cisdi Journal Computing, Information Systems, Development Informatics & Allied Research Journal
Vol. 7 No. 1 March, 2016 - www.cisdijournal.net

Algorithmic Modeling and Software Framework of a Twelve Tone Serialism

Garba, E. J. & Ahmadu, A. S.
Department of Computer Science
Federal University of Technology
Modibbo Adama University of Technology
P.M.B. 2076 Yola, Adamawa State, Nigeria.
E-mail: 1e.j.garba@mautech.edu.ng, 2asabeahmadu@mautech.edu.ng
Telephone: 10803 694 38 81, 208038171818

ABSTRACT

Music is the skillful arrangement and alternation of sound and silence within a given period of time with appreciable aural perception [1]. As musicians play instruments, the sound-producing parts vibrate and oscillate to give out tones perceived as music. For musicians to play a piece they creatively and logically manage several concepts and elements, such as rhythm, melody, dynamics, harmony (chord progression), tone color, texture, form etc. It is imperative to note that those aspects of music that are strongly logical could be modeled mathematically and computationally. It is on the basis of this premise that this paper shall study the underlining mathematical concepts and relations involved in the composition of the Twelve-Tone Serialism. From the mathematical models, algorithms shall be development and presented as flowcharts. Finally, a software framework shall be developed and presented as JAVA codes. This framework is expected to be useful to software designers and application developers in the field of Music and Multimedia Technology.

Keywords: Algorithmic Modeling; Software Framework; Twelve Tone Serialism; Pitch Class And Algorithmic Composition.

1. INTRODUCTION

Mathematics and music are closely related logically. People with musical abilities think both logically and artistically. In fact, musicians tend to make better mathematicians. Basic mathematics is inherent in music through the creation processes of the rhythms and melodies. Music is considered logical and as well as a means of expressing emotions. The creative process of music is both extremely logical and mathematical [2]. According to Robert Morris [3], in modern music theory, twelve notes make an octave. Twelve-tone composition arranges these twelve notes without repetition until each note has been played once. The application of mathematics to twelve-tone composition depends strongly on the permutation of the twelve tones – which is represented by an array of twelve notes. Each note is represented by a number ranging from 0 through 11. The chromatic scale is represented by an array: [0 1 2 3 4 5 6 7 8 9 10 11] – where 0 in the array represents note C; 1 represents note C#, etc.; while 11, the last note of the octave, represents note B. See Table 1. Arnold Schoenberg (1874 -1951) and other musicians of the early twentieth century agreed that music composition, in analogy to science, was not an aesthetic project but rather a kind of empirical and heuristic problem solving [8].

2. ALGORITHMIC COMPOSITION

According to Fernández and Vico [18], algorithmic composition is the partial or total automation of the process of music composition by using computers programs. Some of the algorithmic composition models and techniques closely related to Artificial Intelligence used for algorithmic composition include grammar-based, probabilistic, neural networks, symbolic rule-based systems, constraint programming and genetic/evolutionary algorithms. Manual and traditional approach to music composition requires the musician to be engaged in series of activities, such as the definition of melody and rhythm, harmony, chord progression, counterpoint formulation, arrangement and notation. Those aspects of these activities that are mathematical and logical could be automated by the computer to varying degrees of speed, accuracy and even ([19] and [20]). To achieve algorithmic composition, the focus would be on algorithms development, programming languages, software frameworks and graphical tools. Algorithmic composition is the technique of using algorithms to create music. Music composition is perceived a complex and challenging activity for those not having musical knowledge or skill [21]. According to [6], [22] and [23] some common techniques of algorithmic composition include state machines, rule-based, knowledge-based, grammars, stochastic processes, and genetic algorithms.

3. HISTORY OF TWELVE TONE SERIALISM

Arnold Schoenberg (1874 – 1951) developed this unifying method of composition. He created this technique which he called "the method of composing with twelve tones". It provided rules of composition that resulted in the emergence of the new style of music. Though the new technique was very structured, it still allowed for some artistic freedom in developing musical pieces by the composer ([4] and [5]). The tone-row technique introduced by Arnold Schönberg was further developed into serialism by Anton Webern and his successors [6] and [24].

4. AXIOMS OF THE TWELVE TONE TECHNIQUE

Arnold Schoenberg suggested some rules to adhere to when composing based on this technique [7]:

1. The composition must be based on an arrangement or series of the twelve pitches of the chromatic scale as determined by the composer. This series of tones is then called the tone row or set. This tone row may begin on any of the twelve pitches.
2. The repetition of a note is allowed only when all other pitches have been sounded. However, a pitch may be repeated immediately afterwards.
3. The tone row could be played in its original form, retrograde, inversion, or retrograde inversion.

