

MASSACHUSETTS INSTITUTE OF TECHNOLOGY  
PROJECT MAC

Artificial Intelligence  
Memo. No. 141

September 1967.

EUTERPE-LISP: A LISP System with Music Output.

by

Stephen Smoliar.

EUTERPE (A.I. Memo. No. 129), was designed as a "real-time music program" which would interpret music described as "voice-programs" in DDT. These voice-programs consisted of note words, description of tones to be sounded, and control words which determined the parameters of pitch, tempo, articulation and wave form and allowed for a subroutine feature and transfer within the voice-program. It had been hoped that complex musical forms could be described in terms of a few collections of note words and sequences of control words.

However, musical variation and development is more subtle than the developmental power of these control words. Any transformation of musical materials may be expressed as a LISP function; therefore, the control words were abandoned and EUTERPE was linked to LISP. The voice-programs would be written and loaded by LISP and played by EUTERPE. The principle function in the system is LOAD which takes two arguments: 1) an absolute location in core and 2) a list of note words. The note words are translated into EUTERPE-readable code and loaded into the proper voice program. The addresses of the first location of each of the six voice programs are SETQed by the system with the names VOICE1, ..., VOICE6. The value of LOAD is the next file word in core, so a series of lists may be loaded by the following bootstrapping procedure:

```
(SETQ LOC (LOAD VOICE1 LIST1))
(SETQ LOC (LOAD LOC LIST2))
```

. . .

An example is given at the end of this Memo.

A note is expressed as a dotted pair; the CAR denotes pitch, the CDR denotes duration. If the CDR is NIL, the duration is assumed to be the same as the preceding note (this is the same convention as EUTERPE; see example). The symbols are the same as in EUTERPE, but they are EXPLODED into lists.

Hence middle C is now

```
(K C)
```

and a triplet-sixteenth note is

```
(16 T 3).
```

There are also a few "control-lists" which function as their analogs in EUTERPE; these are

(CANCEL n) (CDR may be NIL)  
 (START n)  
 (RELTEM n)  
 (TEMPO n)  
 (ARTIC n)  
 (WAVE n)  
 (FINE)

Finally, there are functions written into the system which may be used in describing music

---

MA	Major scale starting on pitch given as argument (ascending).
Example: (MA (QUOTE (K B)))	has the value
	((K B) (L C S) (L D S) (L E) (L F S) (L G S) (L A S) (L B))
NA	Natural minor scale (ascending).
HA	Harmonic minor scale (ascending).
ME	Melodic minor scale (ascending).
NUP	(NUP N I) is the pitch I half-steps above pitch N.
NDOWN	I half-steps below pitch N.
TRANSTONAL	(TRANSTONAL X K N) transposes a list of pitches, X, to begin on a new note N, in tonality K = (( <sup>MAJOR</sup> <sub>MINOR</sub> ) pitch)
TRANSCOPE	(TRANSCOPE X N) is a rigid transposition.
TR	(TR X N) takes a full list of input for EUTERPE and raises all pitches N half-steps.
INVERTONAL	(INVERTONAL X K) inversion of list of pitches, X, with respect to tonality K.
INVERT	(INVERT X) is rigid inversion.
RETROG	(RETROG X) is retrogression.
ROTN	(ROTN X N) rotates a list of pitches N steps.

---

Example: (ROTN (QUOTE ((J E) (J D) (J C)) 1)  
 has the value: ((J D) (J C) (J E)).

CHORD	argument is a list of up to six pitches; sounds as a chord until next input to LISP.
SETA	sounds A; alters tuning constant by numbers typed in, terminated by typing non-number.
TRANSFER	transfers to absolute location; (TRANSFER SETUP) prepares compilation; (TRANSFER PLAY) plays compiled version.

---

The user may also prepare his own function.

SETA is used as follows: The user types in the s-expression (SETA) and the machine sounds the pitch it assumes to be 440 cps. If this note is flat, the user types in a number (as an atom) the machine adds this number to its tuning constant, and it sounds a new pitch. If the note is sharp, the user types in a negative number. Once the machine is "in tune," any non-number (e.g., the atom OK) will terminate SETA and return to LISP.

