

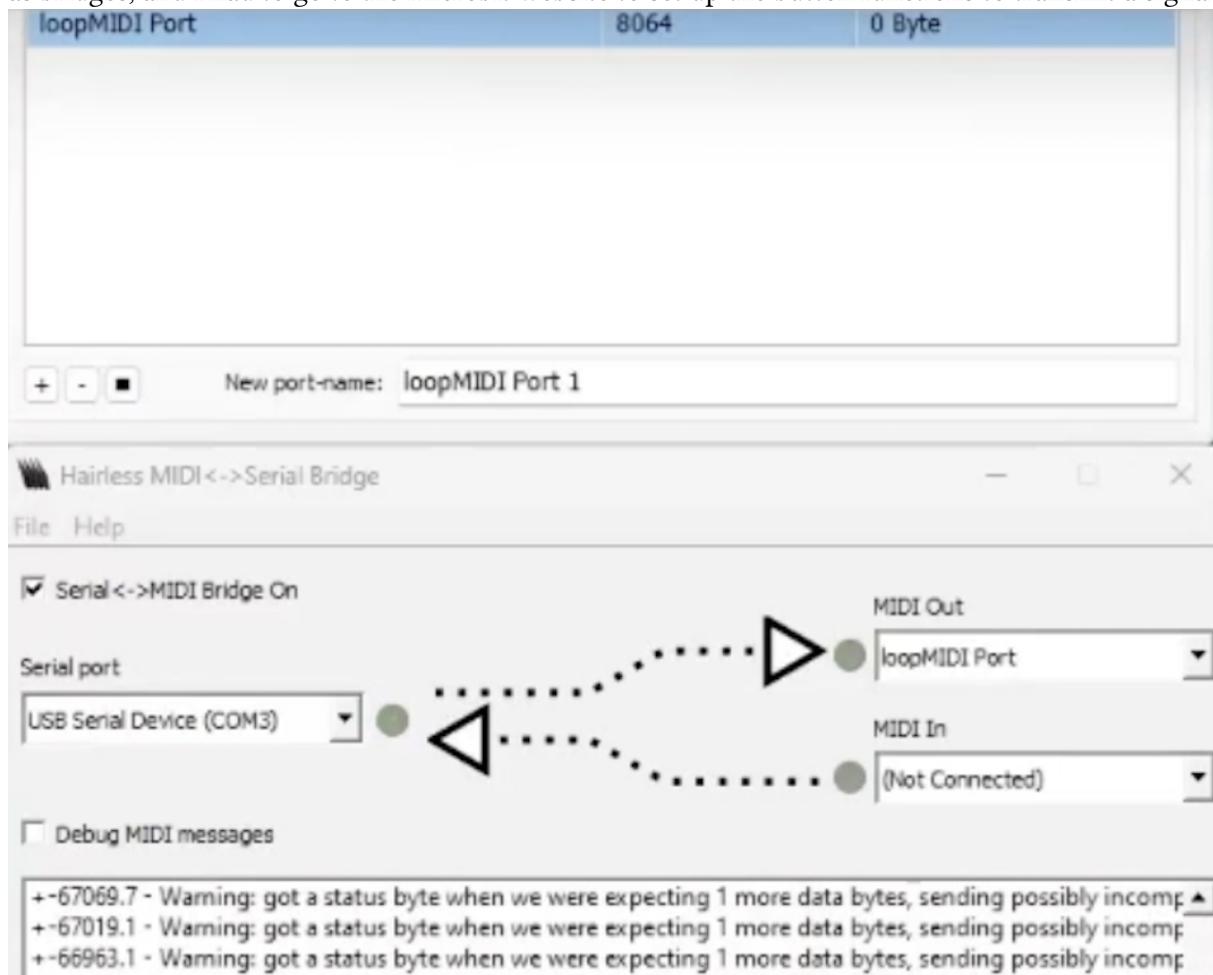
# SMC: Portfolio 2

u7338066

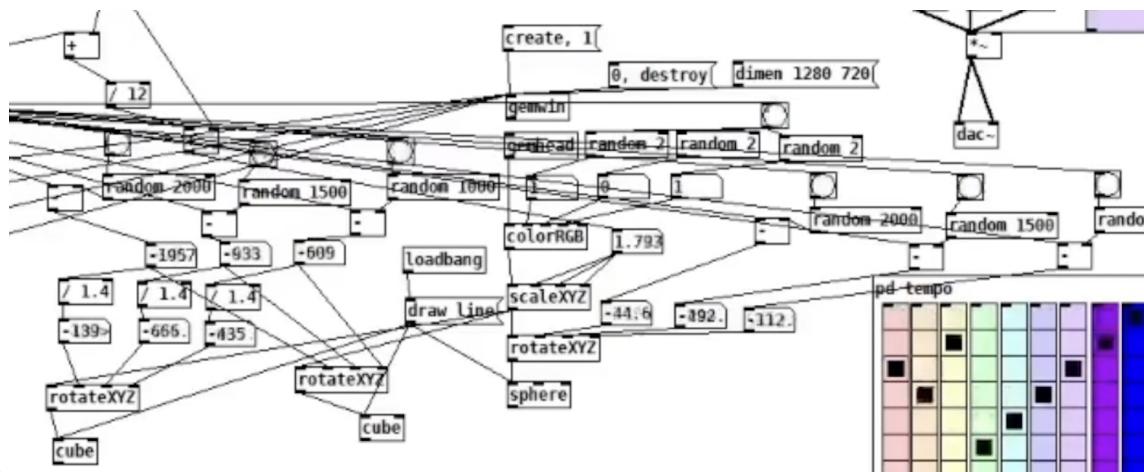
2023-05-15

## Week 6 reflection

This week's theme led me to adapt a piece from a fellow classmate and add a new interface. I chose this piece purely because I thought the original author did a great job and I wanted to understand their code while adapting it. My idea was to use Gem and Processing, along with some built-in Pd GUI elements, to complete the visualization and interface. I would use Processing for part of the interface operations, and then use PdParty on an iPad to operate some sliders and either a Microbit or Microsoft controller to operate the remaining buttons to change the state of the piece. However, due to technical limitations, I was unable to download the iOS version of PdParty, so I unfortunately had to omit that part and move it to the interface operations in Processing. I spent a lot of time figuring out how to connect the Microbit and creating Gem animations. Connecting the Microbit required several apps as bridges, and I had to go to the Microbit website to set up the button functions to transmit a signal.