Torsten Anders [16] opines that this technique allows the composer to organize the pitches to form a tone row (also referred to as twelve-tone series). A tone row is a sequence of 12 tones that permutates the twelve chromatic pitch classes in a particular order. A twelve tone row is usually transformed in many ways: transposition, retrogression (reversal in time) and inversion (mirroring a pitch axis). Composers use tone rows as music composition devices to achieve unity and coherency in a musical piece – regardless of the harmonic dissonance. More information on the practical introduction to this composition technique was written by George Perle [17].

5. PITCH CLASS

Pitch Classes are used for describing harmonies in music. They can describe and manipulate harmony created within a twelve-tone scale. Pitch classes describe pitches independent of octave displacement and enharmonic spelling. There are twelve pitch classes where each octave has 12 chromatic notes. They are numbered from 0 through 11: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11. They are as well referred to as Pitch Class Representatives since the numbers stand in the place of the letters denoting the chromatic pitches [10].

On the other hand, according to ([11], [12], [13], [14] and [15]), Pitch Class Set is made up of a list of pitch class numbers; for instance: [0, 4, 7, 10]. They are also referred to as PC Sets. For example, the PC Set for a C minor chord is [0, 3, 7]. The "modulo" operations is used when manipulating pitch classes. The operator takes the remainder of an integer divided by some other integer. For example, 15 modulo 12 = 3 (that is, 12 goes into 15 once, with 3 as a remainder). Note however, that PC sets use modulo 12 – that is, any number above 12 is reduced using modulo 12 with reference to a number range from 0 to 11.

Table 1: Pitch Class

PC	Tonal counterparts
0	**C** (also B sharp, D double-flat)
1	**C sharp, D flat** (also B double-sharp)
2	**D** (also C double-sharp, E double-flat)
3	**D sharp, E flat** (also F double-flat)
4	**E** (also D double-sharp, F flat)
5	**F** (also E sharp, G double-flat)
6	**F sharp, G flat** (also E double-sharp)
7	**G** (also F double-sharp, A double-flat)
8	**G sharp, A flat**
9	**A** (also G double-sharp, B double-flat)
10	**A sharp, B flat** (also C double-flat)
11	**B** (also A double-sharp, C flat)

6. CREATING A MATRIX FROM THE TWELVE TONE ROW

When creating a twelve tone row, notes of the chromatic scale are represented by their notes or by Pitch Class values (the numbers from zero through eleven). A matrix is then created to display all of the possible permutations/arrangements of the row. This twelve-by-twelve matrix is read from four directions: top to bottom, bottom to top, left to right, or right to left [2].The twelve tones consist in a row of twelve pitches, which constitutes the base to the composition. This results in unity and coherence within the musical piece [9].

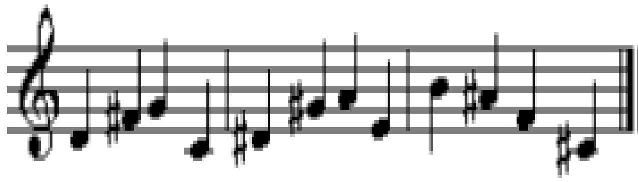

Figure 1: Original Twelve-Tone Notation.

The process of creating a twelve-by-twelve matrix from a given twelve-tone row involves a number of steps:
Step 1: Extraction of musical notes from the twelve-tone notation produces the Original twelve-tone Row (OR). From the original twelve-tone notation in Figure 1, we obtain an OR thus: D, F#, G, C, D#, G#, A, E, B, A#, F, C#.

