# Sonic Pi Composition Lesson Plan #3: Algorithms in Music

*“My kind of composing is more like the work of a gardener. The gardener takes their seeds and scatters them, knowing what they are planting but not quite what will grow where and when - and they won't necessarily be able to reproduce it again afterwards either”*

–Brian Eno

## Lesson Overview:

The aim of this lesson is to learn about generative and algorithmic music, and to **start student’s own individual project**. They will use new Sonic Pi commands that **enable generative and algorithmic music** (Mozart, John Cage, Brian Eno). Students will also experiment with **basic audio sample manipulation**.

Download a PDF version of this lesson

## Contents:

## Learning Outcomes:

| Key concepts | Sonic Pi syntax to be taught this lesson | Interdisciplinary Curriculum Links | | | Learning Outcomes |
|---|---|---|---|---|---|
| | | Computational Thinking | Programming | Music (strands) | |
| Music:<br>-texture<br>-mood | **Activity #1**<br>`rate`<br>`cutoff` | Decomposition, conditional logic, making algorithms | Methods, selection, arguments, parameters | PK, DI, UC | Music:<br>-All students will identify and experiment with characteristics of algorithmic music and mood/timbre |

| | | | | | |
|---|---|---|---|---|---|
| -sampling<br>-stretching, layering, cuttoff/highpass<br><br>Programming:<br>-random number generator<br>-selection<br>-arguments | **Activity #2**<br>`rrand()`<br>`.choose`<br>`use_random_seed`<br>`if else` | | | | -All students will be introduced to sampling and experiment with basic sample manipulation<br><br>Programming:<br>-All students will use a random number generator (`rrand`) within Sonic Pi<br>-All students will experiment conditional logic (`if/else`)<br>-All students will use arguments to stretch and sculpt audio samples with `.rate` and `.cutoff` |

# **Introduction:** What did we cover in the last lesson **(5 minutes)**

**Activity Overview:** Every lesson in this unit of work starts with refreshing knowledge and identifying gaps in understanding from the previous lesson.

**Student Activity:**

1. For three minutes, students are to reflect on the code they saved at the end of the previous lesson
2. The teacher should roam the room, quickly surveying the code produced and ask a sample of students about key music and programming concepts covered in the previous lesson
3. If the teacher deems necessary, have a 2-3 minute discussion on identified concepts that need to be reinforced

**Notes to the Teacher:**

- Reflection will begin each lesson in this unit and the teacher should try to get around to all groups every two lessons

# **Activity 1:** Introduce brief for individual project **(30 minutes)**

**Activity Overview:** This activity gets students to start on their individual project. Their individual project is identical to their group project only it does not require collaboration.

New Sonic Pi syntax to introduce in this activity (click for example code):

`rate`

`cutoff`

`sample :mehackit_phone1`

`sample :ambi_choir, rate: 0.3, cutoff: 70`

**Suggested Teacher Instruction Sequence:**

1. Introduce individual composition brief individual project (individual project is identical to their group project only it does not require collaboration)
2. Play video of using everyday objects to make sound effects video [3 minutes] https://www.youtube.com/watch?time_continue=2&v=7oN1JJughVw
3. Discuss the concept of using everyday objects like telephone sounds to add texture and or rhythms e.g. `sample :mehackit_phone1` (code that plays a sample of a phone)
4. Demonstrate how using arguments like “`rate`” and “`cutoff`” can manipulate a sample being played (stretch, speedup, fade in and fade out etc). E.g. `sample :ambi_choir, rate: 0.3, cutoff: 70`

**Suggested links and resources to facilitate activities:**

- Using everyday objects to make sound effects video https://www.youtube.com/watch?time_continue=2&v=7oN1JJughVw

**Student Activity (20 minutes):**

Instruct students to do the following (similar to the process of beginning their group project):

1. Students in their groups are to first choose a video from https://chrispetrie.github.io/teachervideos.html (different from their group composition)
2. Students brainstorm appropriate sounds
3. Students to explore sounds in Sonic P using https://chrispetrie.github.io/teacherpreviewsamples.html
4. Start individual project (20 minutes)