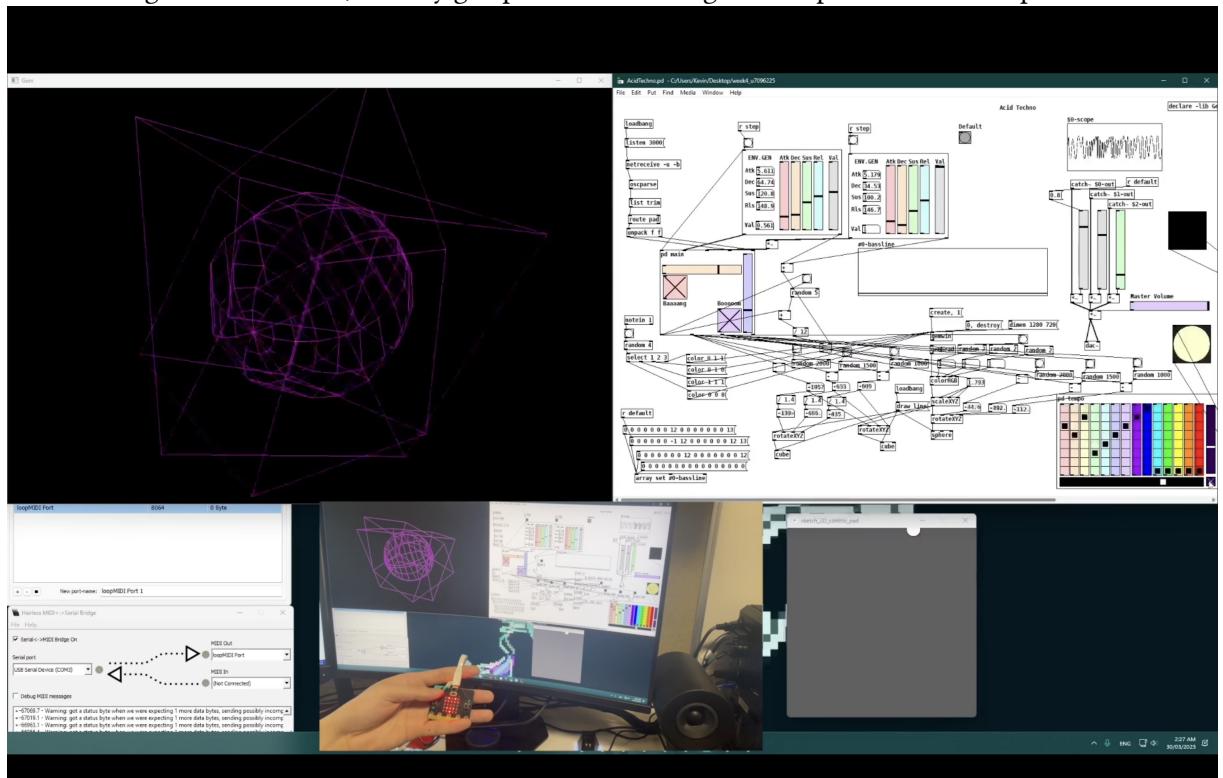


The Gem animations required many parameters, so I decided to use various randomizations to react to



each beat of the music.

Finally, one of the biggest challenges was understanding the original author's code. After asking the author and understanding their intentions, I finally grasped the overall logic of the patch and come up with



this piece.

## References

1. Unkown. 2020. How do I play a MIDI instrument on the micro:bit. <https://support.microbit.org/support/solutions/article/19000053392-how-do-i-play-a-midi-instrument-on-the-micro-bit>
2. Fergo. 2004 - 2019. Fergo JoystickMIDI. Used this to connect the controller. <https://www.fergonetz.net/softwares/fjoy>
3. Nikola Nikita Jeremic (Youtube user). 2020. How to connect your game controller as a MIDI device in Studio One. Retrieved from: <https://www.youtube.com/watch?v=UCT6-CzRAZk>
4. Sound Simulator (Youtube user). 2022. How to Create Live Visuals in Pure Data!. Learn from this tutorial. Retrieved from: [https://www.youtube.com/watch?v=y1rBa\\_STq64&t=278s](https://www.youtube.com/watch?v=y1rBa_STq64&t=278s)

## Week 8 reflection

In this reflection, I would like to discuss my experience in composing and arranging music using Charles and Gibber's built-in tutorials and demos. I experimented with a variety of instruments and explored the convenience that Gibber offers along with my familiarity with coding. For my lead, I chose to base my composition on the chords of a canon, and for my bass, I used a simple melody to accompany the lead. To add rhythm and beat, I incorporated not only drums but also other instruments like Clave, kick, and hat. These additions helped to enhance the percussion and pulse of the piece.

