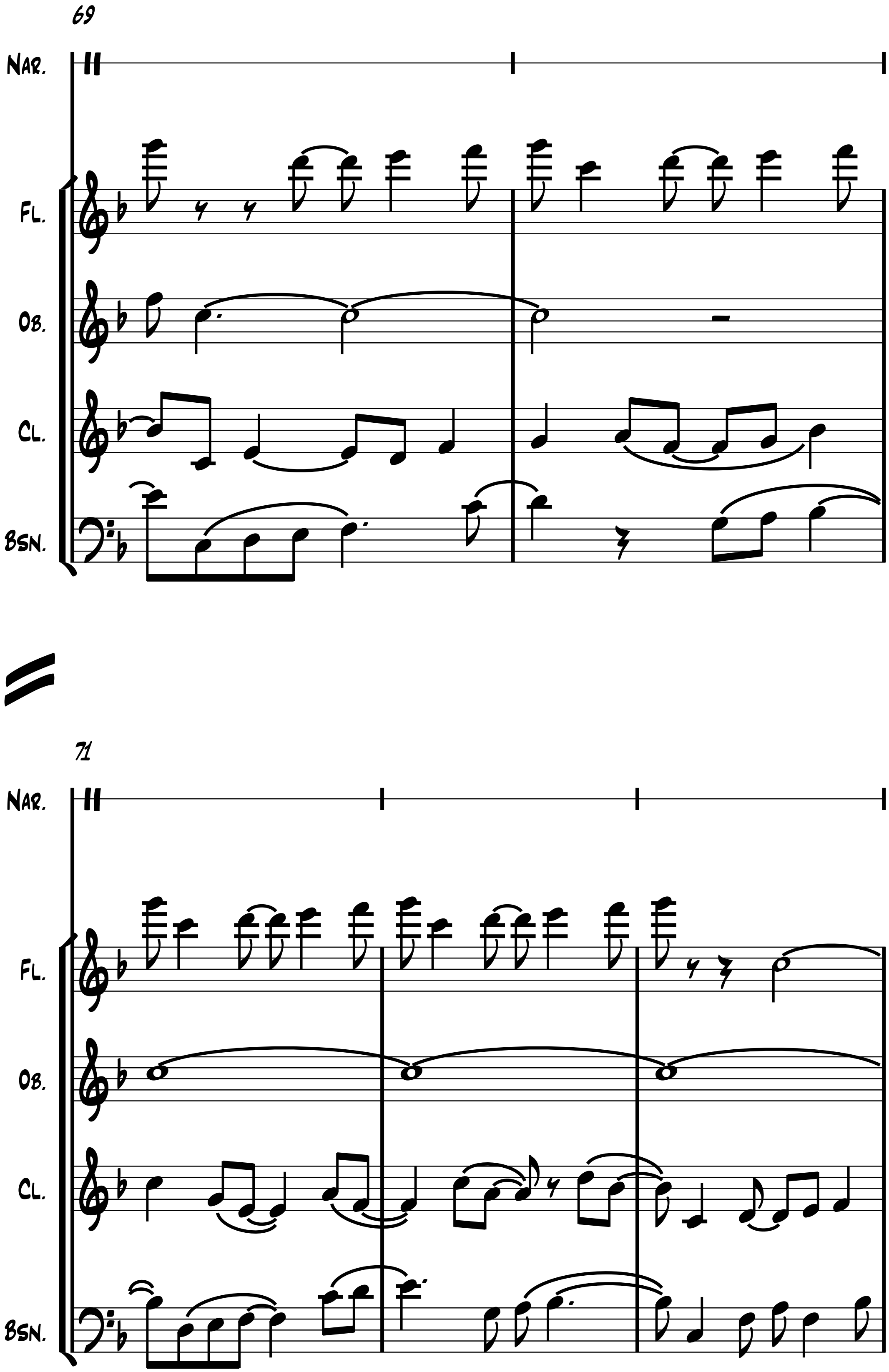

53
Tempo primo 𝅗𝅥=132
Nar.
Fl.
Ob.
Cl.
Bsn.
mf
mf
mf
mf

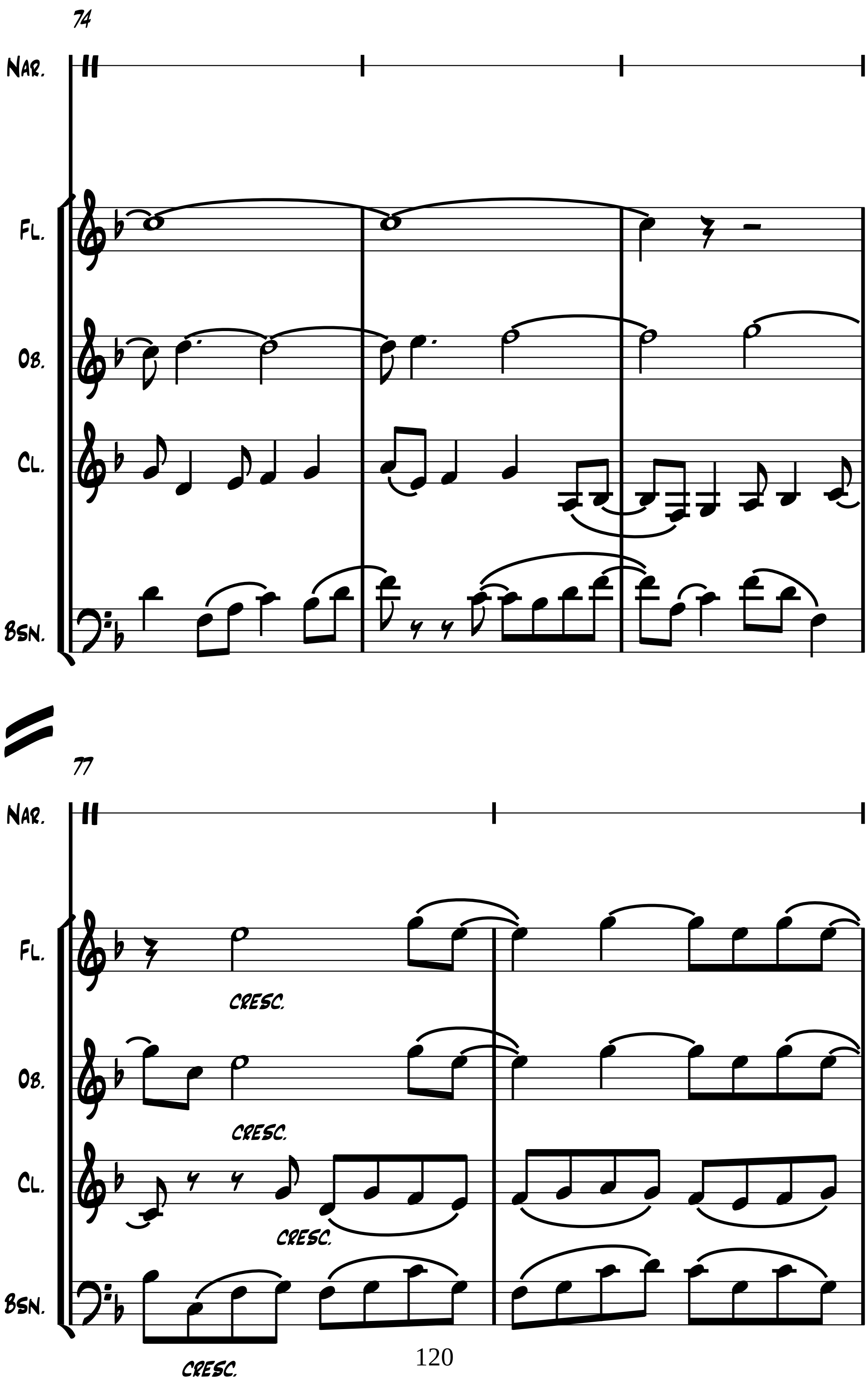

