[SQUEAKING]

[RUSTLING]

[CLICKING]

MICHAEL CUTHBERT: Today, I'm going to be a little bit boring for people watching later and stuff because we're going to mostly be flipping things around. So we're going to use this time for quite a lot of in-class work. Hey, how you doing? You might need a chair there. And try to get things-- getting ready for later, and having videos online for some of the new topics that are coming up. Before we get to-- this is the fifth week of the semester. First, are there any questions so far on problem set four? Adam?

AUDIENCE: So with OPTIC, you would want the two list at the same length?

MICHAEL CUTHBERT: Let me see. Would I want-- no, not necessarily. Let's think through, again, OPTIC. O is-- just shout it out.

AUDIENCE: Octave.

MICHAEL CUTHBERT: Octave. P?

AUDIENCE: Permutation.

MICHAEL CUTHBERT: T?

AUDIENCE: Transposition.

MICHAEL CUTHBERT: Transposition, good. I?

AUDIENCE: Inversion.

MICHAEL CUTHBERT: Inversion. And C?

AUDIENCE: Cardinal.

MICHAEL CUTHBERT: Cardinal. One of those is very important that you test-- with all of them, it's important that you test with different length sets. But what's one of the ones that might sometimes return true if the two pitch sets are equivalent if you have this equivalence class?

AUDIENCE: Cardinality.

MICHAEL CUTHBERT: Cardinality, yeah. So cardinality equivalence, it's kind of an odd thing, isn't it? Because it's not-- it's kind of the backwards one, isn't it? It should be like optic duplication or something because it is specifically not saying that things are equivalent if their cardinality is the same. It is saying two things are equivalent if they're-- what do you want to-- I suppose cardinality is that if duplicates are removed.

Good. So yeah, there will be cases where two pitch sets will have different lengths of things and be equivalent. Oh, one important thing. I'll do the one that I forgot to do. That's on inversion. An inversion is one of these things-- probably like mode or something like that-- we used way too often in music to mean too many related but different things. I'm just going to move that more to the center.

So the two places that we've been talking about inversion are in this-- do you remember the circle from C, C sharp slash D flat. Spelling doesn't matter. D-- am I getting there about right? Yep. E flat, E, F. Now somehow I've-- well, because my circle is more of an egg. So on, and-- sorry, I'm blanking on-- it's been a long time since I've written a circle of notes that is not the circle of fifths on the board.

OK, got that in order. And we've thought about this axis of inversion that you can have. Do you remember that? That can define ways that pitches can invert to each other. We also use inversion. And so this is the primary one that Dimitri Tymoczko is talking about. We also use inversion to mean something which is called tonal inversion. Which is to say-- oh, yeah. We did this in class for a second, right? That this can be inverted to be this.

Even though-- so because here, we go up a step. Here, we go down a step. Here we go up another step. We go down a step, up a step, down a step. Even though the steps are different sizes, right? So F to G is what kind of step? Whole step. Next one. Whole step.

AUDIENCE: Half.

MICHAEL CUTHBERT: Half. And this one?

AUDIENCE: Half.

MICHAEL CUTHBERT: Half, whole, whole. But so in the strictly enharmonic context of inversion, they would not be the same, which is the context that Dimitri Tymoczko is mostly talking about. In the tonal context, they would be the same though, right? But then there's this third part, this third very common use of the word inversion in music that is not what we're talking about in any way on this problem set. John, do you know what that is?

AUDIENCE: Chords inversion.

MICHAEL CUTHBERT: Chord inversion. Chord inversion. Right. So if I have a chord-- let's say, A major-- what is the base if I am in second inversion?

AUDIENCE: E?

MICHAEL CUTHBERT: E, great. So you might have something like that. Now here-- so that's the explanation part. That's the clarification part. And now the critical thinking thing. Chord inversion, in this sense, has to do with what letter of optic?

AUDIENCE: P?

MICHAEL CUTHBERT: P. And there's another one. So P-- we could say that A, C sharp, E is the same as E, C sharp, A, but there's something else we need. Yeah?

AUDIENCE: O?

MICHAEL CUTHBERT: O. So we need that this O is equivalent to this O, that this octave is the same as this octave. I see everybody shivering. It's really cold in here today, isn't it? All right? OK. No, maybe it's just me. You were cold. Other people were cold, and I'm up here.