The most challenging part of this project was working with the different instruments and combining them to create a cohesive composition. I spent a lot of time experimenting with the various instruments and understanding how they worked together. I have made a helpful discussion with the teammates on workshop, so I begin with the piece I have made on the workshop with my teammates.

gibber      intro      ↴      share      gabber      restart engine

```

Clock.bpm = 120

// Seventh
bass = Monosynth('bass')
bass.note.seq( [1, 5, 6, 3, 4, 1, 2, 5], 1/4 )
bass.gain = 0.3
bass.pan.seq( [0.5, 0.7, 0.9, 1, 0.5, 0.3, 0.1, 0 ], 1/4 )
bass.Q = gen( 0.5 + cycle(0.1) * 0.49 )
bass.cutoff = gen( 0.5 + cycle(0.07) * 0.45 )

lead = Monosynth('lead')
lead.note.seq( [1, 2, 3, 4], 1 )
lead.gain = 0.15

bass.stop()
lead.stop()

synth = Synth('square.bass')
synth.note.seq( p = [2, 4, 2, 4, 3, 5, 3, 5], 1/8 )
synth.gain = 1.5
synth.pan.seq( [0.5, 0.3, 0.1, 0, 0.5, 0.7, 0.9, 1], 1/4 )

// Ninth
p.transpose.seq( 1, 1 )
p = [2, 5, 4, 1, 4, 3, 2, 1]
p.reset.seq( 1, 2 )

// Eighth
verb = Reverb( 'space' ).bus()
delay = Delay( '1/3' ).bus().connect( verb, .1 )
perc = FM[3]( 'perc' )
.connect( delay, .65 ).connect( verb, .35 )
.spread(.975)
.note.seq( sine( btof(2),1,3 ), 1/4, 0 )
.note.seq( sine( btof(4),2,4 ), 1/16, 1 )
.note.seq( sine( btof(6),3,5 ), 1/8, 2 )
.loudness.seq( sine(4.33,.35,.7) )
.gain = 0.05

// First
k = Kick('tight')
k.trigger.seq( 1, e = Euclid(5, 8) )

// Second
c = Clave()
c.trigger.seq( 0.5, Euclid(3, 8) )

// Third
d = Drums()
d.fx.add( Distortion({ pregain:1.5, postgain:1 }) )
d.tidal('kd [kd, sd] kd [kd, sd]')