55
Nar.
...know you have it.
Fl.
Ob.
Cl.
Bsn.


57
Nar.
Fl.
Ob.
Cl.
Bsn.
p
p
p
p
59
Nar.
I take a packet...
Fl.
Ob.
Cl.
Bsn.


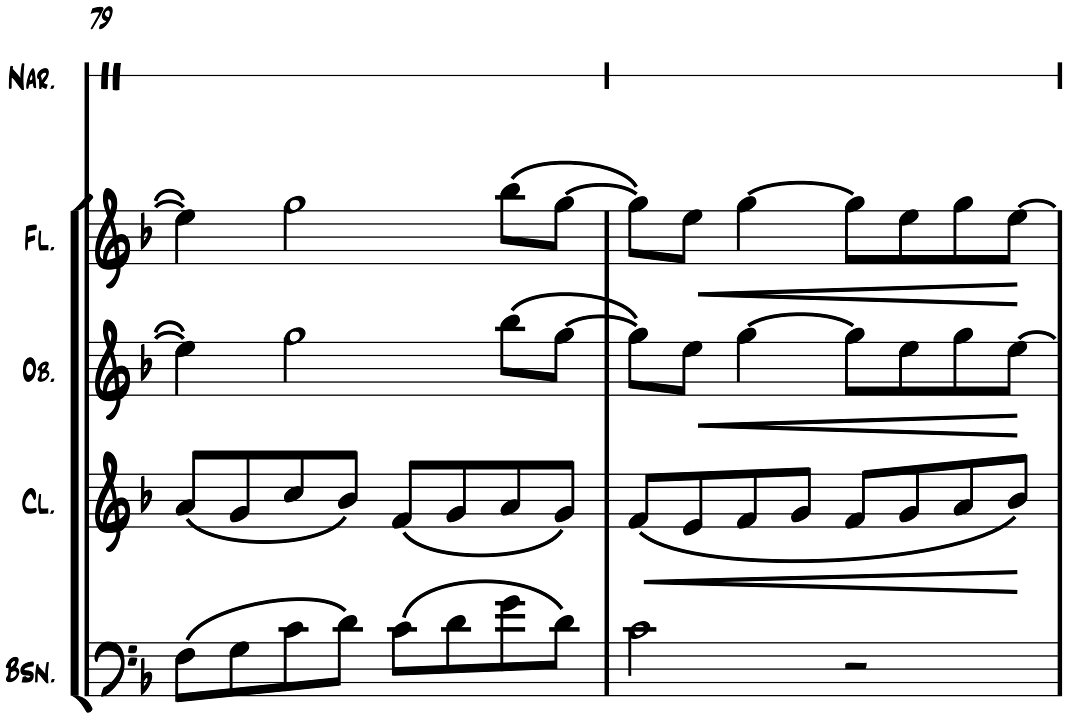
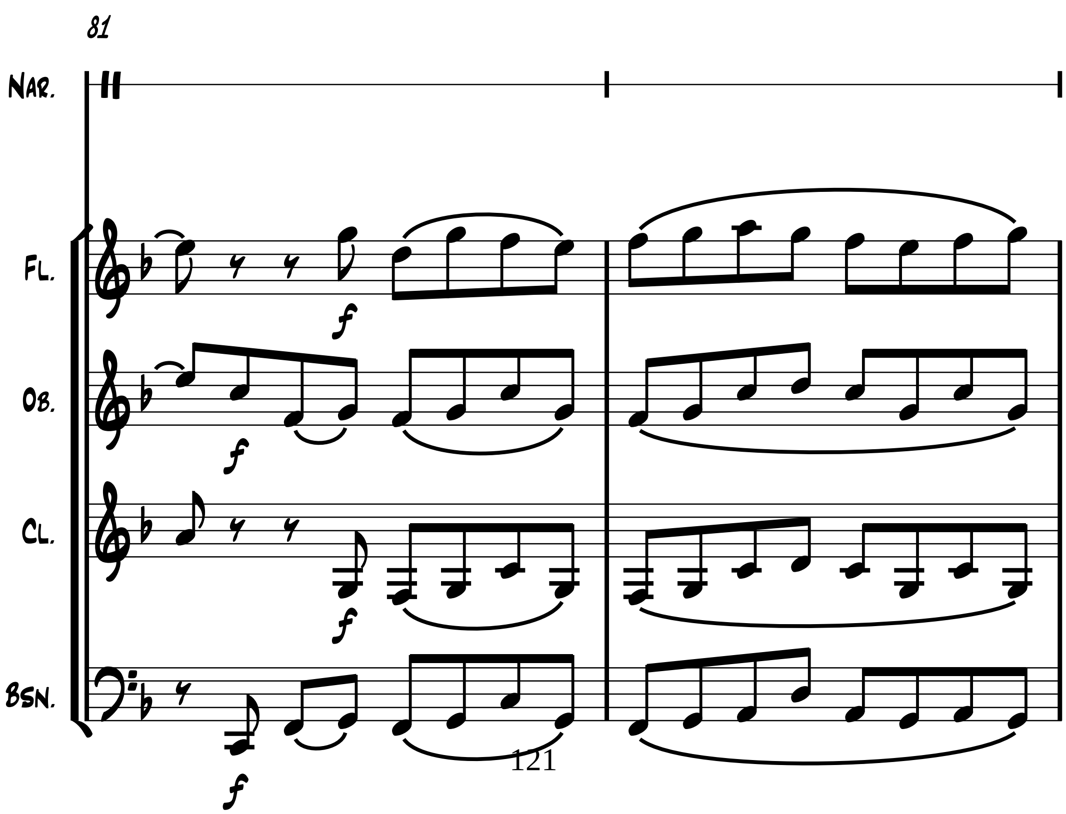
62
Nar.
Fl.
Ob.
Cl.
Bsn.
64
Nar.
Fl.
Ob.
Cl.
Bsn.