OK. So O and octave and permutation. If you have an augmented triad, there's an interesting-- here, let's just keep playing. We'll make a simpler augmented triad that you might have thought of before. Then the different inversions have to do with-- you can think of them as O and P. But specifically for the augmented triad, there's another letter that comes into play. Let's get the sound of the augmented triad into our ears.

[PIANO NOTES PLAYING]

What's a way that we can do, let's say, a first inversion of this augmented triad? Yeah? Actually, I'm just calling on somebody. Yeah, go ahead.

AUDIENCE: I?

MICHAEL CUTHBERT: I? Let's see. Yes, because we can invert around this point. And what are the two intervals from here to here?

AUDIENCE: The same.

MICHAEL CUTHBERT: They're the same. Yes, but what are they? Yeah, go ahead.

AUDIENCE: Major third?

MICHAEL CUTHBERT: Major third and?

AUDIENCE: Major third.

MICHAEL CUTHBERT: And major third because you already said the same. Good. Sorry. Sometimes I forget that it was prefaced with something that required me to use the C operator to map major third to major third, major third. Right? Good. So it can be inversion. There's another way you can think of it. Yeah, Jason?

AUDIENCE: Transpositon?

MICHAEL CUTHBERT: Transposition. So if we go up, what do we have to go up? A major third. So C goes up to E. E goes up a major third to G sharp. And then G sharp goes up a major third to?

AUDIENCE: C?

MICHAEL CUTHBERT: First, it goes up a major third to B sharp. And then what other letter are we thinking of?

AUDIENCE: S.

MICHAEL CUTHBERT: S, spelling. So that the-- what do you call it? The B sharp goes to C natural. You can think of all of these different letters and different equivalence classes as quite often, depending on your context, different ways of getting to the same answer. The hard part of the problem set, though, is sometimes figuring out if you're given multiple of these letters together, you're saying that you have the O and P set.

Can you apply O and then P? And is that the same as applying P and then O? Sometimes it will be, sometimes it-- or some of these combinations of letters, the order won't matter. And some of the combinations of letters will open up new possibilities that chaining one and then the other will not be sufficient for. Anybody far enough on the P set to encounter any of those? No? Some of you have. Some people have turned them in. Do you remember any of the ones that were like that?

AUDIENCE: It was, like, three days ago. Yeah. I think octave and permutation didn't matter, I would say.

MICHAEL CUTHBERT: Octave and permutation didn't matter which one you did. I think that's correct. Yeah. Yep. So I should be thinking there's how many combinations of-- what do you call it? There's five choose two. There's five factorial combinations. But we're just-- no, two to the fifth combinations with optic. But a lot of them are never used. You would never use, I don't know, cardinal. What's something you wouldn't use? Inversion without permutation quite often, or something like that. Yeah. Great. Or inversion and permutation without octave equivalents or something. Fantastic.

I want to introduce you to another friend. We're going to have more as class goes on, these little one to two minute guest speakers. And here is somebody who went through this class like you. Let's see.

SOPHIA SUN: Hi, I'm Sophia, and I worked with Mike with Music21 in 2016, 2017. We wrote this paper called Emotion Planning-- called Emotion Painting: Lyric Affect Musical Relationship in a Large Scale Lead-Sheet Corpus. And then the idea is there's this concept called word painting in classical music, where the music reflects what the lyric says. So rise, you will sing like rise. Or fall, you would sing, fall. And then we want to see how prevalent that is in, say, folk or popular music.

We had this corpus of around 1,900 songs of folk music or popular music, and then we're able to analyze it with Music21. What we did is we used the Natural Language Toolkit and analyzed the emotion conveyed by the lyrics, and then looked at the corresponding notes in the music.

We found a bunch of interesting things, like more emotive words that convey an emotion are more likely to fall on strong beat strengths-- like the first beat or the third beat on beat-- whereas words that doesn't convey emotion-- like is, and-- will fall on weaker beat strengths. And also interesting things like words with strong negative emotions like anger, disgust, or fear tend to fall on shorter notes. And that's what we did.