```

```

// Fourth
f = FM('glockenspiel.short')
f.note.tidal( '0*2 2*2 <4 5> [7 8]' )

// Fifth
h = Hat({ decay:.0125 })
h.trigger.seq( [1,.5], 1/8, 0, 1/2 )
h.fx.add( Distortion({ pregain:100, postgain:.1 }) )

// Sixth
b = Julia().render()
  .scale(2)
  .fold.seq( [1, 2, 3, 4, 5], 1/4 )

lead.decay.seq( [1/32, 1/16, 1/8], 1/4 )

e.rotate.seq( 1, 1 )

// Ending
bass.gain.fade( null, 0, 4 )
lead.gain.fade( null, 0, 4 )
synth.gain.fade( null, 0, 4 )
lead.gain.fade( null, 0, 4 )

k.stop()
c.stop()
d.stop()
f.stop()
h.stop()

```

## References

1. charlie roberts. 2022. gibber 2: intro tutorial. Learn through this video. Retrieved from: <https://vimeo.com/728710492>
2. charles martin. 2021. Laptop Music Coding with Gibber. Learn through this tutorial. Retrieved from: <https://charlesmartin.au/blog/2021/01/10/laptop-music-workshop>

## Week 9 reflection

In this group diary project, we had multiple meetings and discussions to create our musical performance. During our first two meetings, we discussed our creative direction and allocated roles for each member to play. Our musician, Noah, composed the chord progressions for our piece, and I was responsible for playing the lead and some of the clave, kick, and hat. To learn how to write a lead that could accompany the chord progressions, I spent a lot of time researching online tutorials and created two leads for our performance. One lead had four fixed patterns, while the other used functions that could generate random notes based on the chord progressions. I also developed various ad hoc functions, such as range, rotate, and transpose, to



Noah Rose 28/4 3:25 pm Edited

Drum - Wang Zilong

Bass - Wenshilan Ren

Harmony - Noah Rose

Lead - Kevin Zhu

Samples/Pluck - Wangshu Cai

Structure:

Intro - written melody

Middle - improvised, free, live-coding

Outro - written melody

enhance the second lead.

As we progressed, we tried out Gibber in our meetings and fine-tuned our coordination. We even managed to crash seven rooms in the process! Finally, we scheduled a meeting to plan the performance's order and worked on it in five recording sessions in a library room.



Noah Rose 28/4 4:15 pm

Here's a cool chord progression i think we could use for our diary, especially the intro and outro sections: ([[3,5,7],[1,4,6],[2,4,6], [2,5,7]],1/2)