**Notes to the Teacher:**

- The teacher or student could use any video that they want to make music to. However, this unit has been designed for short 1-2 minute videos on climate change, pollution, and the refugee crisis so that they engage musically with meaningful and current contexts
- Remember initial brainstorms on appropriate sounds for students’ chosen films are often only starting points → the intention should be encourage experimentation and find solutions from unexpected sounds/sound manipulation
- Remember if students are struggling for ideas – ask them to just explore and survey the sounds available in Sonic Pi. If they get stuck after this, an idea could be that they could listen to film music they like for inspiration

# **Activity 2:** Introduction to generative music with Sonic Pi and student time on individual projects **(30 minutes)**

**Activity Overview:** Students will be introduced to using random number generators in Sonic Pi to create music that never plays the same twice.

New Sonic Pi syntax to introduce in this activity (click for example code):

`rrand()`

`.choose`

`use_random_seed`

`if else`

**Suggested Teacher Instruction Sequence:**

1. View Mozart's musical dice game (only need to watch a sample part of this video) https://www.youtube.com/watch?v=9Zdg6Ec4mVw&t=1s. One musical Dice example with harpsichord https://www.youtube.com/watch?v=Jtpb1DkPx7I
2. Discuss conventional music and the history of recorded music → the idea of playing a piece of music in exactly the same way twice has only become prevalent since the invention of recorded music. Additionally, written music has only being around since 1473 (music has been around for much longer in all cultures), with the invention of the printing press.
3. Explain how this idea of 'chance based music' or music that does not play the same twice has become more prevalent in modern experimental music.
4. A modern example of generative music is with Brian Eno's app - https://www.youtube.com/watch?v=Dwo-tvmEKhk
5. Demonstrate and discuss the Sonic Pi syntax that utilises random generated numbers to make systems based music `rrand(), .choose, use_random_seed, if else`
6. Explain how `rrand()` can be used within arguments. E.g. `play 60, amp: rrand(0.2, 0.6)` which randomly generates amplitude

**Suggested links and resources to facilitate activities:**

- Mozart's musical dice game https://www.youtube.com/watch?v=9Zdg6Ec4mVw&t=1s
- Musical Dice example with harpsichord https://www.youtube.com/watch?v=Jtpb1DkPx7I
- Brian Eno generative music app example - https://www.youtube.com/watch?v=Dwo-tvmEKhk

**Student Activity (20 minutes):**

1. Students are to experiment with the following syntax `rrand(), .choose, use_random_seed, if else` in order to create music that continuously changes

**Notes to the Teacher:**

- Experimental composer John Cage wrote a piece called ‘I Ching divination’, which involves obtaining a hexagram by random generation (such as tossing coins), then reading the chapter associated with that hexagram

## **Activity 3:** Listening and reflecting **(30 minutes)**

**Activity Overview:** Standups as per lesson #2. Students also get a chance to listen to each other’s work and give constructive feedback.

**Suggested Teacher Instruction Sequence:**

→ standups introduced in Lesson #2

1. Brief students on using constructive and helpful feedback to their classmates

**Student Activity (15 minutes):**

1. Give students 2 minutes to think of what to say and who is going to go first (these will be conducted every lesson)
2. Commence standups. Tell students that they will be completing a reflection diary at the end of the lesson and they can use the information they talk report in their stand-up for this
3. Give students 5 minutes to listen to each other’s work and give constructive feedback

**Notes to the teacher:**

- As per lesson #2

## **Wrap-up activity:** Quiz and reflection **(15 minutes)**

**Activity Overview:** All students to complete a quiz containing 10 questions on music and programming -  as well as a few reflective questions on this lesson (all students will complete this each lesson).

**Student Activity (10 minutes):**

- Students individually complete the quiz and reflection on the key concepts in this lesson within 10 minutes linked here.

**Administrative Details**

| | |
|---|---|
| **Contact info** | petriechris@gmail.com |
| **Credits** | Developed by Chris Petrie. |
| **Last updated on** | 27/06/2018 |
| **Copyright info** | The content of this page is licensed under the Creative Commons Attribution 3.0 License, and code samples are licensed under the Apache 2.0 License. |