Table 2: Pitch Class of the Original twelve-tone Row

OR	D, F#, G, C, D#, G#, A, E, B, A#, F, C#
PCOR	2, 6,7, 0, 3, 8, 9, 4, 11, 10, 5, 1

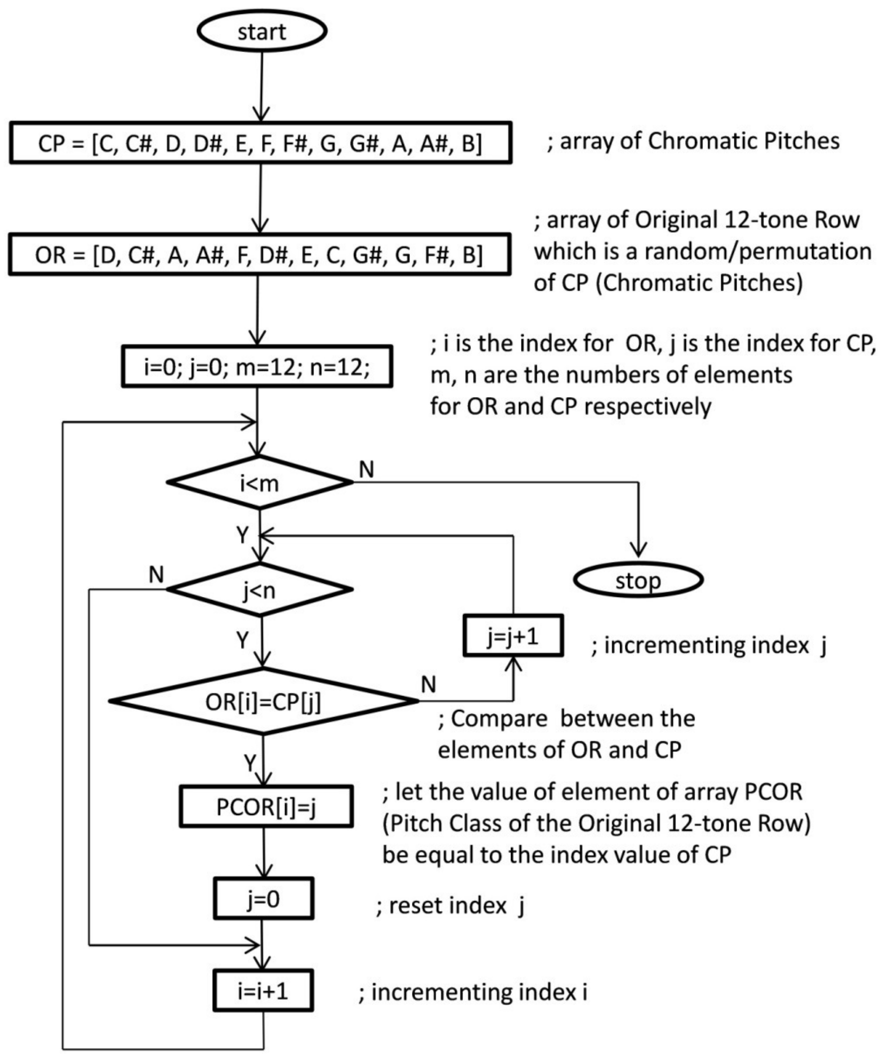

Figure 2: Algorithm for generating the Pitch Class of the Original 12-tone Row (PCOR).

Step 2: Creation of the Pitch Class for the notes of the Original twelve-tone Row (PCOR) from the Original twelve-tone Row (OR) with the aid of the Pitch Class Table (see Table 1). Based on the OR above, the PCOR is thus formulated in Table 2. The algorithm for creating a PCOR from an OR is shown in Figure 2; while the Java code listing for the algorithm is shown in Listing 1.

Step 3: Creation of the first row of the twelve-tone Matrix from the PCOR. That is, the PCOR shall occupy the first row of the twelve-by-twelve Matrix; where m and n indicate the number of rows and columns respectively. Note also that i and j indicate the indexes of the elements for the rows and columns respectively. Therefore, at this step, the Matrix shall have just the first row. See Figure 3.

Listing 1: Java code for generating the Pitch Class of the Original 12-tone Row (PCOR).

```
for(byte j=0; j<getPichClassOriginalRow.length; j++){ matrix[0][j] = getPichClassOriginalRow[j]; }//end for j...
```

Listing 2: Java code for creating the first row of the 12-tone Matrix from PCOR.

```
public Object[] getPichClassOriginalRow(Object[] chromaticPitches, Object[] originalRow){
Object[] pitchClass = new Object[originalRow.length];
for(byte i=0; i<originalRow.length; i++){
for(byte j=0; j<chromaticPitches.length; j++){
if(originalRow[i]==chromaticPitches[j]){
pitchClass [i]=j;
break;
}//end if...
}//end for j...
}//end for i...
return pitchClass;
}//end…
```