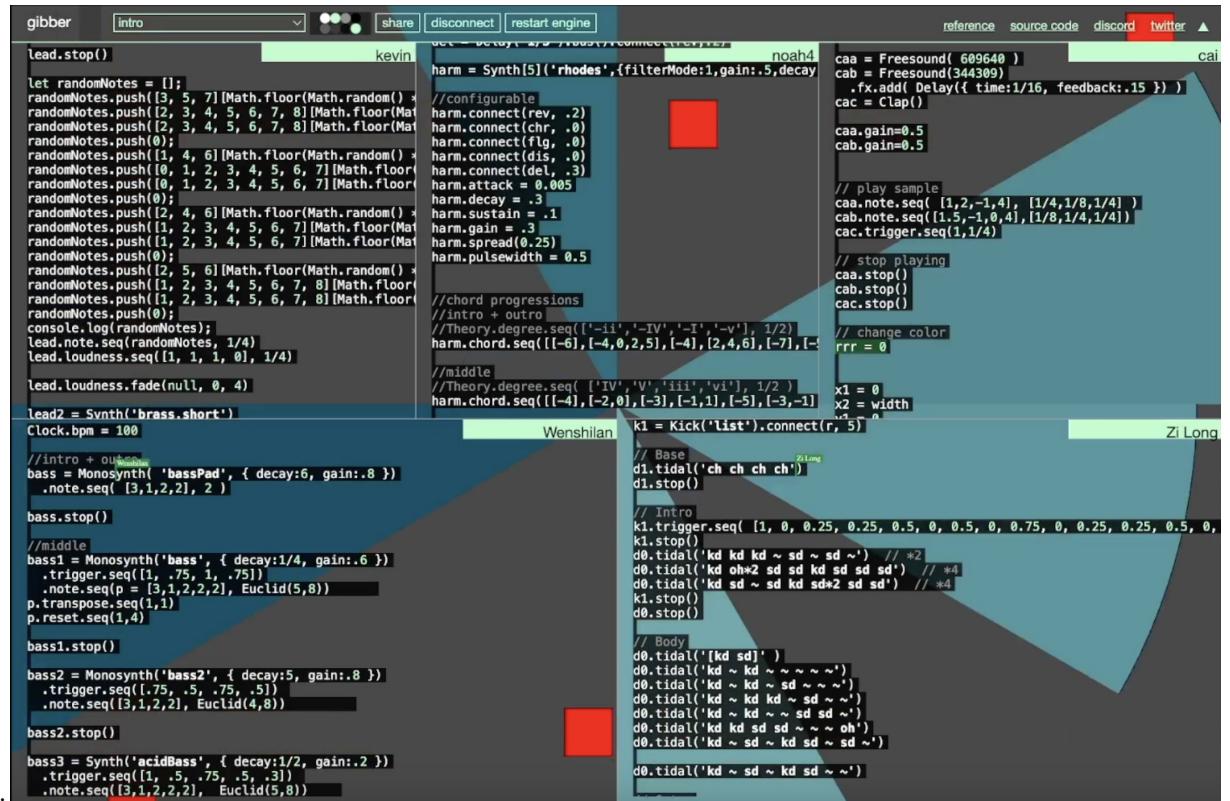
(Not really sure how best to share this so I hope this is good 😊)



Kevin Zhu 28/4 4:31 pm

Now learning how to produce leads from chords, hope I get it 😊

Overall, this project was a great learning experience in collaborating with others and developing our musical skills. We faced many challenges along the way, but we overcame them with determination and teamwork. I enjoyed working with my group members and look forward to collaborate with them again in the

