

Computational Music

R. K. JOSHI and SHRUTI PATEL, Indian Institute of Technology Dharwad

1 INTRODUCTION

Music is something that we all enjoy in the course of our lives. It is the sound that is brought together through a harmony of instruments and singing voices. It is something that is part of us ever since we enter this world. Many of us play musical instruments. To play any instrument, it is necessary to remember the notes. When a human plays an instrument, he is prone to make mistakes. He may either play a wrong note or may not keep up with the tempo of the song. To overcome these problems, we developed a simple tool that takes notes as input and a harmonious sound file is generated from the input notes.

The notes to music generator tool provides the following features:

- Adjusting the tempo of the music piece
- Setting the base frequency of the music piece
- Adding a fading effect to the music piece

The tool supports 34 "swaras" of different frequencies. The frequency ratios of the "swaras" are given in following table. Hence, if the base frequency is set to 140, then the frequency of "swara" S is 140, "swara" r is $(16/15)*140$ and so on.

Swara	Note	Frequency Ratio
S	C	1
r	C#	16/15
R	D	9/8
G	E	5/4
M	F	4/3
P	G	3/2
D	A	27/16
n	A#	9/5
N	B	15/8

2 INPUT

An input text file is given as an input to the tool. The input file format is shown in the figure 1. The first line specifies the amount of time in seconds each "swara" should be played. If the unit time is increased, the tempo of the music decreases and vice versa. The second line specifies the base frequency of the swara "sa". The third line should be the name of the output wav file. The next lines should contain the notes of the music piece that the user wants to generate.

Authors' address: R. K. Joshi; Shruti Patel, rkj@cse.iitb.ac.in, 160010002@iitdh.ac.in, Indian Institute of Technology Dharwad.

```

unit = 0.25
base = 110
output = birthday.wav
SSR$ S$M$ G$$$
SSR$ S$M$ G$$$
SSD$ P$M$ G$R$ $$$
MMG$ S$R$ S$$

```

Fig. 1. Example Input

Each line of the input file containing the notes are divided into groups. A group consists of individual "swaras" to be played. The "\$" symbol is used as an indication to continue playing the current "swara" for one more unit but with lesser amplitude to give a fading effect in sound.

3 FLOWCHART

- The text file is given as an input to convert.cpp process. convert.cpp parses the input file, allocates time to each "swara" according to the unit and groups and returns a temporary file storing this information.
- The temporary file is given as an input to create.awk process. create.awk reads the temporary file and generates a csd file that is readable by csound library.
- Finally, the csd file is given as an input to sound.cpp which returns the final wav file.

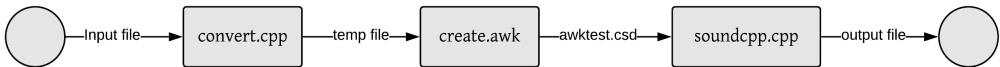


Fig. 2. Sequence of procedures