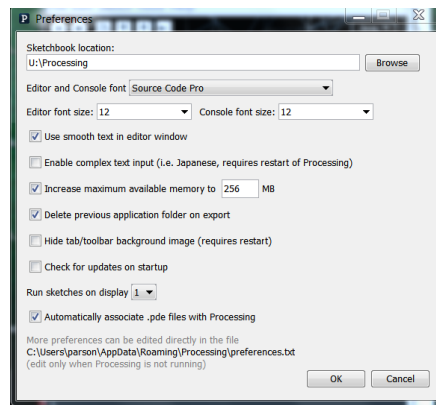


CSC 480 Object Oriented Multimedia Programming, Spring 2020

Dr. Dale E. Parson, Assignment 4, dome music server with chords controlled by Android client.¹

This assignment is due via **D2L Assessments** -> **Assignments** -> **Assignment 1 Chords** by **11:59 PM on 19 February**. There is a video of a brief demo at <https://youtu.be/DgozbuXYrY>.

When using Processing on the Kutztown campus Windows computers, make sure to start out **every time** by setting your Processing Preferences -> Sketchbook Location to U:\Processing. The U:\ drive is a networked drive that will save your work and make it accessible across campus. If you save it to your desktop or the lab PC you are using, you will lose your work when you log out. You must save it to the U:\ drive. If you do not have a folder called Processing under U:\, you must create one using the Windows Explorer. Processing Preferences is under the File menu on Windows.



If you will be downloading Processing 3.X and running it using an off-campus computer (do not use version 2.X for assignments), you can copy your project sketch to a flash drive on one machine, and then copy it from the flash drive to another Processing sketch folder.

Pitfall: You will create sketch **CSC480S20Androidassn1Client** from handout sketch **CSC480S20Androidassn1Client**, with editor tabs for **Musician**, **Note**, **CartesianPolar**, **Menus** and **LoopState**. You will also copy and save sketch **CSC480S20MIDIassn1Server** with its own, distinct **Musician**, **Note**, and **CartesianPolar** tabs. Be careful to copy & paste the correct tabs from my handouts on the course page.

1. Copy Dr. Parson's **CSC480S20Androidassn1Client** sketch on the course page into a new Processing editor and Save As **CSC480S20Androidassn1Client**. Save it after each step.
2. One at a time, copy the **Musician**, **Note**, **CartesianPolar**, **Menus** and **LoopState** tabs from **CSC480S20Androidassn1Client** and paste them as new tabs into **CSC480S20Androidassn1Client** tabs. I have made changes since the pre-assignment handout, so make sure to copy them from the Assignment 1 **CSC480S20Androidassn1Client** section on the course page.

¹ If you receive an error message **The package oscP5 or netP5 does not exist** when attempting to run with OSC for the first time, go to **Sketch** -> **Import library** -> **Add library** on the editor's menu, **Filter** on **oscP5**, click and Install this networking library on your PC or laptop. Also, the first time you try to change from **Java mode** to **Android mode** (step 13 below) at the upper right corner of the editor, you will have to walk through **Add mode** -> **Android mode** -> **Install**. After **Install** completes, switching to Android mode the first time walks you through **Download** (android) **SDK automatically** and its installation. This SDK is the underlying Google library for programming in Java on Android.

3. Copy Parson's **CSC480S20MIDIassn1Server** sketch into a new Processing editor and Save As **CSC480S20MIDIassn1Server** (same name).
4. One at a time, copy the **Musician, Note, and CartesianPolar** tabs from **CSC480S20MIDIassn1Server** and paste them as new tabs into tabs, similar to steps 1 and 2 above. Save **CSC480S20MIDIassn1Server**. You must also save the .svg vector files and .png image file from the course page into directory **CSC480S20MIDIassn1Server**.
5. Test the client and server sketches on a single PC or laptop, entering IP address 127.0.0.1 in the client menu user interface, since you will not be using a wireless network or Android tablet. 127.0.0.1 is "localhost", meaning client-server network messages stay on that single computer.

All of your required changes are documented with an upper-case **STUDENT** comment in the code. Here is the list from each sketch.