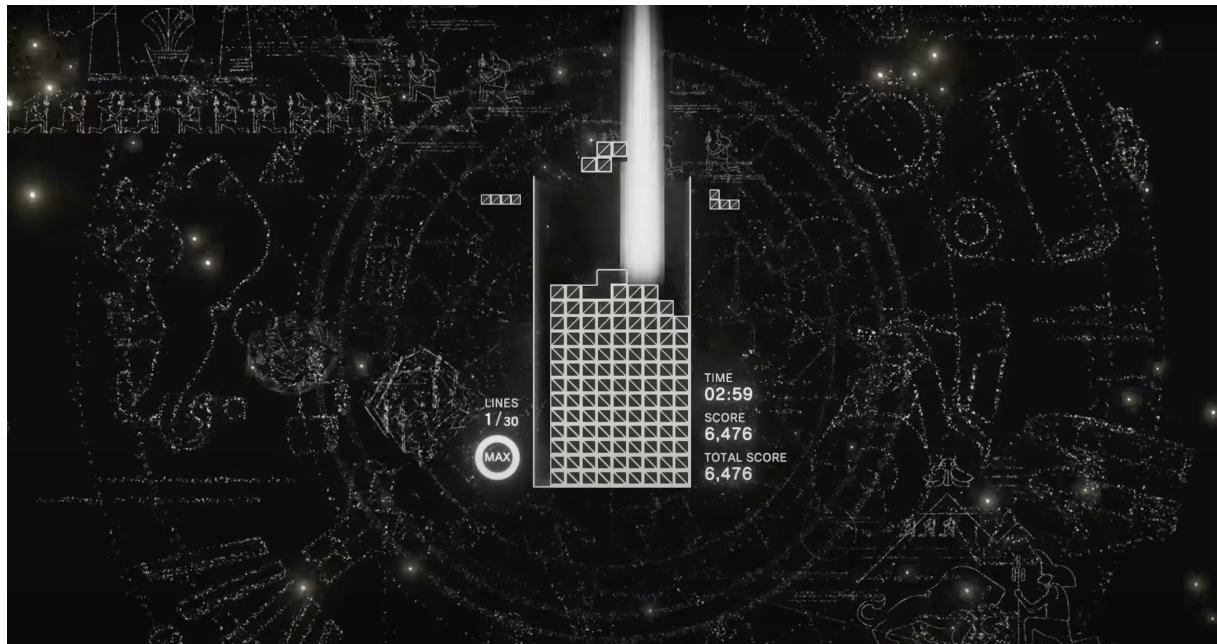


## References

1. r/guitarlessons forum. 2020. How to create leads from chords?. Retrieved from: <https://www.reddit.com/r/guitarlessons>

## Week 10 reflection

For my final project, I decided to create a game that I really enjoy playing called Tetris. I focused on creating sound effects for the game, as the visualization and game mechanics of Tetris are relatively simple. The movement, elimination, and rotation of the blocks can all interact with the sound effects, creating a unique audio-visual experience. I planned to use the changing patterns of the Lead and chords to guide the game into different musical themes, with the game's background colour and particle effects adapting to the music. The result may be like a Tetris-based game called Tetris: Effect.



During the workshop and team meetings, I discussed and decided on each person's role and collaboration in detail. However, many parts of the project are still unfinished. The biggest challenge for me has been synchronizing the visuals and music while allowing for player interaction with the visuals.

Final performance

Idea from TETRIS EFFECT.

Maybe Noah  
(LEAD, Bass, chords) (Drum, Kick, snare)

Tools: Make sure to cover everything we learn  
(Main, Frontend)

2 MEMBERS 1. **Gibber** — Mainly where the sound's from.  
(Main, Frontend)

1 MEMBER 2. **PureData** — Control samples, play sound with synthesizer  
(Support, Back) and controller of physical "players".

1 MEMBER 3. **P5.js** — which runs on gibber so that variables  
(Main, Front) Can be shared.

1 MEMBER 4. **Physical player** — Not decided yet.  
(Main, Front)  
Maybe based on Camera/microbit/  
Microphone.

Must be something interesting!

ME

MEMBER 1

MEMBER 2

MEMBER 3

MEMBER 4

So basically it is Tetris game which is similar to Tetris Effect, but mainly focus on the sound, not the particle effects. When the blocks move/rotate/drop, sound will cooperate (maybe drums). The theme of the game will start with a slow and dark theme. When gamer reach the goal (or stacks higher) the sound/rhythm gets faster, background gets brighter, and particles increase.

Gives presets and full controls to the performers so that they all can change the code and play lively. But should all based on the situation of the game.

In addition, I have left a lot of flexibility in every part of the performance for each member. Everyone can play in their style and code lively to change the piece but should all fit the theme planned.

```
// Things borrowed from last diary Member3
Clock.bpm = 200
rev = Reverb('space').bus()
chr = Chorus('lush').bus()
flg = Flanger('moderate').bus()
dis = Distortion('earshred').bus()
del = Delay('1/3').bus().connect(rev,.2)
harm = Synth[5]('rhodes',{filterMode:1,gain:.5,
d0.stop()})
```

```
// Intro
k1.trigger.seq([1, 0, 0.25, 0.25, 0.5, 0, 0.5, 0, 0.75, 0, 0.25, 0.25, 0.5, 0, 0.5, 0], 1/16)
k1.stop()
d0.tidal('kd kd ~ sd ~ sd ~') // *2
d0.tidal('kd oh>2 sd kd sd sd sd') // *4
d0.tidal('kd sd ~ sd kd sd>2 sd sd') // *4
k1.stop()
d0.stop()
```

```
use('p5') Member1
// The 'Gibber' version is still not completed
// Still working in javascript
// Basically something like this
// but much more fancy and more particles surround
// Things borrowed from last diary Member2
Clock.bpm = 200
rev = Reverb('space').bus()
chr = Chorus('lush').bus()
flg = Flanger('moderate').bus()
dis = Distortion('earshred').bus()
del = Delay('1/3').bus().connect(rev,.2)
harm = Synth[5]('rhodes',{filterMode:1,gain:.5,decay:.5,attack:.001,Q:.7,useADSR:true})
```

```
Member3
Member2
Member1
```