MICHAEL CUTHBERT: So we're going to be moving on to the concept of corpus and corpora just a little bit. And so what do we do-- and statistics because the only reason to look at a huge number of pieces is to come up with some kind of statistical summary or extraction, or to find things that are outliers within it. And even if you're thinking-- oh, yeah, or to do deep learning stuff. And those who have taken artificial intelligence and deep learning classes, you'll know that at its heart, these are all just statistics, right? This is all just a whole bunch of numbers that are being trained and extracted from things.

The corpus I was hoping for us to work with today, I haven't been able to get it working well enough to not crash on my system. And that's my bad for not having this up. But I want to point out to you, we're going to be talking about more-- it's a collection of, I think, 2,000 encoded lieds-- art songs, songs that are sung in concerts and things primarily by women and Black and other minority composers. You'll see that there are some people in here-- Beethoven-- who don't fit in any of those categories. And that was in order to have a comparison set in there.

I don't know, I'll have to ask Mark Gotham, the great researcher who is doing this. I think also, he had really hoped to get 2,000 and had a certain day and found that he already had some Beethoven lying around or something. But it's a really amazing corpus, and I hope that we're going to be able to work with it later in the year. Today being International Women's Day, I thought that would be a particularly good day to work with it. But alas, things don't often hit our deadlines, and we can come back to that later.

The second one that we're going to be working with-- and the reason why you have your laptops out today that we're going to be working with together in class-- is part of this folk music data set. And we're going to be looking at the Irish folk music corpus. There's a lot of Irish folk music corpora online, and we'll be looking at that. Then just to sum up, then we'll be looking at probably the most overlooked at corpus of all time, but one that there's a lot of written about it, and that is Johann Sebastian Bach's *Chorales.* The ones that you've already been working with, a couple of them, we're going to see what happens when we work with all of them.

One of the things I'm going to be asking you to look at is one by next class, if you can, is to start reading an article by Bret Aarden and David Huron, which is-- as you can see from the quality of the scans-- a little bit of an old article. I think it's about 20 years old now. But it was one of the first ones to see what happens when you take musical features and compare them to geographic features. So looking at a density contour map of melodies where most phrases end on the tonic, we can see that there's a huge change from here to here in-- that's Denmark up here. This is Germany, right where the Alps began. And so we can see how geographic features might change how music is.

So I'm going to blank that for a second. I'm going to open up Jupyter Notebook. And we're also going to unlock a few things along the way. So I'm hoping that we'll get through some of these Music21 topics as we go.

So we'll start by importing something that we've already unlocked from Music21 import corpus. And we'll call IRL-- abbreviation for Ireland-- corpus parse. We'll be safe. Essen-- E-S-S-E-N-- folk song slash IRL. See how close I can get this to me so I'm still looking at you. Oops. Put this topic five, 2023.

Great. So everybody, give me a quick thumbs up if we're at this place. Awesome. Super. OK. One of the things you can do, anything that you get from the corpus, just in case you ever need to look at it-- oops, sorry, I have an old thing-- is look at where it is on your hard drive. OK, now somebody can hack my hard drive, and so on.

What kind of thing would you expect IRL to be? What kind of object? An int. Who votes for int? Float? String? You think it's string? No. You don't think--

AUDIENCE: No, I was going to say something else.

MICHAEL CUTHBERT: What usually are we working with in Music21.

AUDIENCE: Stream.

MICHAEL CUTHBERT: Stream. Great. What kind of stream do we think this--

AUDIENCE: Score.

MICHAEL CUTHBERT: Score. Score would be most logical, right? In this case, we're working-- this our first time working with something bigger than a score, something called an opus. So a collection of scores. And so what we can do is we can look at how many scores are in here. So this is a pretty small corpus, but big enough to start doing some statistics on.

So we can do irl.scores0.show. And good deterministic. It's still the same one that I thought it would be. Let's see this. One of the things you'll find if you're really interested in being efficient is instead of doing what do you call it? Square bracket zero? On Music21 scores, once you've done all your filtering, you can do dot first and dot last. And all they are is O of one-- or that they stop finding all the scores once they found the first one. So it's a little bit-- if you're into algorithmic optimization, it can sometimes be quite a bit faster for very long notes.

OK. So what I want to know is-- this is interesting. It's in 3/4. Pick up. Good. It does some of the things that we expect tonal pieces from folk era to do. If you begin with-- what is that-- a dotted eighth, a dotted quarter missing at the beginning or added to the pickup measure at the beginning, you expect that at the very end, you also have that. So the pickup measure and the ending measure usually represent one. Is everything OK? Yeah, go ahead.