66
Nar.
Fl.
Ob.
Cl.
Bsn.


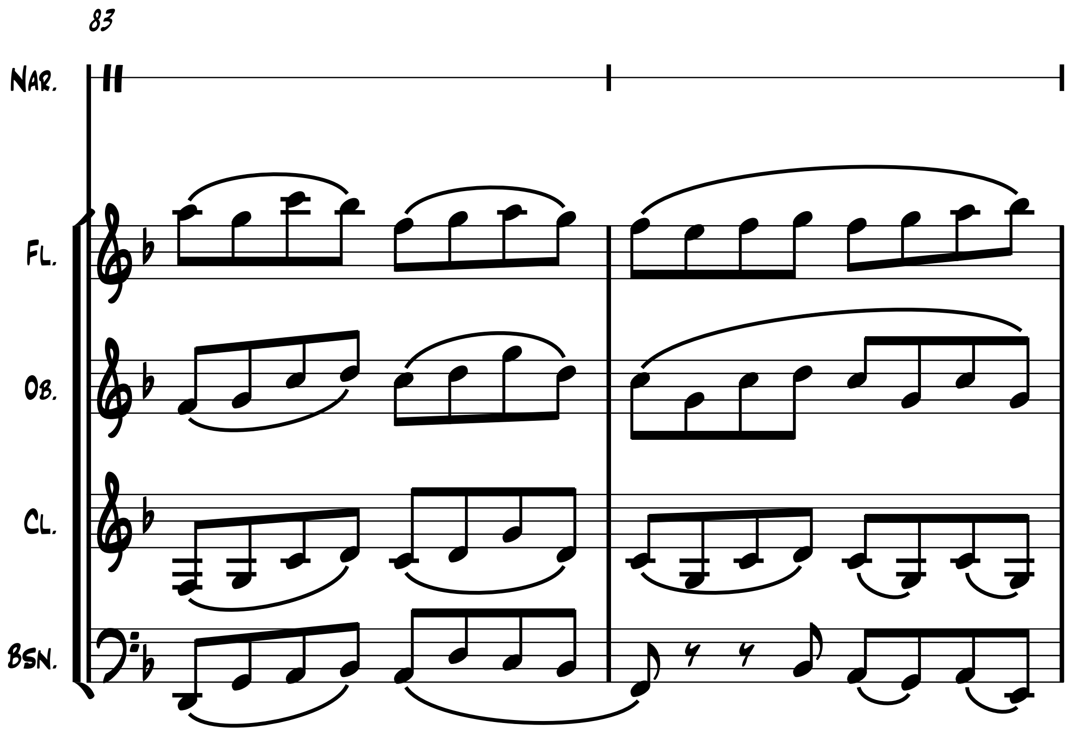
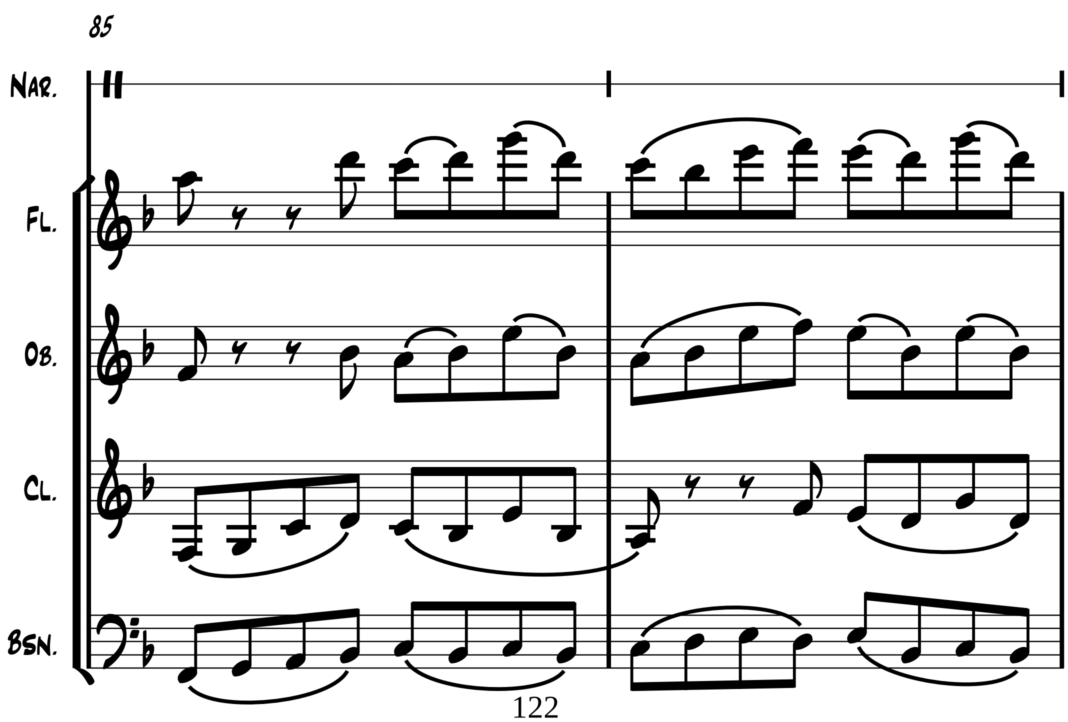
67
Nar.
Fl.
Ob.
Cl.
Bsn.