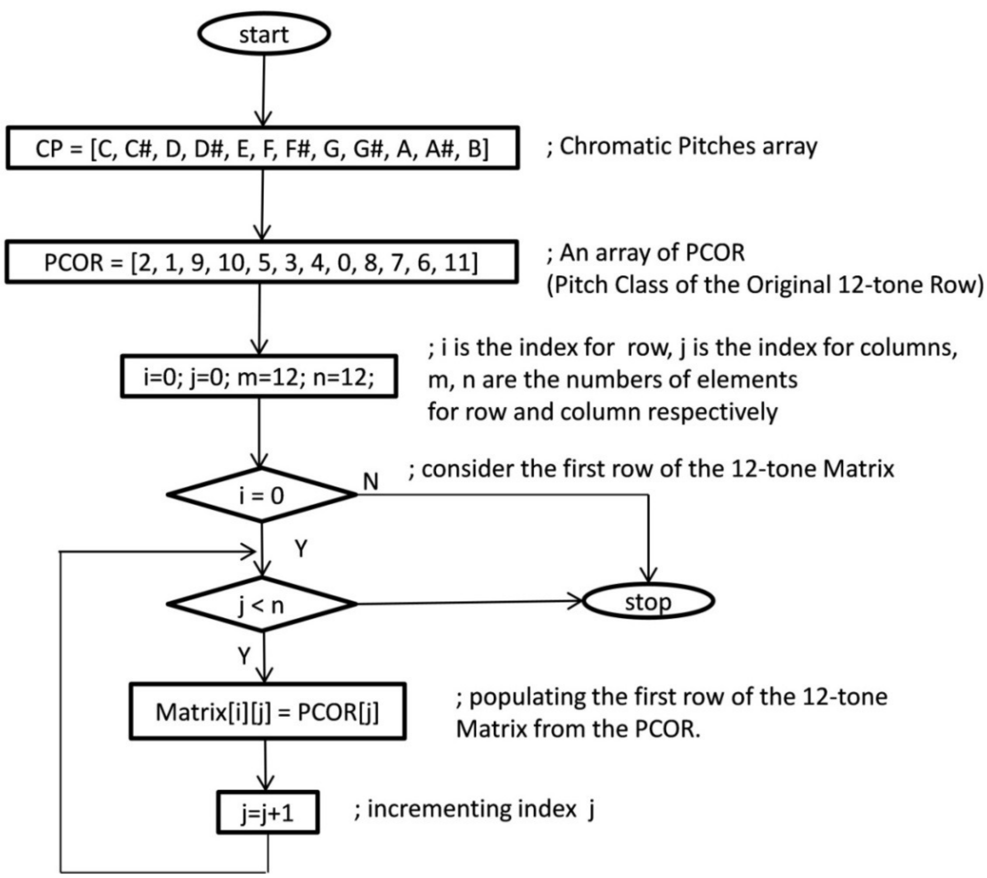

Figure 4: Algorithm for creating the first row of the 12-tone Matrix from PCOR.

The algorithm for creating first row of the Matrix from PCOR is shown in Figure 4; while the Java code listing for the algorithm is shown in Listing 2. Step 4: Creation of the first column of the Matrix from the "inverted" PCOR. The elements of the first column are obtained from transpositional subtraction of PC values of the first row from 12 (a modulo 12 operation is performed on the result of the subtraction). By the end of this step, the Matrix shall look thus (see Figure 5). The algorithm for creating first column of the Matrix from transpositional subtraction of PC values of the first row from 12 is shown in Figure 6; while the Java code listing for the algorithm is shown in Listing 3. Step 5: Creation of the remaining rows and columns of the Matrix – which are often referred to as Prime Rows (PR). This is obtained by the transpositional summation of the PC values of the first row and first column (a modulo 12 operation is performed on the result of the summation). The result of this step is shown in Figure 7.

2	6	7	0	3	8	9	4	11	10	5	1
-	-	-	-	-	-	-	-	-	-	-	-
…	…	…	…	…	…	…	…	…	…	…	…
-	-	-	-	-	-	-	-	-	-	-	-

Figure 3: Creating first row of Matrix from PCOR.

2	6	7	0	3	8	9	4	11	10	5	1
6	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-
0	-	-	-	-	-	-	-	-	-	-	-
9	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-
8	-	-	-	-	-	-	-	-	-	-	-
1	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-
7	-	-	-	-	-	-	-	-	-	-	-
11	-	-	-	-	-	-	-	-	-	-	-

Figure 5: Creating first column of the Matrix from transpositional subtraction of PC values of the first row from 12.

Listing 3: Java code for creating the first column of the 12-tone Matrix from an inverted Pitch Class of the PCOR.

```
for(byte i=1; i<getPichClassOriginalRow.length; i++){
        int pitchClassValue = 12 -
Integer.parseInt(matrix[0][i].toString());
        if (pitchClassValue >= 12){
                matrix[i][0] = pitchClassValue % 12;
        }else{matrix[i][0] = pitchClassValue;}
}//end for i…
```