AUDIENCE: Could you go-- scroll up, please?

MICHAEL CUTHBERT: Sure, of course. Always glad to. So did we get it? Is that--

AUDIENCE: That's good. Yeah, the piece looks a little bit different on mine.

MICHAEL CUTHBERT: The piece looks a little bit different on yours? Hugely, terribly different? No, just a little bit different? OK. Yeah. I mean, depending on your version, here's what it looked like a couple of years ago. It's also quite a bit smaller back then and things, but OK.

So it's in 3/4. Is that common for Irish folk music? What do you think? Who has heard Irish musicians play Irish music? Not very many. Keep your hand up if you're also over 21. No, just kidding. Because they do usually play-- one of the places you can hear quite often Irish folk musicians play in Boston is in pubs and things.

So of people who have heard or people who think they might know this, what's a time signature that you think would be common? Adam?

AUDIENCE: 2/4.

MICHAEL CUTHBERT: 2/4. [? Sruthi? ?]

AUDIENCE: 6/8?

MICHAEL CUTHBERT: 6/8. John? 6/8 also?

AUDIENCE: I was going to say 6/8.

MICHAEL CUTHBERT: 6/8? Well, this is one of those cool questions where if you have the same answer, somebody else would still want to hear it because that starts building up consensus. Somebody give me a time signature you don't expect to be the highest-- the most commonly used time signature. Yeah, Jason?

AUDIENCE: 7/4.

MICHAEL CUTHBERT: 7/4. Good, good. OK, so we have some hypotheses. What we're going to do is try to figure out what's the most commonly used time signature in this Irish folk thing. And always start with a hypothesis because the number of times I've seen people go ahead and publish things where they don't have a hypothesis, and then they don't realize-- like, oh, I printed out the measure number divided by the number of notes, not the time signature. And therefore, 137 over 14,000 probably wasn't the most common time signature.

So let's start. This was a question that came up at some point. Can you use-- are anything in the normal Python library unlocked in this class? Yes. So let's import the counter. Who has seen counter? Not everybody. OK, some people have. Counter is just-- it's a very cool little-- it's just a dictionary. But it starts off at zero when something's not there. So you don't have to say-- when you're adding things up, you can just start from that. So you can just start adding one to things. So I kind of like that.

So let's say, for song in the Irish corpus, we can do-- we're going to want to get the time signature. So I want you to song dot something. Oops. Pull that up a little bit. And fill out the rest. Or maybe song-- maybe we don't even need the dot something. I don't know, song is probably the next thing.

And we did put IRL scores first up there, so you know he can test this on one. Let's say we're only-- for simplicity's sake, we're only going to be counting the first time signature in each song. And that most of the songs in this collection-- I think all of them are monophonic, so you're only going to have one part. What are some words you're typing?

AUDIENCE: Get element by class.

MICHAEL CUTHBERT: Good. Get element by class, or get-- there's one little type. Get elements, plural. Yeah. So unfortunately, there isn't a first element thing. Great. Anybody else having other words that you're typing that you'd like everyone else to know? No? Unless I tell you otherwise, like the midterm or something, you're generally allowed to talk to people around you. So what's a word that you've typed?

AUDIENCE: I'm going to just select the [INAUDIBLE].

MICHAEL CUTHBERT: OK, cool. Yeah, Misha? What's--

AUDIENCE: I guess we can, yeah, use meter.TimeSignature.

MICHAEL CUTHBERT: OK, so meter.TimeSignature. Yeah, so you'll probably want to import meter. Sorry if I didn't do that. You can import anything that's unlocked always. So from Music21, import meter. And what's the word that you said? Meter.TimeSignature. That one might be helpful. Anybody else have any words that you've been typing?

I'm going to type a word. My word is recurse. Anybody type that word? No? Or flatten. Anybody type that word? I'm only asking for this line. So I'm just trying to figure out who has gotten a time signature out? OK, so some people have. Vincent, what did you type to get your time?

AUDIENCE: Oh, I made a get suggestion. So recurse.getElementsbyClass.

MICHAEL CUTHBERT: OK, good. Recurse.getElementsbyClass. And anybody, help. What's the next thing I'm going to type? Meter.TimeSignature. Great. Anything else that I should be typing on that? Is this correct? Will TS now be a meter.TimeSignature?