CSC480S20MIDIassn1Server main tab:

/ STUDENT NAME:** Enter your name at the top of the tab. **10 point deduction if missing.**

```
// STUDENT: (15%) Extend this function to copy tonic and scaleix from the OSC
// message into the ClientMessage object constructed here.
// Make sure to update the args.length test to account for the two new
// fields for tonic and scaleix.
void oscEvent(OscMessage theOscMessage) {

// processOSCEvents() runs in the Processing draw()'s thread
// STUDENT: Extend this function to pass tonic and scaleix from the
// incoming ClientMessage object to function m.playNote below.
void processOSCEvents() {
```

CSC480S20MIDIassn1Server Musician tab:

```
// STUDENT: (10%) Extend this function to pass tonic and scaleix from the
// incoming function call to function note.display() below
// by adding parameters to playNote(). When you try to run you will
// see errors in one or more calls to playNote() because of the added
// parameters tonic and scaleix. For an incoming OSC message those
// parameters must be the values arriving in the OSC message.
// For mousePressed played notes in the main server draw() code, just
// pass arguments of -1 for these new parameters.
void playNote(Note note, boolean isSounding) {
```

CSC480S20MIDIassn1Server Note tab:

```
// STUDENT: (10%) Extend this function to add parameters tonic and scaleix
// to this function. Add code after the final popMatrix() below;
// see STUDENT comment at the bottom of this function.
void display(boolean isBeingSounded, int channel) {
```

```
// STUDENT (50%) (THESE INSTRUCTIONS ADDRESS NOTES, AND THEN GRAPHICS):
// IF TONIC and SCALEX ARE BOTH > 0, THEN DO THE FOLLOWING:
// Call function mapNoteToChord() that appears at the bottom
// of this tab and store the returned Note [] in a local variable.
```

```

// For each of the non-null Note objects in the returned array:
// Call its Note.display function, passing parameters isBeingSounded
// and channel as they are passed into this function, and passing
// tonic and scaleix as -1 values. Otherwise, each chord Note would
// play *ITS* chord Notes, in an infinite recursion.
// Use pushStyle() before this block of code and popStyle() after to
// save the current stroke color & weight via pushStyle(), and popStyle()
// to restore it.
// Draw a thick line from the current Note to its opposite across the
// center -- from x, y to -x, -y at Z coordinate 0 in the current Hue --
// and do the same for the Note objects returned by mapNoteToChord(),
// using their x and y. I will post a video of chord playing.

// Added by Parson Groundhog Day 2020 for CSC480 Assignment 1
// STUDENT: You must figure out how to use mapNoteToChord().
// WARNINGS: Return value will not be null, but it may have a .length of 0,
// and if non-0, some of its elements may be null because they are out of the
// server's range of Note objects for upper octaves.
// DO NOT MAKE ANY CHANGES TO FUNCTION mapNoteToChord().
// You may add comments if you like, to help you remember how it works.

```

CSC480S20Androidassn1Client main tab:

**** STUDENT NAME:** at the top. **10 point deduction if missing.**

CSC480S20Androidassn1Client LoopState tab:

```

volatile boolean isChord = false ; // STUDENT Parson's code will toggle true/false.
...
// All outgoing OSC messages go through this function.
// Command must be one of:
// "client" "noteon" "noteoff" "clearmine" "clearall"
// See STUDENT comment below.
void sendOSCMMessage(String command, int midiChannel, int pitch, int velocity) {
...
// STUDENT: (15%) If global variable isChord is true, pass the values of
// global variables tonic and scaleix as two additional Integer
// objects before calling oscP5.send(); if global variable isChord is false,
// pass two Integer objects with values of -1 in these two new positions.

```

TURNING IT IN:

Zip your CSC480S20Androidassn1Client/ and CSC480S20MIDIassn1Server/ directories into standard zip archives and turn them in via D2L in order for me to get all of your .pde files from the editor tabs. Use the zipping instructions here to create a zip file. “Here is how to compress your sketch folder into a .zip file (NOT 7z) on our Windows network. Right click the sketch folder.”:

<http://faculty.kutztown.edu/parson/spring2018/CSC120Spring2018.html>

Mac and Linux have command line “zip -r CSC480S20Androidassn1Client.zip CSC480S20Androidassn1Client” command line zippers, as well as zip utilities available from their command GUIs. Zip CSC480S20MIDIassn1Server similarly.

You can use the 7z utility on Windows, but you must choose the “Add to CSC480S20Androidasn1Client.zip” option to create a standard .zip file archive of the sketch folder. If you turn in a .7z or other compressed format that requires me to go fishing for an extractor, I will deduct points. You can substitute your sketch directory name for CSC480S20Androidasn1Client, but this must be in a standard .zip format. Zip CSC480S20MIDIasn1Server similarly. Turn that .zip file in via D2L by the deadline to avoid a 10% per day late penalty.