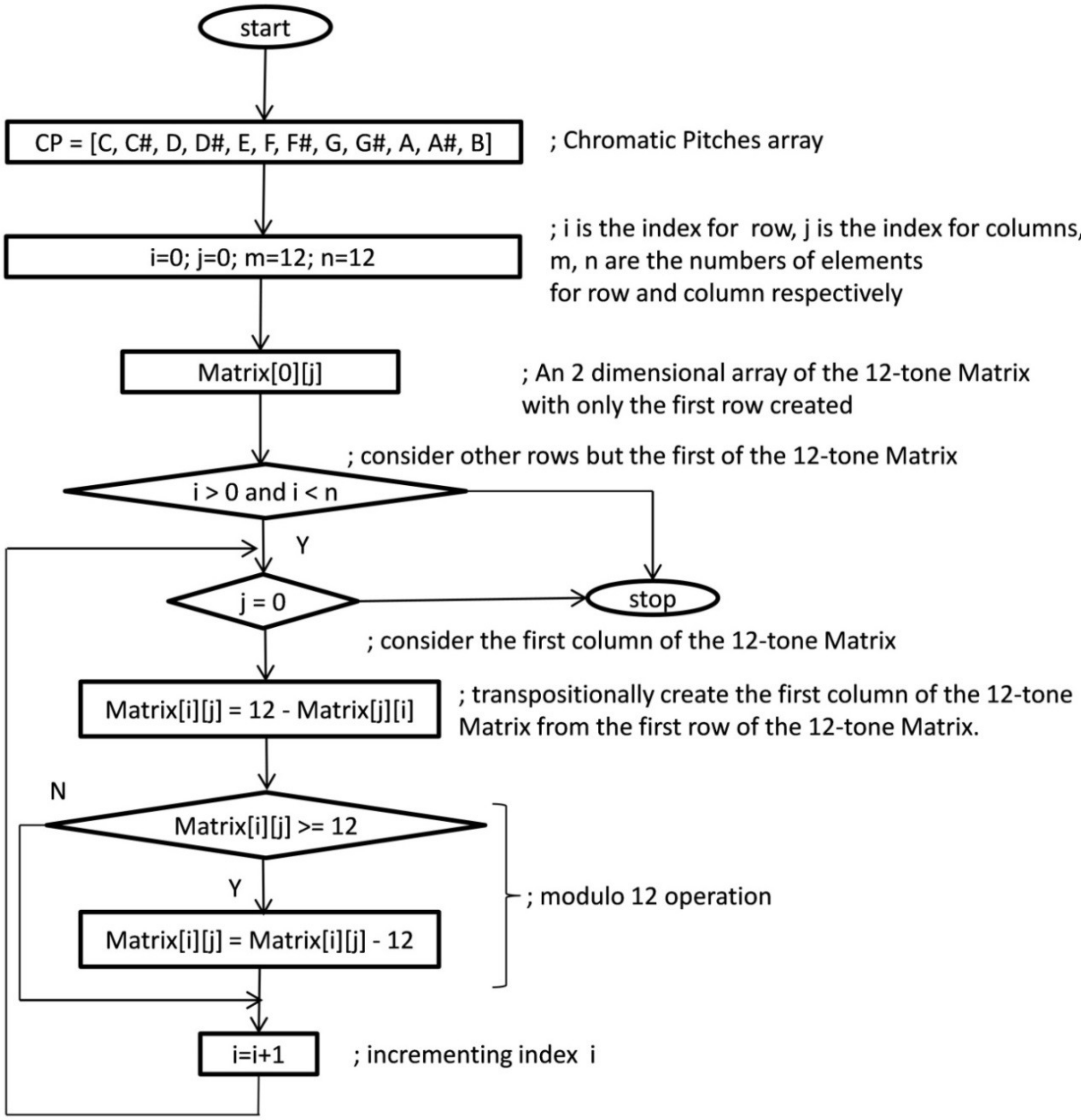

Figure 6: Algorithm for creating the first column of the 12-tone Matrix from an inverted Pitch Class of the PCOR.

2	6	7	0	3	8	9	4	11	10	5	1
6	0	1	6	9	2	3	10	5	4	11	7
5	11	0	5	8	1	2	9	4	3	10	6
0	6	7	0	3	8	9	4	11	10	5	1
9	3	4	9	0	5	6	1	8	7	2	10
4	10	11	4	7	0	1	8	3	2	9	5
3	9	10	3	6	11	0	7	2	1	8	4
8	2	3	8	11	4	5	0	7	6	1	9
1	7	8	1	4	9	10	5	0	11	6	2
2	8	9	2	5	10	11	6	1	0	7	3
7	1	2	7	10	3	4	11	6	5	0	8
11	5	6	11	2	7	8	3	10	9	4	0

Figure 7: Creating Prime Rows by the transpositional summation of the PC values of the firstrow and first column.