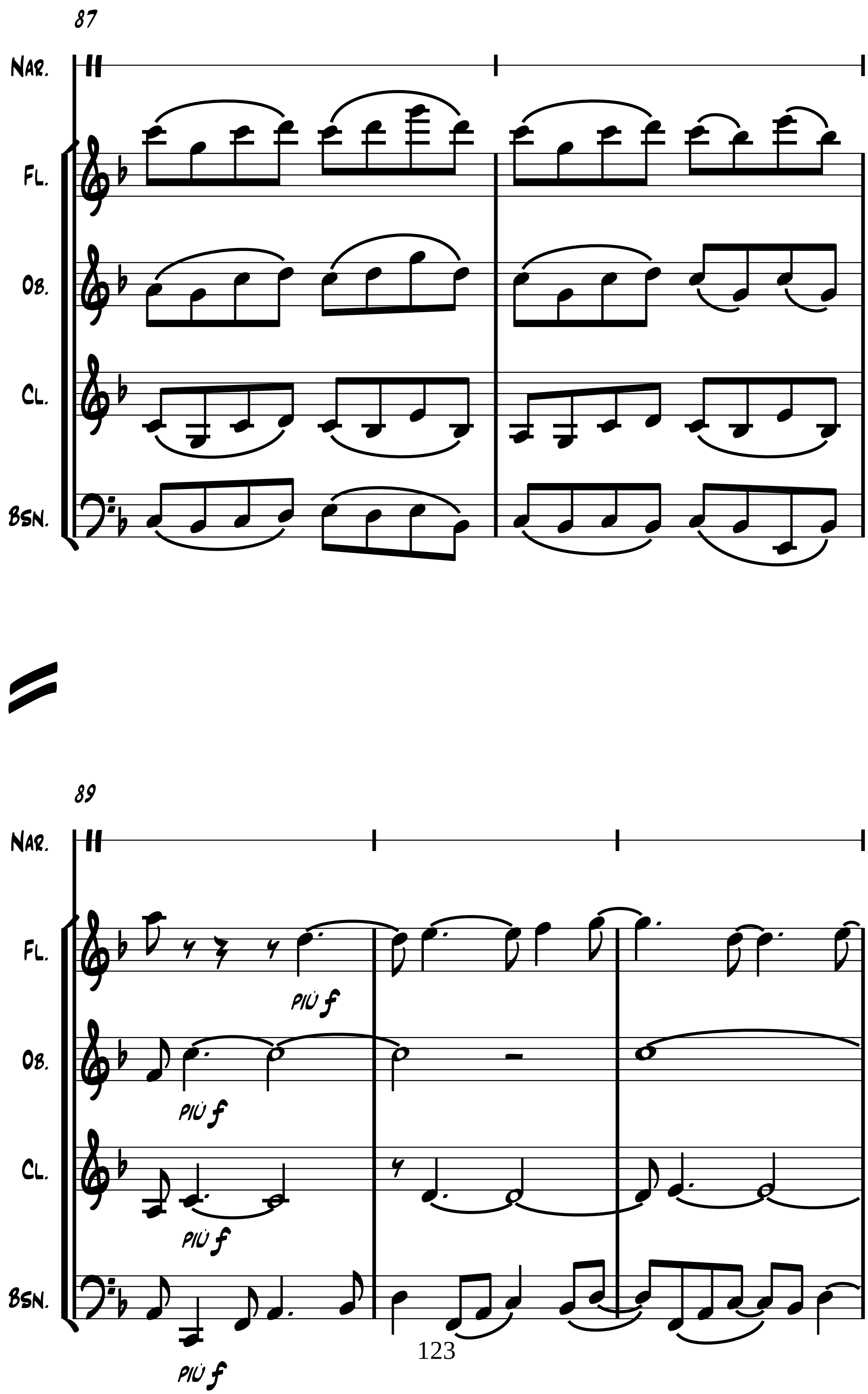
69
Nar.
Fl.
Ob.
Cl.
Bsn.
71
Nar.
Fl.
Ob.
Cl.
Bsn.