The following program describes the Canon from Bach's Kunst der Fuge attached at the end of this Memo. The major portion of the program consists of SETQing the necessary thematic elements which are then loaded by the two PROG's at the end.

(SETD IBASE 10.)  
 (SETD T1 (QUOTE ( (ARTIC LEGATO)  
 ((J D) . (8 T 3))  
 ((J E))  
 ((J D))  
 ((J C S))  
 ((J D))  
 ((J E))  
 ((J F))  
 ((J G))  
 ((J F))  
 ((J E))  
 ((J F))  
 ((J G))  
 (ARTIC SLUR)  
 ((J A) . (2 T))  
 (ARTIC LEGATO)  
 ((J A) . (8 T))  
 (ARTIC SLUR)  
 ((J G) . (8 T))  
 ((J F))  
 (ARTIC LEGATO)  
 ((J E) . (8 T))  
 (ARTIC SLUR)  
 ((J F) . (2 T))  
 (ARTIC LEGATO)  
 ((J F) . (8 T))  
 (ARTIC SLUR)  
 ((J E) . (8 T))  
 ((J D))  
 (ARTIC LEGATO)  
 ((J C S) . (8 T))  
 ((J D) . (4 T))  
 (ARTIC SLUR)  
 ((J E) . (8 T))  
 ((J F))  
 ((J G))  
 (ARTIC LEGATO)  
 ((J A) . (8 T))  
 ((J B F) . (4 T))  
 (ARTIC SLUR)  
 ((J B F) . (4 T))  
 (ARTIC LEGATO)  
 ((J C S) . (4 T))  
 (R . (8 T))  
 ((I A))  
 ((I B))  
 ((J C S))  
 (ARTIC SLUR)  
 ((J D) . (8 T))  
 ((J C S))  
 ((I B))  
 (ARTIC LEGATO)  
 ((I A) . (8 T))  
 (ARTIC SLUR)  
 ((J E) . (8 T))  
 ((J C S))  
 ((I B))  
 (ARTIC LEGATO)

```

((I A) . (B T))
(ARTIC SLUR)
((J F) . (B T))
((J E))
((J D) . (4 T))
(R . (B T))
((J B F))
((J A))
(ARTIC LEGATO)
((J G S) . (B T))
(ARTIC SLUR)
((J A) . (B T))
((J F))
((J E))
(ARTIC LEGATO)
((J D) . (B T))
(ARTIC SLUR)
((J F) . (B T))
((J E))
((J D))
(ARTIC LEGATO)
((J C S) . (B T))
)))
(SET2 T2 (QUOTE (
(ARTIC LEGATO)
((J D) . (4 T))
(R)
((I A) . (B T 3))
(PERT ((I B)) (1 . D))
((I A))
(PERT ((I G S)) (1 . D) (3 . D))
((I A))
(PERT ((I B)) (1 . D))
(PERT ((J C) . (B T)) (2 . U))
((I A))
(PERT ((I B)) (1 . D))
(PERT ((J C)) (2 . U))
((J D))
((J E))
(PERT ((J F S)) (2 . D))
(PERT ((J G S)) (1 . D) (2 . D))
((J A))
((J B))
(PERT ((J F)) (1 . U))
((J E))
((J D) . (4 D))
(PERT ((J E) . (B T)) (1 . D))
)))
(SET2 T3 (QUOTE (
(ARTIC LEGATO)
((J F) . (B T))
(PERT ((J E)) (1 . D))
((J D))
((J C))
(PERT ((I B)) (1 . D) (2 . D))
((J C))
((J D))
(PERT ((I B)) (1 . D) (2 . D))
(PERT ((I G S)) (1 . D) (2 . D))
((I A))