D	F#	G	C	D#	G#	A	E	B	A#	F	C#
F#	C	C#	F#	A	D	D#	A#	F	E	B	G
F	B	C	F	G#	C#	D	A	E	D#	A#	F#
C	F#	G	C	D#	G#	A	E	B	A#	F	C#
A	D#	E	A	C	F	F#	C#	G#	G	D	A#
E	A#	B	E	G	C	C#	G#	D#	D	A	F
D#	A	A#	D#	F#	B	C	G	D	C#	G#	E
G#	D	D#	G#	B	E	F	C	G	F#	C#	A
C#	G	G#	C#	E	A	A#	F	C	B	F#	D
D	G#	A	D	F	A#	B	F#	C#	C	G	D#
G	C#	D	G	A#	D#	E	B	F#	F	C	G#
B	F	F#	B	D	G	G#	D#	A#	A	E	C

Figure 8: The complete twelve-tone Matrix

Listing 4: Java code for generating the Prime Row (PR) of the 12-tone Matrix from the first row and column of the 12-tone Matrix.

```
for(byte i=1; i<getPichClassOriginalRow.length; i++){
for(byte j=1; j<getPichClassOriginalRow.length; j++){
int pitchClassValue = Integer.parseInt(matrix[i][0].toString())
+Integer.parseInt(matrix[0][j].toString());
if (pitchClassValue >= 12){
                        matrix[i][j] = pitchClassValue % 12;
}else{matrix[i][j] = pitchClassValue;}// else...
}//end for j...
}//end for i...
```

The algorithm for creating Prime Rows by the transpositional summation of the PC values of the first row and first column is shown in Figure 9; while the Java code listing for the algorithm is shown in Listing 4.

Step 6: The interpretation of the Matrix (in Figure 8) is thus:

a. Prime Row: reading from left to right.
b. Retrograde Rows: reading from right to left.
c. Inversion Rows: reading from top to bottom.
d. Retrograde-Inversion Rows: reading from bottom to top.