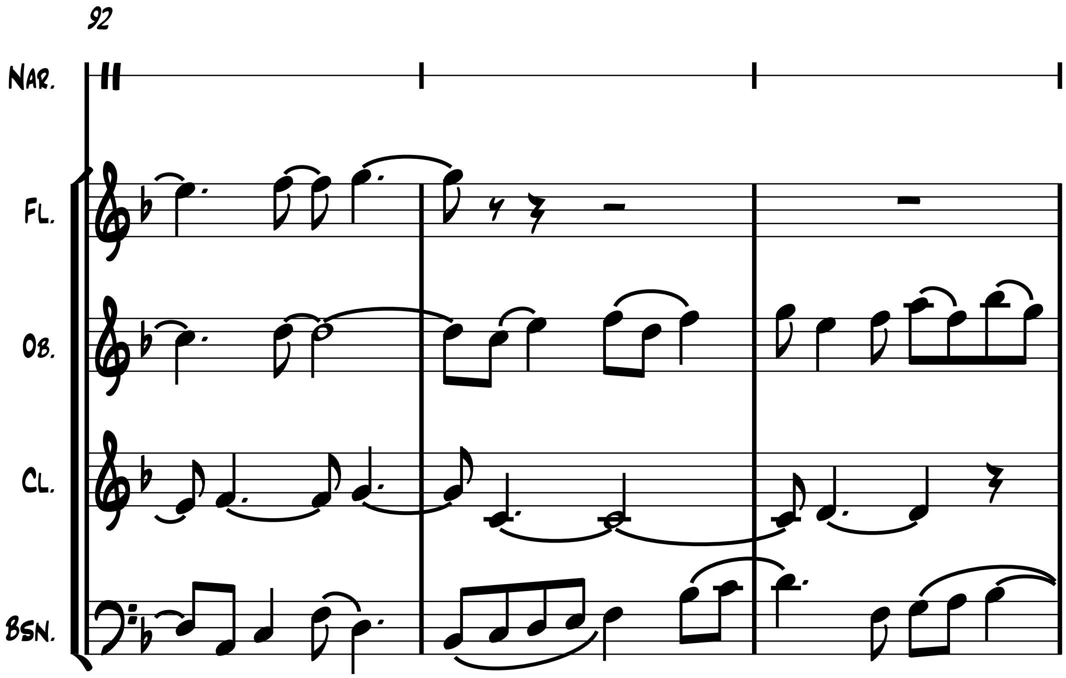
74
Nar.
Fl.
Ob.
Cl.
Bsn.
77
Nar.
Fl.
cresc.
Ob.
cresc.
Cl.
cresc.
Bsn.
cresc.