OK, let's try this on one song since I don't know if we have this at all. So we'll call-- we'll go back to the irl.scores.first. And what do we-- what was that name of that song? "A Lady Fair". We'll call it lady equals irl.scores.first. Now we'll try this. Lady.recurse.getElementsbyClass meter.TimeSignature. Who votes this is going to get me a time signature? Who votes not? OK.

AUDIENCE: Not sure.

MICHAEL CUTHBERT: Not sure. But not a time signature. You had your hand up on not a time signature? Good. Anybody else? Vanessa, you look like you're--

AUDIENCE: [INAUDIBLE].

MICHAEL CUTHBERT: It's a what?

AUDIENCE: I think it's like a getter function [INAUDIBLE].

MICHAEL CUTHBERT: Yeah, something that will get what?

AUDIENCE: [INAUDIBLE]. From the song as opposed to the song element itself.

MICHAEL CUTHBERT: Oops. Do I not have-- oh, sorry. Yes, that won't return anything if I don't put these, brackets right? As I was trying to do it on a function. Oh, this comes up, by the way, in problem sets sometimes in answers. The order matters, right? I hadn't run the Music21 import meter yet.

Just something to keep in mind. Always make sure when you're done-- when you think your problem set is done, hit Restart and Run All, and see if it gets to the end because you might have been solving a problem along the way and then have-- oh, in order to solve question two, I imported something from question three, but then it's not ready when I run it.

So yeah, so now that we have this, what is something called a recursive iterator? So it's not a time signature. How do I get the time signature from this if we have this? Anybody? Hannah?

AUDIENCE: You can index zero.

MICHAEL CUTHBERT: Index zero. Perfect. It's kind of like what we did here, indexing zero iterator over scores. But now we have something that's indexing over the time signature. So we should be able to do this with-- if I remember my parentheses, I do get used to an IDE that fills in my parentheses for me, which I'm not using. And I can do index zero or I can do dot first, either one.

This is a particular case where dot first will save you a lot of time because you're pretty sure the time signature should be at the beginning of the score. So once you hit one, stop. Don't go through the whole rest of the score looking for more time signatures if you only want the first one.

Good. So we have the time signature. Now what we can do-- this is something we haven't seen. The ratio of a time signature-- the 3/4 sign-- is in its ratio string. There's also a numerator and denominator, but this is the one that's pretty helpful. And then meter counter ratio, what should I be doing now if I want to count them?

AUDIENCE: Plus equals one?

MICHAEL CUTHBERT: Plus equals one. Good. And then at the very end, let's see our meter counter. So we had hypotheses of 3/4 and 6/8. I put my money on 6/8. And whoops, what did I forget? Oh, irl.scores. Thank you. Irl.scores because the first thing there was metadata. So what's our winner?

AUDIENCE: 3/4.

MICHAEL CUTHBERT: 3/4. Our runner up?

AUDIENCE: 4/4.

MICHAEL CUTHBERT: 4/4 And then 6/8. At least 6/8 was on-- those of us who were 6/8, the 6/8 crew, we did OK on there. But yeah, I might have the difference between this being sung music versus fiddle music, I don't know. We can try some of the other things in there. By the way, with a counter, you can also do things like meter counter.most common, three most common, and see what they are in that order. Cool.

So that's one of the things we can do with this corpus. Let's jump over to the Bach corpus for now for a little bit. Are there questions about this? I mean, that's just the tiniest little thing. We can figure out all the questions that we had before with it.

Oh, here. Let's do this. Let's do this type of experiment that Sophia Son was talking about with position on the beat or whatever. Let's look at where notes with accidentals appear. Are they most commonly in this little passing position in the middle of the bar? Or are they more often contradicting right at the downbeat?

Hypothesis? Who votes for accidentals more in the middle of the bar? Good. 7, 8-- and for on the downbeat? Good. Who's not participating? That didn't add up to-- let's one more time. In the middle of the bar? Or downbeat? A couple of downbeat. Good. I don't know what the answer to this is. Always just be thinking about these things.

So let's do the same type of thing. We'll create another counter. And we'll unlock another little thing for you. So we'll call this accidental-- we're not counting accidentals, so maybe just accidental positions. But maybe we don't even want to do a counter. Maybe let's just do a list so we can run averages on it.

