Unix Style Computer Aided Composition

Appendix A: Function Listings

Ray Garner

20156967

psyrg4@nottingham.ac.uk

Computer Science with Year in Industry Bsc

1. Common Data

Data	Encoding
Pitch	Int
Interval	Int
Degree	Int
Scale	○[Interval]
Mode	(Scale, Degree)
Root	Pitch
Key	(Root, Mode)
Chord	[Pitch]
Line	[Pitch]
Harmony	[Line]
Alteration	Int

2. Common Functions

2.1. Input/Output

Function	Description
read_accidental(a)	Return encoded alteration a
read_note(p)	Return encoded natural pitch p
read_tone(p, a)	Return encoded pitch p with alteration a
read_mode(m)	Return encoded mode m
init_key_field(k, i)	Initialise all cells of M×N matrix k with value i where M is number of pitches and N is number of major scale modes
read_key_list(k, x)	For each key(root, mode) read from STDIN set k[root][mode] to x
print_matching_keys(k, x)	For each k[root][mode] equal to x print key(root, mode)
is_accidental(p)	Returns true if the decoding of p requires a sharp or flat else returns false
is_correct_accidental(k, a)	Returns true if the decoding of key k can be represented using accidental a
get_correct_accidental(k)	Returns an accidental which the decoding of key k can be written using
print_note(a, p)	Print decoding of pitch p using accidental a

2.2. Internal

Function	Description

clock_mod(x, m)	Returns a member of {0..m} congruent to x where x may be positive or negative
step(d, k)	Returns the pitch one step up from degree in key
calc_degree(p, k)	Returns the degree of pitch p in the context of key k
is_diatonic(p, k)	Returns true if pitch p is in key k, false otherwise
apply_steps(d, k, s)	Returns the pitch s steps from degree d in key k where s may be positive or negative
min_tone_diff(p, q)	Returns the minimum pitch difference between pitches p and q in semitones

3. Mode Generating

Function	Description
check_relative_modes(r, k)	For all k[root][mode] in matrix k which are relative to key r, increment the cell value
process_notes(n, k)	For each pitch in list n call check_relative_modes(key(note,m), k) for each mode m

4. Interval Filtering

Function	Description
degree_val(d, m)	Return the interval between the first and degree d in mode m of the major scale in semitones
correct_alteration(d, m, a)	Returns true if the interval between the first and degree d in mode m is different to the corresponding interval in the major scale

5. Melody Generation

Function	Description
count_scale_steps(k, start, end)	Return the steps it takes to reach pitch end from pitch start in key k
generate_line(len, tones, k)	Returns a melody line of length len using pitches from list tones as a skeleton and filled out with pitches from key k

6. Melody Harmonisation

Function	Description
is_primary_degree(p, k)	Return true if pitch p is degree 1, 4 or 5 in key k else returns false
add_middle_note(b, m, k)	Return the pitch x such that the chord made up of pitch b in the bass, pitch x in the middle and pitch m in the melody forms the most complete chord possible in key k
generate_middle_line(b, m, k)	Return a line between the bass line b and melody line m that would such that they would be harmonious together in the key k
pick_primary_chord(d)	Return a primary chord degree which melody degree d is a part of
faulty_note(b, m, k)	Return the number of faults incurred by having bass pitch b with melody pitch m in key k
count_faults(b, m, k)	Return the number of faults incurred by having the bass line b with melody line m in key k
alt_chord_choice(c, d)	Return another primary chord degree other than c which degree d is a part of if possible, otherwise return degree c
improve_bass_line(b, m, k)	Returns an improved version of a simple bass line b using melody line m and key k as context
generate_bass_line(m, k)	Returns a simple bass line to work with melody line m in key k

7. Conversion to MusicXML

Function	Description
write_headers()	Print MusicXML headers
write_part_def(i, n)	Print the definition for a part with name n and ID i
write_part_line(i, l, o, c)	Print the MusicXML representation of line l with ID i in octave o using clef c

7.1. Stave Key Signature Display

Function	Description
spacing(a, l)	Returns the indent as a number of spaces required for correct placement of accidental a on stave line l
print_key_sig(a, l)	Prints the key signature on a stave to the terminal where p is a list of flags defining which lines should be altered and a is the alteration which should be applied if so
note_status(a, n)	Returns a list of flags representing which lines should be altered using accidental a to represent the key signature with n instances of accidental a
is_flat_key(k)	Returns true if key k must be represented using flats rather than sharps, otherwise returns false
calc_accidentals(a, k)	Returns the number of accidentals of type a which must be used to represent key k
relative_ionian(k)	Return the root pitch of the relative Ionian for key k
note_to_cf(p)	Returns the number of sequential perfect fifth steps pitch p is from the pitch C

7.2. Fretboard Mode Display

Function	Description
write_string(k)	Return a single guitar string representation of key k
note_to_fret(p)	Return the guitar fret which pitch p lies on a guitar E string