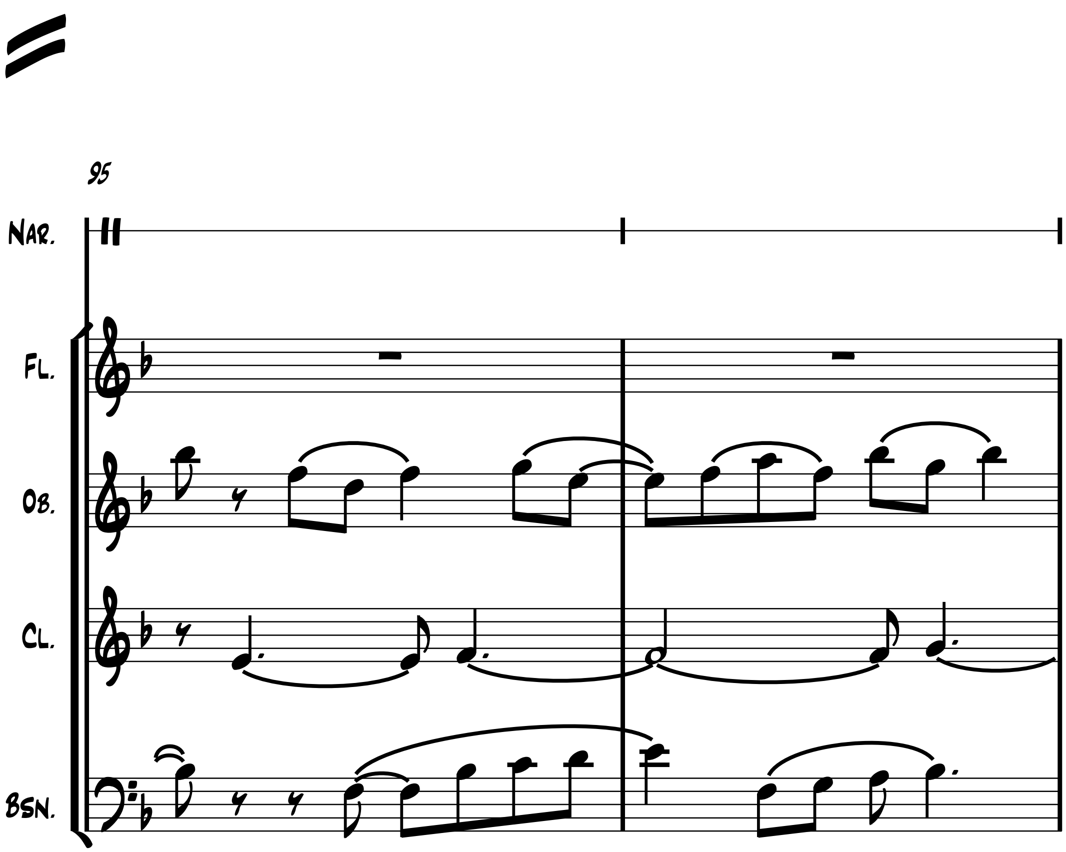

79
Nar.
Fl.
Ob.
Cl.
Bsn.


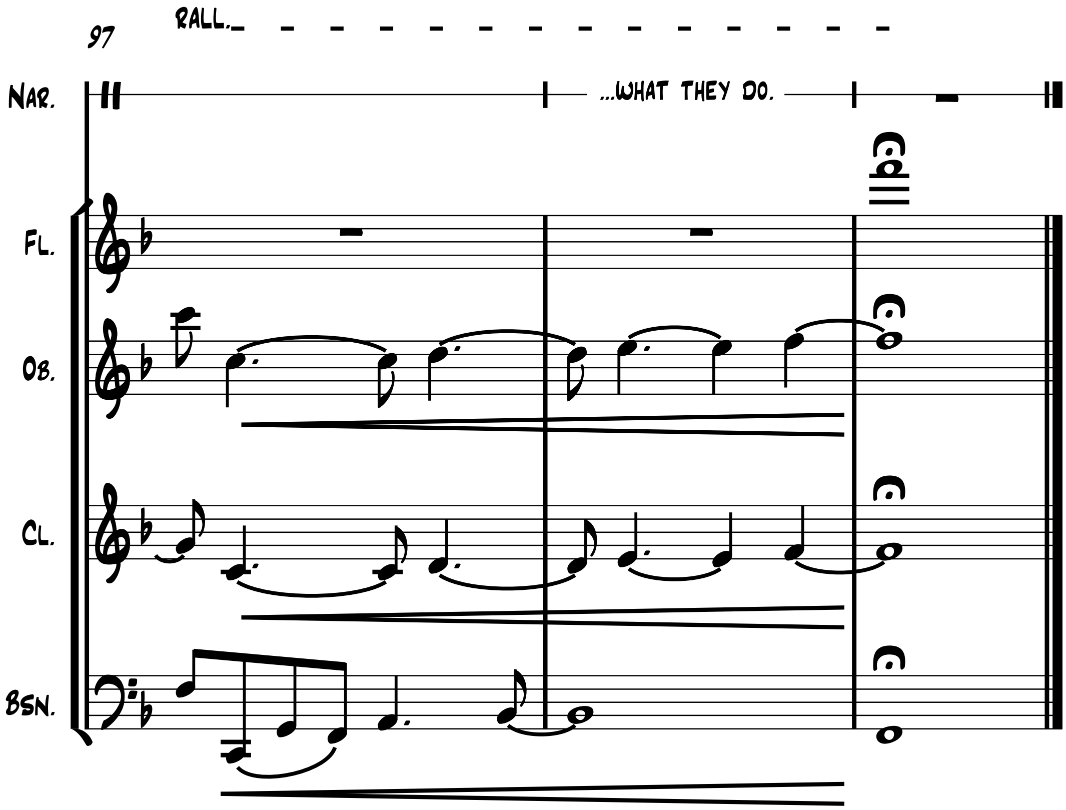
81
Nar.
Fl.
f
Ob.
f
Cl.
f
Bsn.
f


83
Nar.
Fl.
Ob.
Cl.
Bsn.


85
Nar.
Fl.
Ob.
Cl.
Bsn.


87
Nar.
Fl.
Ob.
Cl.
Bsn.
89
Nar.
Fl.
più f
Ob.
più f
Cl.
più f
Bsn.
più f


92
Nar.
Fl.
Ob.
Cl.
Bsn.


95
Nar.
Fl.
Ob.
Cl.
Bsn.


97
rall.
Nar.
...what they do.
Fl.
Ob.
Cl.
Bsn.