Now, we'll be fair to compare-- well, I don't know-- where in the measure it is. Maybe we want to just say we'll just look at the 21 3/4 one so that all of our measures are the same length, except for, hopefully, the pickup measure and the end measure are rounding errors, but maybe not. You know, because they don't have the same length.

So what we can do is, again, copy and paste. Probably, if we're doing this more, we can get this. And instead, we want to look at every note. So there's something called notes on this. Or actually, we'll do even better. We haven't imported note yet, right? From Music21, import note.

Because of a bad decision I made a while back, dot notes also will get you chords if there are any chords in here. And chords, we will get to soon. They have to be dealt with separately. So we want to say .recurse.getElementsbyClassnote.Note. But there's another way to do that. Anyone remember what's the shortcut for recurse? Get elements by class? Yeah, go ahead.

AUDIENCE: You use the bracket and the class name.

MICHAEL CUTHBERT: Good. So we'll put the bracket for n in song note.Note. So that will get us every note there. And how do we filter out ones without accidentals? So try that yourself. Try to figure out-- pretty sure that this is right for now. Yep. We've got a whole bunch of notes. We want to filter out ones without accidentals. Oh, why do we have a five space thing? That bothers me. Filter out no accidentals.

We're going to have some-- we might not get to the Bach example because I came up with this one, and there's a neat little twist that I had forgotten about. Go ahead and interrupt the class, and ask that for everybody in a loud voice.

AUDIENCE: What do people think about accidentals in the key signature? Are we counting those as accidentals?

AUDIENCE: No.

MICHAEL CUTHBERT: OK, good. So we're going to have to do some-- that was the wrinkle that I thought of. One of the ways you could do is you could say, oh shoot. OK. We're just going to look at ones with C major so we can do it. But presumably, sometimes you can filter down your corpus to get rid of some corner cases that might be a little bit too hard to deal with, at least in the first run. But getting rid of everything but C major is probably not going to leave us with enough of a corpus.

So let's temporarily unlock a second module from Music21 import key. And first, let's get the first key signature for each thing so we can do-- how do we get the first key signature in the piece? Let's assume that for now, that it doesn't switch key signatures.

AUDIENCE: Song.recurse parentheses.

MICHAEL CUTHBERT: Song.recurse parentheses.getElementsbyClass. Or what's a simpler way?

AUDIENCE: Square bracket.

MICHAEL CUTHBERT: Square bracket key dot-- I'll tell you it's key signature-- dot first. So first, let's filter out everything-- so now, let me make a little introduction. Man, I wish I could edit at this part. So let's go back. Always work through one score.

So we have the Lady score, and we'll get ks equals lady-- was it "My Lady Fair"? Lady key-- I didn't import key yet from Music21. Let's pause for a little bit and help me get caught up so that anybody watching this later can get caught up. Can you all come with me for a second and eyes for a second? Thanks.

Can somebody tell me what I should be putting here? I think we're mostly at about this line together. So who have I not heard from very much? Angelica?

AUDIENCE: OK, so one doesn't work. But I tried to say if n.pitch.accidental, and if the key signature.accidental by step, and then n.name.

MICHAEL CUTHBERT: And-- sorry, and ks.accidental by step, n.name? Good. That's a pretty good start. I might change one thing on it. Anybody else? Anyone have a suggestion of one thing to change on that? So we're looking to see there is an accidental and there is an accidental-- yeah.

AUDIENCE: I would change name to step.

MICHAEL CUTHBERT: Change name to step. So the difference between name and step, accidental by step expects a step. C sharp's name is C sharp. C sharp step is-- shout it out. Nobody knows. Total mystery. C. Good. Step-- I know you knew that. But just keep that going. Good.

So what's this logic? If the note pitch has an accidental and there is an accidental there, then what? Anybody who feels confident that they can code the next line of this, even if you didn't have this? Or even just logically, what do we want to do now?

AUDIENCE: If they're the same--

MICHAEL CUTHBERT: Uh huh.

AUDIENCE: --we don't add one.

MICHAEL CUTHBERT: OK. If they're the same, we don't add one. Good. If n.pitch.accidental and equals ks.accidental by step, n.step-- what is it? Something plus equals zero. We won't do anything. Nil, we can pass.