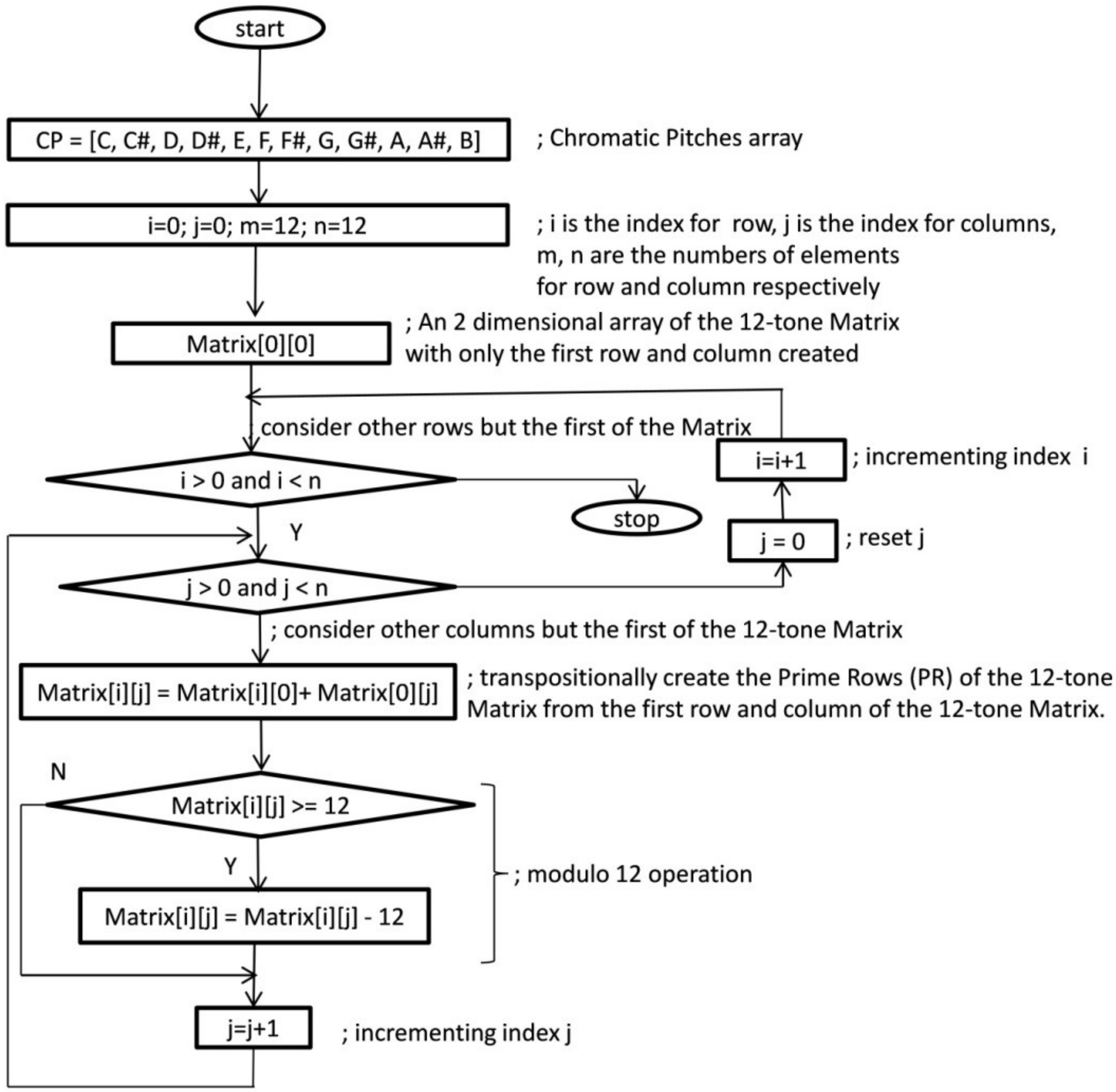

Figure 9: Algorithm for generating the Prime Row (PR) of the 12-tone Matrix from first row and column.

7. CONCLUSION

This Objected Oriented software framework (in form of Java objects) provides the basic building blocks for developers who wish to integrate audio, sound, music and multimedia into their applications. This framework is expected to be useful to music and multimedia programmers who develop music application and tools that would facilitate music creation and interaction as it reflects the needs of the user and the musician in light of Twelve Tone Serialism. The direction for future research is multifaceted. However, worthy of note is that this algorithm is applicable not only to twelve-tone serialism but to any piece of musical piece of any length. Also, multiple order variations could be created from the generated matrix.

REFERENCES

[1] Garba, E. J. (2003). *Computer Music – Rhythm Programming, Processing and Mastering*. Bloomington, Canada: Trafford Publishing Canada.

[2] Moorhead, E. (2003). Twelve Tone Serialism. University of Puget Sound. United States: Washington 1500 N Warner St, Tacoma, WA 98416. Retrieved from http://mathcs.pugetsound.edu/~bryans/Current/Journal_Spring_2003/213_2003_EMoorheard.PDF. Accessed: 20/01/2015 17:51

[3] Morris, R. (2007). *Mathematics and the Twelve-Tone System: Past, Present, and Future*. Proceedings of the 1st International Conference, Mathematics and Computation in Music (MCM) 2007 (pp. 266-288) Berlin, Germany, May 18–20, 2007.

[4] Forney, K. and Machlis, J. (1999). *The Enjoyment of Music*. New York, United States of America: W. W. Norton & Co., New York.

[5] Schoenberg, A. (1975). Composition with Twelve Tones. Style and Idea: Selected Writings. University of California Press, Berkeley, 1975. Retrieved from http://www.toddtarantino.com/hum/compositionwithtwelvetones.html. Accessed: 08/04/2016 15:36.

[6] Järveläinen, H. (2000). Algorithmic Musical Composition.Helsinki University of Technology, Telecommunicationssoftware and multimedia laboratory. Retrieved from www.tml.tkk.fi/Studies/Tik-111.080/2000/papers/hanna/alco.pdf. Accessed: 5/12/2007 14:47

[7] Kostka, S. and Payne, D. (2000). *Tonal Harmony*. San Francisco, United States of America: McGraw Hill, San Francisco.

[8] Ashby, A. (2001). Schoenberg, Boulez, and Twelve-Tone Composition as "Ideal Type". *Journal of the American Musicological Society*, 54(3), 585 -625. University of California Press. Retrieved from http://www.jstor.org/stable/831912 Accessed: 17/02/2016 03:57.

[9] Gado, A., B. (2005). *A study of twelve-tone technique in selected pieces: Hans Joachim Koellreutter and César Guerra-Peixe* (Dissertation for the Degree of Master in Musical Analysis presented to the Arts Institute from UNICAMP Universidade de Campinas). Retrieved from http://www.adrianogado.com.br/pesquisa/dissertation.pdf. Accessed: 08/04/2016 12:57.