```

(PERT ((I B)) (1 . D) (2 . D))  
 (PERT ((I E)) (1 . D))  
 (PERT ((J E)) (1 . D))  
 (PERT ((J C)) (2 . U))  
 ((J D))  
 (PERT ((J E)) (1 . D))  
 (PERT ((I F S)) (1 . D) (2 . D))  
 ((I A))  
 (COM (ARTIC STACO) (2 . (LEGATO)))  
 ((J D) . (4 T))  
 (COM (ARTIC LEGATO) (2 . (LEGATO)))  
 (PERT ((I G S) . (8 T)) (1 . D) (2 . D))  
 (PERT ((I B)) (1 . D) (2 . D))  
 (COM (ARTIC STACO) (2 . (LEGATO)))  
 ((J E) . (4 T))  
 (COM (ARTIC LEGATO) (2 . (SLUR)))  
 ((I A) . (8 T))  
 (PERT ((J C)) (2 . U) (3 . D))  
 ((J D))  
 (ARTIC LEGATO)  
 ((J E) . (8 T))  
 (COM (ARTIC LEGATO) (2 . (SLUR)))  
 ((J F) . (8 T))  
 ((J D))  
 (PERT ((J C)) (2 . U) (3 . D))  
 (ARTIC LEGATO)  
 (PERT ((I B) . (8 T)) (1 . D))  
 (PERT ((J C) . (4 T)) (2 . U) (3 . D))  
 ((J F))  
 ((J D))  
 ((J E))  
 ))  
 (SETQ T4 (QUOTE ( (ARTIC LEGATO)  
 ((J F) . (8 T 3))  
 ((J G))  
 ((J F))  
 ((J E))  
 ((J F))  
 ((J G))  
 ((J A) . (4 D))  
 ((J G) . (15 T))  
 ((J F))  
 ((J E) . (8 T))  
 (PERT ((J C)) (2 . U) (3 . D))  
 ((J D))  
 ((J E))  
 ((J F) . (4 D))  
 ((J E) . (15 T))  
 ((J D))  
 ((J C S) . (4 T))  
 ((I A))  
 (R . (8 T))  
 ((J D))  
 ((J C))  
 ((I B F))  
 ((I A) . (4 T))  
 ((I F) . (8 T))  
 ((I A))  
 ((J D) . (4 T))

(PERT ((I F)) (1 . U) (2 . U) (3 . D))  
 ((I G))  
 ((I D))  
 (R . (B T))  
 ((I E))  
 (PERT ((I F)) (1 . U))  
 ((I G))  
 ((I A))  
 ((I G))  
 (PERT ((I F)) (1 . U))  
 ((I A))  
 ((I B))  
 ((I A))  
 ((I G))  
 ((I B))  
 ((J C))  
 ((I B))  
 ((I A))  
 ((I G))  
 ((I A))  
 ((J F))  
 (PERT ((J E)) (1 . D) (3 . D))  
 ((J D))  
 (PERT ((J E)) (1 . D) (3 . D))  
 ((J C))  
 ((I A))  
 ((J C))  
 (PERT ((I F S)) (2 . D) (3 . U))  
 ((I A))  
 (PERT ((I G S)) (1 . D) (2 . D))  
 (PERT ((I B)) (1 . D) (2 . D))  
 ((I A) . (4 T))  
 (R)  
 ((J E) . (B T 3))  
 ((J F))  
 ((J E))  
 ((J D))  
 ((J E))  
 ((J F))  
 ))  
 (SETG DS1 (QUOTE (
 (ARTIC SLUR)
 ((J G) . (4 T))
 (ARTIC LEGATO)
 ((J G) . (16 T))
 ((J F))
 ((J E))
 ((J D))
 ((J C) . (B T 3))
 ((J D))
 ((J C))
 ((I B))
 ((J C))
 ((J D))
 ((J E) . (B T))
 ((J D))
 (PERT ((J C)) (1 . U))
 ((I B))
 ((I A) . (4 D))
 ((I B) . (B T))



```

((J C))
((I B F))
((I A))
((I G))
((I F))
((I G))
((I A))
((I F))
((I B F))
((I G))
((I F))
((I E))
((I D) . (4 T))
(R)
(R)
((I A))
((I D) . (8 T))
((I F))
((I B F) . (4 T))
((I E) . (8 T))
((I G))
((J C) . (4 T))
(R . (8 T))
((I A))
((I G))
((I F S))
((I G) . (4 T))
((I A) . (2 T))
((I B) . (4 T))
((I C S) . (8 T))
((I D))
((I E))
((I F))
((I G) . (8 T 3))
((I A))
((I G))
((I F))
((I G))
((I A))
((I B F) . (4 T))
((I G))
((J D) . (8 T 3))
((J E))
((J D))
((J C S))
((J D))
((J E))
)))
(SET3 CS2 (QUOTE (
((L F) . (4 T))
((L D))
((K A) . (8 T 3))
((K B))
((K A))
((K G S))
((K A))
((K B))
((L C S))
((L D))
((L C S))