Since we want to do something at this point, how can we flip those? How can we flip it so we can add this? Don't worry, we'll improve some of the redundancy in a second. If it's like this, we don't want to do anything. If it's not like this, what do we want to do? Add one. So how can we save ourselves? I suppose we can put else something plus equals one. But--

AUDIENCE: You could do a not.

MICHAEL CUTHBERT: I can do a--

AUDIENCE: A not.

MICHAEL CUTHBERT: A not. Not equal. If these are not equal, then we can do that. So that could work. What's another thing I can probably save some redundancy? Let's store these things at acc equals n.pitch.accidental. ks acc equals this. When you find yourself writing something twice in two lines, you will probably continue to write it.

And so we can do this. We might even be able to reduce this further because Python will allow you to compare none not equal to something else, but good. Does that cover all the cases? We have an accidental and we have the accidental in the key signature. Angelica, by the way, that was great. Thank you for getting us going on the right path.

And we have a key signature accidental, and so we have an accident on the note. We have a key signature accidental. And the two are not equal to each other, so therefore, we're going to add one to something. We'll figure out what we're going to add something, but we care about that note, right? Any other notes we care about? Yeah?

AUDIENCE: What about notes where the key signature has them sharped or flatted, But, then we natural them?

MICHAEL CUTHBERT: OK, good. Good. And the problem with natural is what?

AUDIENCE: Technically, it's not an accidental.

MICHAEL CUTHBERT: Technically, it's not an accidental. This is a thing that I did to speed up so that there would be one fewer object creation back in 2004 and 2005, when object creation was still something that was slow in Python. Do I regret not putting an accidental of natural on every single note now? Yes, I do regret that.

Here's a trick that I've started to do a lot since I don't really care. We'll import pitch for a second. Music21 import pitch, and we'll just make one natural. And I'll put it in all caps because that means to me that I'm going to try as much as possible never to mutate this.

So one of the things we could do now is since none returns falsy-- so therefore, we continue in the or-- we can say the accidental is this one or natural, and this one is or natural. Now we are always comparing apples to apples. So this is just a little programming trick. And in case you really care about the runtime, I only had to create one extra accidental object. I think that should work. I see some confusing. Yeah?

AUDIENCE: Why can't we just have the not equal sign? Like if there's a non-object and it's something that's not, then will it not compare?

MICHAEL CUTHBERT: Sorry. Can you say a little bit louder, both for me and others?

AUDIENCE: Yeah. Once we have the accidental in the key signature--

MICHAEL CUTHBERT: Yep.

AUDIENCE: --why do we need to have the or natural? If we're comparing the not equals in the next line, will that not work with non--

MICHAEL CUTHBERT: Ah. Yeah, so one of the reasons might be-- yeah, that's true. We probably don't. Wait, no, no. Then we get to Matthew's case. What case does this not cover?

AUDIENCE: I don't see a case that this doesn't cover.

MICHAEL CUTHBERT: Great. So here's the case that doesn't cover. ks, again, is our D major thing. Accidental by step of G is none, right? n equals note.Note G sharp-- G sharp four. Great. Next n equals note.Note G four. When I generate this, it's going to get an accidental because it needs something. So you're going to end up with next n.pitch.accidental-- oops. dot. pitch.accidental equals pitch.accidental natural because it will actually have a natural sign.

And then this line up here, next n.pitch.accidental. Does it equal ks accidental by step G? Oops, what did I forget? Didn't I import pitch, or did I-- I never ran that. I typed that, but I forgot to run it. False because this is a natural, an explicit accidental of natural, and this is a none. Not a nun like the ones that wear the habits and go around, but N-O-N-E.

So that's the particular case. Now, one of the things we never talked about is how do accidentals compare? First, we want to make sure does one accidental flat equals another accidental flat? Yes, they do. So that's the reason why I put in these or-- where'd they go-- or natural, so that I'm always comparing apples to apples. Great idea. Yeah, Jack?

AUDIENCE: I'm confused how the example-- didn't we want it to be false?

MICHAEL CUTHBERT: Didn't we want it to be false?

AUDIENCE: In the extra thing you just wrote below.

MICHAEL CUTHBERT: In the extra thing?

AUDIENCE: Yeah, that.