# APPENDIX E

“Sarajevo Marlboro” excerpts by Miljenko Jergovic

Know why you should never bury people in a valley? Because a graveyard needs to be located on a hill somewhere above town. Just imagine you're climbing up the slope because you want to rest your eyes perhaps, or walk among the tombs flicking through the album of headstone photographs. Let's say you meet a stranger idling through the deep grass and he expresses an interest in the life story of a person buried up there - well, there you have it! At least if you're on a hillside, you don't just have to regurgitate the story. You can actually map out the life history of the deceased as it moved through the downtown area, from shop to bar towards the grave.

When I was digging a grave for Salem Bicakcija, who was killed in the road by a sniper, an American journalist came to interview me. Perhaps he'd heard that I lived in California for a while, and had seen the world, spoke languages and knew important people. But now I was working as a gravedigger again, so perhaps he thought that I might be able to explain to him what had happened to the people of Sarajevo. So I'm digging away, and he's asking me lots of questions. He wants to know everything, he says. 'About the living or the dead?' I ask. 'Both,' he replies.

I point out that you can't talk about the living and the dead at the same time, because the dead have their lives behind them while the living don't know what's just around the corner, and in what way it could spoil or ruin their lives. In other words, it's much harder for the living, or so I tell the American, because they have no idea where their grave will be - in the valley or on the slope - or if anybody will remember whether they walked happily or unhappily through the dunjaluk. The American asks me to explain what I mean by the dunjaluk. I give him a wry look, because I really don’t know the English word for it. In the end I laugh and say, ‘It means something like "all over the world".’ For some people, of course, ‘all over the world’ is just the distance between Marijindvor and Bascarsija, and for others it’s five continents and seven seas. You end up happy or you don’t - and that’s all.

The American nods his head. I can tell that he doesn’t understand or even care what I’m saying, but I don’t take offence. Why should I? I like to have a chat while I’m digging; it helps to pass the time. He asks me if I’m sorry that I ended up in Sarajevo under siege after having been round the world three times. I tell him that I didn’t end up here. I was born here – and God forbid that I’d ended up dead and buried anywhere else. Who on earth would remember me, or speak about me in respectful tones? Besides which, the graveyards in the rest of the world, and especially in America, are not like the ones in Sarajevo. Elsewhere they line the dead up in rows like soldiers in uniform, with identical headstones, as if their souls had been cast from a mould. The American continues to nod his head. I say that he shouldn’t hold it against me if I utter disparaging remarks about his country. But then the idiot asks me if I’m ready to die now in Sarajevo. I tell him that I’ve thought up hundreds of ways to stay alive, and I like all of them. Each one reminds me of the joys and pleasures of my life, because nobody’s happier than me when I escape a shell on my way here to dig graves in this beautiful spot for the unlucky ones. I know that the dead used to celebrate being alive too, and that they just happened to lose a life the way some people lose a pinball at the end of the game, having scored a hundred points a hundred times – you could have scored more, but... you didn’t. Life is only valuable because you know you have it.

I take a packet of cigarettes out of my pocket. ‘See this? I begin. D’ you know why the packet is completely blank?’ He shakes his head. ‘It’s because there isn’t anywhere in Bosnia to print the brand names and logos. I bet you think we’re poor and unhappy because we don’t even have any writing on our cigarettes. That’s what you think, isn’t it? Know why? Because you haven’t a clue where to look.’

I begin to unwrap the packet because I know that something is printed on the inside: it might be the label from a box of soap or a detail from a movie poster or part of an advertisement for shoes. I’m very curious to find out what’s inside – I make a point of checking – and it’s always a surprise. The American is curious too, but he has no idea what I’m doing. At last I undo the cigarette packet to reveal a Marlboro wrapper – the old brand from Sarajevo.* The American is nonplussed but I swear under my breath. I don’t know what else to say. Whatever I say, he’ll just think, ‘Look at these mad people! They turn cigarette wrappers inside out, then tear them apart to see what cigarettes they’ve bought. If you want my opinion, the people here are just like their packs of cigarettes: everything is back to front – what they say and what they think and what they do.’

APPENDIX F

"Random Poetry" by Igor Karaca

*****

The silver hand shattered the

sky:

An elegy for the death of people

Driven from their homes,

killed

in the swirling dust

hungry and sick

who had no graves but only epitaphs.

*****

Feel fear;

if I survive

memories will become history.

The entire object of my

existence

descends into chaos.

Seldom the ghosts come back.