```

```

((K B))
((L C S))
((L D))
((L E) . (8 T))
((K A))
(ARTIC SLUR)
((L A) . (4 T))
(ARTIC LEGATO)
((L A) . (8 T))
((L G))
((L F))
((L E))
(ARTIC SLUR)
((L F) . (2 T))
))
(SETQ CS3 (QUOTE (
((J G) . (8 T 3))
((J A))
((J G))
((J F))
((J G))
((J A))
((J B F) . (4 T))
((J G))
))
(SETQ CS4 (QUOTE (
((L F) . (8 T))
((L E))
((L D) . (4 T))
(R . (8 T))
((L B F))
((L A))
((L G S))
(ARTIC SLUR)
((L A) . (8 T))
((L F))
((L E))
(ARTIC LEGATO)
((L D) . (8 T))
(ARTIC SLUR)
((L F) . (8 T))
((L E))
((L D))
(ARTIC LEGATO)
((L C S) . (8 T))
((L D) . (1 T))
(FINE)
))
(SETQ CS5 (QUOTE (
((J D) . (8 T))
((I F))
((I G))
((I A))
((I B F))
((I G))
((I F))
((I E))
((I F) . (4 T))
((I B F))
((I G))

```

```

((I A))
((I D) . (1 T))
(FIVE)
))
~ (SETQ C56 (QUOTE (
((J F) . (8 T 3))
((J 3))
((J F))
((J E))
((J F))
((J 3))
((J A) . (8 T))
((J D))
(ARTIC SLUR)
((K D) . (4 T))
(ARTIC LEGATO)
((K D) . (8 T))
((K C))
((J B F))
((J A))
(ARTIC SLUR)
((J B F) . (2 T))
))
(SETQ IBASE 8.)
(SETQ T2A (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (CADR X))
(T X))))
T2))
~ (SETQ T2B (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (COND
((NULL (ASSOC 1 X)) (CADR X))
((EQ (CDR (ASSOC 1 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (CAADR X))) (CDADR
X)))
(T (CONS (ENNOTE (SEMIDOWN (CAADR X))) (CDADR X))))))
(T X))))
T2))
(SETQ T2B (TR T2B 23))
(SETQ T2C (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (COND
((NULL (ASSOC 2 X)) (CADR X))
((EQ (CDR (ASSOC 2 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (CAADR X))) (CDADR
X)))
(T (CONS (ENNOTE (SEMIDOWN (CAADR X))) (CDADR X))))))
(T X))))
T2))
(SETQ T2C (TR T2C 30))
(SETQ T2D (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (COND
((NULL (ASSOC 3 X)) (CADR X))
((EQ (CDR (ASSOC 3 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (CAADR X)))
(CDADR X)))
(T (CONS (ENNOTE (SEMIDOWN (CAADR X))) (CDADR X))))))
(T X))))
T2))
~ (SETQ T3A (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((OR (EQ (CAR X) (QUOTE PERT)) (EQ (CAR X) (QUOTE COM))) (CADR X))
(T X))))
T3))
(SETQ T3B (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((OR (EQ (CAR X) (QUOTE PERT)) (EQ (CAR X) (QUOTE COM))) (COND