This is the backend, where three members will be controlling|

Synthesizers| Sample Space|

better-fm.pd

Log: 2 error openin

gibber.cc

```
use('p5') Member1
// The 'Gibber' version is still not completed
// Still working in javascript
// Basically something like this
// but much more fancy and more particles surround
// Things borrowed from last diary Member2
Clock.bpm = 200
rev = Reverb('space').bus()
chr = Chorus('lush').bus()
flg = Flanger('moderate').bus()
dis = Distortion('earshred').bus()
del = Delay('1/3').bus().connect(rev,.2)
harm = Synth[5]('rhodes',{filterMode:1,gain:.5,decay:.5,attack:.001,Q:.7,useADSR:true})
```

```
Member1: This is the Frontend (What the audience can see)
Member1: This is the Frontend (What the audience can see)
Member1: This is the Frontend (What the audience can see)
Member1: This is the Frontend (What the audience can see)
Member1: This is the Frontend (What the audience can see)
Member1: This is the Frontend (What the audience can see)
```

Screenshot of week 10 (Part 3) ## References 1. Nick Robinson (Youtube user). 2019. 10 Minutes of TETRIS EFFECT Music and Gameplay. Inspired by this game. Retrieved from: <https://www.youtube.com/watch?v=urbLIyd-VsQ&t=412s>

2. Easy websify (Youtube user). 2023. How to Make a Tetris game using p5.js - Step by step | JS Game EP-2. Learn through this tutorial. Retrieved from: <https://www.youtube.com/watch?v=wLZRCH3yMl8&t=1s>
3. SolidCode (Youtube user). 2019. Programming a Tetris Game with JavaScript and P5.js (Part I - Gravity). Learn through this tutorial. Retrieved from: [https://www.youtube.com/watch?v=Wcb0\\_Q9r6i4&list=PLU1XJPC](https://www.youtube.com/watch?v=Wcb0_Q9r6i4&list=PLU1XJPC)
4. SolidCode (Youtube user). 2019. Programming a Tetris Game with JavaScript and P5.js (Part I - Gravity). Learn through this tutorial. Retrieved from: [https://www.youtube.com/watch?v=Wcb0\\_Q9r6i4&list=PLU1XJPC](https://www.youtube.com/watch?v=Wcb0_Q9r6i4&list=PLU1XJPC)
5. nnxa (Github user). 2019. tetris. Learn from this code. Retrieved from: <https://github.com/nnxa/tetris>

## Portfolio 2 Discussion

### Intro

In the past four weeks, with the help of my teammates, teachers, and friends, I have made continuous progress in my understanding of music and computer music. While challenging myself with the fresh Gibber, I also consolidated my knowledge of Puredata and revisited p5js and various plugins that I had learned before. Overall, this four-week learning journey has been extremely beneficial to me.

### Discussion

**As a group,** I have learned more about collaborating with others to create music. It's important not to impose your own ideas on everyone, but to take in everyone's suggestions. In Week 9's group diary, we got to know each other's strengths and weaknesses through multiple meetings and helped each other progress. For example, Wangshu had no knowledge of music but was familiar with coding, while Noah was a music major but the opposite. We have treated each other respectfully and in the end, each of us had a more comprehensive understanding of music and coding. For example in week 9 and week 10, we ask for each others opinions and give suggestions. In addition, collaborating with different people during workshops has helped me improve. Expressing my own ideas, listening to others' suggestions, and working together to create unknown works through live coding.

**As an individual,** I have tried many extracurricular activities, such as connecting Microbit, learning and using p5js, Processing, and Gem. Every time I see my classmates mastering knowledge that I don't know in class, I will ask them for their patches, hoping that I can use them next time. Whenever I get stuck, I ask my teammates on Teams, and they patiently help me solve the problem. Like in the week 6, it showcases the ability to adapt existing code to create a new interface. I experimented with different tools and techniques to connect the Microbit and create Gem animations. I tried to highlight the original author's technical skills, especially in working with different technologies. I appreciate the chance to learn from others patches, which allows me to absorb knowledges from others. It's clear that my learning journey has been a gradual process. The UI exploration I did in solo performance has been carried over to Puredata later on, while the music production skills I learned in Gibber as part of a group will definitely be used in my final performance.