[10] Nelson, P. (2007). Pitch Class Sets. Retrieved from http://composertools.com/Theory/PCSets/PCSets1.htm#_Toc72662199. Accessed: 08/04/2016 13:46

[11] Huron, D. (1994). Interval-Class Content in Equally Tempered Pitch-Class Sets: Common Scales Exhibit Optimum Tonal Consonance. *Music Perception: An Interdisciplinary Journal*, 11(3), 289-305. University of California Press. Retrieved from http://www.jstor.org/stable/40285624. Accessed: 08/04/2016 15:28

[12] Pendergrass, M. (2013). *Two Musical Orderings*. Department of Mathematics and Computer Science, Hampden-Sydney College, Hampden-Sydney, Virginia, May 24, 2013. Retrieved from http://www.hsc.edu/Documents/academics/MathCS/Pendergrass/twoMusicalOrders_rev1_arxiv.pdf. Accessed: 08/04/2016 16:45.

[13] Lewin, D. (2001). Special Cases of the Interval Function Between Pitch-Class Sets X and Y. *Journal of Music Theory*, 45(1), 1–30.

[14] Mead, A. (1988). Some Implications of the Pitch-Class/Order-Number Isomorphism Inherent in the Twelve-Tone System: Part One. *Perspectives of New Music*, 26(2), 96–163.

[15] Mead, A. (1989). Some Implications of the Pitch-Class/Order-Number Isomorphism Inherent in the Twelve-Tone System: Part Two. *Perspectives of New Music*, 27(1), 180–233.

[16] Anders, T. and Miranda, E. R. (2011). Constraint Programming Systems for Modelling Music Theories and Composition. *ACM Computing Surveys (CSUR) Surveys Journal*, 43(4), October 2011 Article No. 30 ACM New York, NY, USA ISSN: 0360-0300 EISSN: 1557-7341 doi: 10.1145/1978802.1978809

[17] Perle, G. (1991). *Serial Composition and Atonality: An Introduction to the Music of Schoenberg, Berg, and Webern, Revised, 6th ed*. California, United States of America: University of California Press, California.

[18] Fernández, J. D. and Vico, F. (2013). AI Methods in Algorithmic Composition: A Comprehensive Survey. *Journal of Artificial Intelligence Research*, 48, 513-582.

[19] Loy, G., & Abbott, C. (1985). Programming languages for computer music synthesis, Performance and composition. *ACM Computing Surveys*, 17 (2), 235–265.

[20] Pope, S. T. (1993). Music Processing. *Music composition and editing by computer* (pp. 25–72). Oxford, United Kingdom: Oxford University Press, Oxford.

[21] Unehara, M. and T. Onisawa. (2009). Construction of Music Composition System with Interactive Genetic Algorithm. University of Tsukuba, Tennodai, Tsukuba, Ibaraki, 305-8573, JAPAN. Retrieved from http://www.idemployee.id.tue.nl/g.w.m.rauterberg/conferences/CD_doNotOpen/ADC/final_paper/549.pdf.

[22] Todd, P. M. and G. M. Werner. (1998). Frankensteinian Methods for Evolutionary Music Composition. In Griffith and Todd, P. M. (Eds.), *Musical networks: Parallel perception and performance* (pp. 313-339). Cambridge, MA: MIT Press/Bradford Books, Cambridge.

[23] Espi, D., P. J. Ponce de Leon, C. Perez-Sancho, D. Rizo, J. M. Inesta, F. Moreno-Seco, and A. Pertusa. (2009). A Cooperative Approach to Style-Oriented Music Composition. Departamento de Lenguajes y Sistemas Informaticos University of Alicante, Spain. Retrieved from http://193.145.231.49/repositori/grfia/pubs/186/wijcai07.pdf.

[24] Brown, S. C. (2015). Twelve-Tone Rows and Aggregate Melodies in the Music of Shostakovich. *Journal of Music Theory*, 59(2), 191-234. doi: 10.1215/00222909-3136000.

AUTHOR'S BIOGRAPHIES

Etemi Joshua Garba is a Music/Multimedia Consultant with Ethereal Multimedia Technology. He is also a Senior Lecturer (Assistant Professor) in the department of Computer Science at the Federal University of Technology (Modibbo Adama University of Technology), Yola, Adamawa State, Nigeria. He is a graduate of B.Sc. Computer Science and has Master Degree in Software Engineering from the St. Petersburg State University of Information Technology, Mechanics & Optics. St. Petersburg, Russia.

Asabe Sandra Ahmadu is a Lecturer of Computer Science in the department of Computer Science at the Federal University of Technology (Modibbo Adama University of Technology), Yola, Adamawa State, Nigeria. She is currently a Ph.D research fellow at Abubakar Tafawa Balewa University, ATBU, Bauchi, Nigeria. She is a graduate of BTech. Computer Science and has Master Degree in Computer Science from Abubakar Tafawa Balewa University, ATBU, Bauchi.