```

```

(T (SEMI DOWN (CAADR X)))) (CDADR X))))
(T X)))
T4))
(SETQ CS1A (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (CADR X))
(T X))))
CS1))
(SETQ CS1B (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (CONS (ENNOTE (SEMI UP (CAADR X)))
(CDADR X)))
(T X))))
CS1))
(SETQ CS1B (TR CS1B 30))
(PROG NIL
  (SETQ REP NIL)
  (SETQ LOC (LOAD VOICE1 (QUOTE (
(WAVE SQUARE)
(TEMPO 2)
))))
  A
  (SETQ LOC (LOAD LOC T1))
  (SETQ LOC (LOAD LOC T2A))
  (SETQ LOC (LOAD LOC T3A))
  (SETQ LOC (LOAD LOC T4A))
  (SETQ LOC (LOAD LOC CS1A))
  (SETQ LOC (LOAD LOC CS6))
  (SETQ LOC (LOAD LOC (TR T3A 5)))
  (SETQ LOC (LOAD LOC T1))
  (SETQ LOC (LOAD LOC T2D))
  (SETQ LOC (LOAD LOC T3D))
  (SETQ LOC (LOAD LOC T4D))
  (SETQ LOC (LOAD LOC CS3))
  (SETQ LOC (LOAD LOC T1))
  (COND (REP (GO B)))
  (SETQ REP T)
  (GO A)
  B
  (SETQ LOC (LOAD LOC CS5))
  (RETURN LOC))
(PROG NIL
  (SETQ REP NIL)
  (SETQ LOC (LOAD VOICE2 (QUOTE (
(TEMPO 20)
(R . (1 T))
(TEMPO 2)
))))
  A
  (SETQ LOC (LOAD LOC (TR T1 23)))
  (SETQ LOC (LOAD LOC T2B))
  (SETQ LOC (LOAD LOC T3B))
  (SETQ LOC (LOAD LOC T4B))
  (SETQ LOC (LOAD LOC (TR T1 30)))
  (SETQ LOC (LOAD LOC T2C))
  (SETQ LOC (LOAD LOC T3C))
  (SETQ LOC (LOAD LOC T4C))
  (SETQ LOC (LOAD LOC CS1B))
  (SETQ LOC (LOAD LOC CS2))
  (SETQ LOC (LOAD LOC T3C))
  (COND (REP (GO B)))
  (SETQ REP T)
  (GO A)
  B
  (SETQ LOC (LOAD LOC CS4))
  (RETURN LOC))

```

```

((NULL (ASSOC 1 X)) (CADR X))
(T (CONS (ENNOTE (SEMIDOWN (CAADR X))) (CDADR X))))
(T X)))
T3))
(SETQ T3B (TR T3B 23))
(SETQ T3C (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (COND
(NULL (ASSOC 2 X)) (CADR X))
(EQ (CDR (ASSOC 2 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (CAADR X)))
(CDADR X)))
(T (CONS (ENNOTE (SEMIDOWN (CAADR X))) (CDADR X))))))
(EQ (CAR X) (QUOTE COM)) (CONS (CAADR X) (CDR (ASSOC 2 X))))
(T X))))
T3))
(SETQ T3C (TR T3C 30))
(SETQ T3D (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((OR (EQ (CAR X) (QUOTE PERT)) (EQ (CAR X) (QUOTE COM))) (COND
(NULL (ASSOC 3 X)) (CADR X))
(EQ (CDR (ASSOC 3 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (COND
(NULL (ASSOC 2 X)) (CAADR X))
(EQ (CDR (ASSOC 2 X)) (QUOTE U)) (SEMIJP (CAADR X)))
(T (SEMIDOWN (CAADR X)))))) (CDADR X)))
(T (CONS (ENNOTE (SEMIDOWN (COND
(NULL (ASSOC 2 X)) (CAADR X))
(EQ (CDR (ASSOC 2 X)) (QUOTE U)) (SEMIJP (CAADR X)))
(T (SEMIDOWN (CAADR X)))))) (CDADR X))))
(T X))))
T3))
(SETQ T4A (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (CADR X))
(T X))))
T4))
(SETQ T4B (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (COND
(NULL (ASSOC 1 X)) (CADR X))
(EQ (CDR (ASSOC 1 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (CAADR X)))
(CDADR X)))
(T (CONS (ENNOTE (SEMIDOWN (CAADR X))) (CDADR X))))))
(T X))))
T4))
(SETQ T4B (TR T4B 23))
(SETQ T4C (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (COND
(NULL (ASSOC 2 X)) (CADR X))
(EQ (CDR (ASSOC 2 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (CAADR X)))
(CDADR X)))
(T (CONS (ENNOTE (SEMIDOWN (CAADR X))) (CDADR X))))))
(T X))))
T4))
(SETQ T4C (TR T4C 30))
(SETQ T4D (MAPCAR (FUNCTION (LAMBDA (X)
(COND ((EQ (CAR X) (QUOTE PERT)) (COND
(NULL (ASSOC 3 X)) (CADR X))
(EQ (CDR (ASSOC 3 X)) (QUOTE U)) (CONS (ENNOTE (SEMIJP (COND
(NULL (ASSOC 2 X)) (CAADR X))
(EQ (CDR (ASSOC 2 X)) (QUOTE U)) (SEMIJP (CAADR X)))
(T (SEMIDOWN (CAADR X)))))) (CDADR X)))
(T (CONS (ENNOTE (SEMIDOWN (COND
(NULL (ASSOC 2 X)) (CAADR X))
(EQ (CDR (ASSOC 2 X)) (QUOTE U)) (SEMIJP (CAADR X)))