### Important Lessons

The importance of collaboration: Collaboration is crucial in music computing, as it involves a wide range of skills and knowledge. By working with others, I learnt from their strengths and weaknesses, and create

Noah Rose 28/4 7:43 pm

I think different chords would be a good idea actually for different parts, maybe we could use this other chord progression as well

The chord progression I chose I think will work better in the middle, live-coding section so we could use the a

Just a suggestion tho 😊

Then I will design three different leads for intro ou

Ah, just to make sure I'm in the right track tho, melodies is kinda the s

To

Noah Rose 28/4 7:50 pm

Yeah, usually the lead instrument in a band or ensemble plays a melody, usually based around the chords more complex and interesting projects.

The power of experimentation: Experimentation is a critical aspect of music computing, as it allows me to try out different ideas and see what works best. In week 9, I have tried the possibility to make a random

```
let randomNotes = [];
randomNotes.push([3, 5, 7][Math.floor(Math.random() * 3)]);
randomNotes.push([2, 3, 4, 5, 6, 7, 8][Math.floor(Math.random() * 3)]);
randomNotes.push([2, 3, 4, 5, 6, 7, 8][Math.floor(Math.random() * 3)]);
randomNotes.push(0);
randomNotes.push([1, 4, 6][Math.floor(Math.random() * 3)]);
randomNotes.push([0, 1, 2, 3, 4, 5, 6, 7][Math.floor(Math.random() * 3)]);
randomNotes.push([0, 1, 2, 3, 4, 5, 6, 7][Math.floor(Math.random() * 3)]);
randomNotes.push(0);
randomNotes.push([2, 4, 6][Math.floor(Math.random() * 3)]);
randomNotes.push([1, 2, 3, 4, 5, 6, 7][Math.floor(Math.random() * 3)]);
randomNotes.push([1, 2, 3, 4, 5, 6, 7][Math.floor(Math.random() * 3)]);
randomNotes.push(0);
randomNotes.push([2, 5, 6][Math.floor(Math.random() * 3)]);
randomNotes.push([1, 2, 3, 4, 5, 6, 7, 8][Math.floor(Math.random() * 3)]);
randomNotes.push([1, 2, 3, 4, 5, 6, 7, 8][Math.floor(Math.random() * 3)]);
randomNotes.push(0);
console.log(randomNotes);
lead.note.seq(randomNotes, 1/4)
lead.loudness.seq([1, 1, 1, 0], 1/4)
```

lead base on the chord progressing.

The importance of learning multiple tools: Music computing involves a wide range of software and hardware tools, and learning to use them can greatly expand my creative possibilities. For example, in week 8, I tried to use microbit all by own before the lecture teaches.

### Common threads

A focus on sound design: Many diaries discuss the importance of sound design in music computing, and how it can be used to create unique and interesting compositions.

A focus on experimentation and exploration: Many diaries also emphasize the importance of experimentation and exploration in music computing, and how it can lead to new and exciting ideas.

A focus on collaboration and community: Finally, many diaries highlight the importance of collaboration and community in music computing, and how working with others can help you learn and grow as a musician and a person.

### Conclusion

In conclusion, my journey in music computing has been a valuable experience that has taught me many important lessons. Through collaboration, experimentation, and learning multiple tools, I have gained a more comprehensive understanding of music and computer music. As I continue to explore this field, I will keep these important lessons in mind and continue to focus on sound design, experimentation, and collaboration with the community.

I am grateful to my teammates, teachers, and friends for their support and guidance throughout this journey. I look forward to applying what I have learned in my future projects and performances such as the upcoming LENS.

### References

1. Learn from the example resources given by Charles. Retrieved from: <https://github.com/cpmpercussion/ComputerMusic>
2. SMC2023. 2023. Get inspired by all the classmates, viewed almost everyone's video each week. Retrieved from: <https://anu365.sharepoint.com/sites/SMC2023>