Schweizerschule Mexiko, CDMX
M.Sc. William Bombardelli
wbombardellis@gmail.com

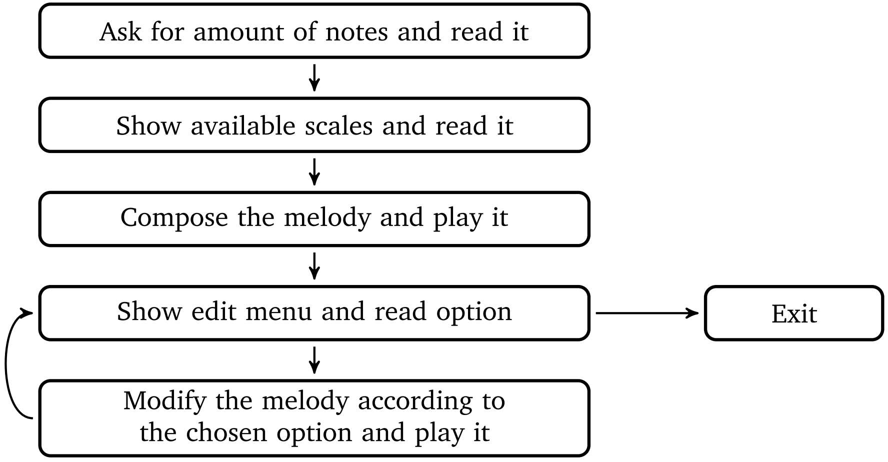
Assignment 2
Java

Colegio Suizo de México, A.C.
Schweizerschule Mexiko
México CDMX | Cuernavaca | Querétaro

1 Task

Write a program in Java that composes a melody and gives the user the chance to modify it. The program should first ask the user how many musical notes he/ she wants and in which scale should the melody be. After that, the program randomly samples the desired amount of notes from that scale and plays it for the user. Then, the program shows a menu of options to modify this melody. The user chooses one of the options and hears the new modified version of it. The program must repeat this cycle until the user chooses the option to exit.

Below is a diagram of the behavior of the program.


Ask for amount of notes and read it
Show available scales and read it
Compose the melody and play it
Show edit menu and read option
Exit
Modify the melody according to the chosen option and play it

The time between each note must be randomly chosen from 100ms to 1000ms. You can choose which scales you want to make available, three is the minimal amount, for example, natural major (C,D,E,F,G,A,B), natural minor (C,D,Eb,F,G,Ab,Bb) and locrian mode (C,Db,Eb,F,Gb,Ab,Bb). Other options available on internet include blues and jazz scales.

The edit menu options must be, at least, the following.

1: Up Pitch Shift *(Add a semi-tone to all notes in the melody)*
2: Down Pitch Shift *(Subtract a semi-tone to all notes in the melody)*
3: Stretch ×2 *(Double the time between each note)*
4: Squeeze ÷2 *(Divides by two the time between each note)*
5: Harmonize (add 5th) *(Creates a second melody consisting of the 5th degree of the main melody, that is, if the scale is the natural major and the note in the main melody is C, the note in the second melody is G - which is the 5th note in that scale)*

For the completion of this exercise, you will need the skills of *if*, *loops* (*while/for*) and *arrays*.

2 Groups

You may work in pairs or alone.

3 Work Plan

You will undergo a weekly check-in to present your advances. Every Wednesday you must write an e-mail to *wbombardellis@gmail.com* with the following items.

April, 29th	Present plan of implementation, which must include a sketch of how you code will look like; how you store the scales; how the melody gets created; how notes' duration gets defined.
May, 6th	Present basis of the source-code, which must include the first 3 steps of the diagram in Section 1
May, 13th	Present a working edit menu loop, with the exit option and one edit option implemented.
May, 20th	Present the fully-working edit menu loop with all options implemented.
May, 27th	**2nd Semester Exam**
May, 29th	Deliver the .java file with your source-code, via e-mail *wbombardellis@gmail.com*, until 23:59.

4 Delivery

Final delivery due to **May, 29th, 2020, 23:59**, via e-mail *wbombardellis@gmail.com*. A .java file containing the source-code. The file's name must contain the first and last names of all authors.

5 Evaluation

The grade will be given using the following schema:

- 80% Code correctness: Compiles and runs correctly with all features implemented.
- 20% Presentation: Code organization/ indentation
- Plagiarism implies invalid solution.
- Late delivery implies that you get only 70% of the grade.

6 Alternative

If you have your own idea for a program you may propose it until April, 24th. If it gets accepted you may write it, instead of solving the exercise described above.