```

## CANON alla Duodecima in Contrapunto alla Quinta

The image displays a musical score for a canon in G major, 12-measure intervals, in counterpoint with the fifth. The score is written for piano and consists of five systems, each with a grand staff (treble and bass clefs). The key signature has one sharp (F#), and the time signature is common time (C). The score is marked with measure numbers 5, 10, 15, 20, and 25 in boxes. The music features a rhythmic pattern of eighth and sixteenth notes, with frequent sixteenth-note runs. The bass line is more active than the treble line, often providing a harmonic foundation with sustained notes and moving lines. The piece is characterized by its intricate counterpoint and the consistent 12-measure interval between the two parts.

[30]

Musical score for measures 30-35. The piece is in 2/4 time with a key signature of one sharp (F#). The right hand features a melodic line with eighth and sixteenth notes, while the left hand provides a steady accompaniment of eighth notes. Measure 35 ends with a repeat sign.

[33] [35]

Musical score for measures 33-38. The right hand continues the melodic development with some sixteenth-note passages. The left hand maintains the eighth-note accompaniment. Measure 38 ends with a repeat sign.

[40]

Musical score for measures 40-45. The right hand has a more active melodic line with slurs. The left hand accompaniment remains consistent. Measure 45 ends with a repeat sign.

[45]

Musical score for measures 45-50. The right hand features a series of slurred eighth notes. The left hand accompaniment continues. Measure 50 ends with a repeat sign.

[50]

Musical score for measures 50-55. The right hand has a melodic line with some sixteenth-note runs. The left hand accompaniment continues. Measure 55 ends with a repeat sign.



Musical score system 1, measures 54-59. The system consists of two staves (treble and bass clef) with a brace on the left. The music is in 2/4 time. Measure 54 is marked with a box containing the number 54. The piece concludes with a double bar line at the end of measure 59.



Musical score system 2, measures 60-65. The system consists of two staves (treble and bass clef) with a brace on the left. The music is in 2/4 time. Measure 60 is marked with a box containing the number 60. The piece concludes with a double bar line at the end of measure 65.



Musical score system 3, measures 66-71. The system consists of two staves (treble and bass clef) with a brace on the left. The music is in 2/4 time. Measure 66 is marked with a box containing the number 66. The piece concludes with a double bar line at the end of measure 71.



Musical score system 4, measures 72-77. The system consists of two staves (treble and bass clef) with a brace on the left. The music is in 2/4 time. Measure 72 is marked with a box containing the number 72. The piece concludes with a double bar line at the end of measure 77.



Musical score system 5, measures 78-83. The system consists of two staves (treble and bass clef) with a brace on the left. The music is in 2/4 time. Measure 78 is marked with a box containing the number 78. The word "Finale" is written below the staff in the middle of the system. The piece concludes with a double bar line at the end of measure 83.