MICHAEL CUTHBERT: Right. We want this one to be false because we're in D major, right? Accidental on G is none, right? So this is none, and this is a natural. But the note is still going to be a G natural. It's just one of them is represented with no accidental, and the other is represented with the accidental of natural. So--

AUDIENCE: Oh, you're saying having the natural sign [INAUDIBLE]?

MICHAEL CUTHBERT: So by putting the or natural-- so I've defined this now. Acc equals next n's pitch.accidental or natural. What's acc now? It's a natural. ks-- key signature's accidental equals ks accidental by step. G or natural ks accidental is now also a natural. Acc equals ks accidental is now true.

The notion that you don't care about the difference between a G that has no accidental and a G that has an accidental of natural, right? In this case, we do not. So does that start to clarify things?

OK. Here are some notes. Let's always go back to actual music. That's how this-- that's why we're confused. We haven't looked at some actual music in a little while. So we're in D major. Here, we have two sharps. Let's ignore the time signature for now. And now we have something like--

[MARKER SCRIBBLING]

How many notes here contradict the key signature if we're looking at this?

AUDIENCE: One.

MICHAEL CUTHBERT: One. Which one? That G sharp. But this has a-- the accidental on this note is probably none. The accidental on this note is almost certainly natural. And the key signatures-- give me another color. The ks of-- it would go right here-- the accidental by step for G is none. So in this case, these two would compare equal, but these two would not.

Not a very good instructor of software engineering today. Sorry about that. But I'll try to get better. Any other questions? We have about four or five minutes. I'd love to try to get a little bit further in this, so I'm going to jump back.

Oh, we have something I never did. Let's just make it-- in the interest of time, we'll create a counter and just keep track of downbeat versus non downbeat. So what we'll say is if the accidental is not the same as the key signature accidental, then it's an interesting note. So then we want to know if it's on the downbeat or off the downbeat. What do y'all use to figure out if something was on the downbeat? What'd you say?

AUDIENCE: Offset.

MICHAEL CUTHBERT: Offset, right. So if the note's offset-- and here is where recurse and flat are going to be very different, right? Because recurse's offset is with respect to what? Vincent?

AUDIENCE: Measure.

MICHAEL CUTHBERT: The measure. And flatten's offset is with respect to what, Misha?

AUDIENCE: The start of the--

MICHAEL CUTHBERT: The start of the--

AUDIENCE: The score.

MICHAEL CUTHBERT: Yep. Start of the piece, start of the score. Start of, yep, of whatever it is. So this one, we want this here. If the offset is 1.0, we can say accidental positions. I'm going to do that for, I don't know, downbeat. Plus equals one else accidental. Did I somehow not type properly? There we go. Dental positions offbeat plus equals one.

Oh, the other thing that we might want to do is eventually normalize this for how many notes are on the downbeat versus how many notes are on the offbeat. You all know about normalization, right? You want to figure out how much more likely is it. No, but I can look at the time and see we're not going to be able to do all that, so yeah. So let's-- I think this might work. So let's look at the accidental positions. And I'm only looking at 3/4 ones. You might be-- anybody get something like that? Hannah, you had some results?

AUDIENCE: Well, I did it a little differently, but I had a slightly different output.

MICHAEL CUTHBERT: Yep. But It's kind of great when we have two different people doing different ways. So you have a slightly different output. Did you find something like this in your results or--

AUDIENCE: I took the average--

MICHAEL CUTHBERT: The average?

AUDIENCE: --of where it appears in the measure, I guess. So I guess the rest of what I got was closer to the downbeat, I think.

MICHAEL CUTHBERT: It was a little bit closer to the downbeat? Great.

AUDIENCE: I do know. It's closer to the middle.

MICHAEL CUTHBERT: Closer to the middle? Fantastic. I'll give you one little thing, as we do it, that might be interesting going forward. We're not going to have time to implement this, but I will be unlocking this for you right now for n in-- let's just look at lady note.Note. Print n and ends beat strength.

And we'll look at this a little bit. And this tells you-- here, do we have the-- I think somewhere I have the score. Again, sorry, I wish I could split this here. So the first D is on the downbeat, and then we're going to go up to F natural and the C natural. And when we look at this-- oops-- later, we can see we have the three pickup notes. And then we have something on the downbeat. And then we have a bunch of offbeat things.

Yep. Beat strength is just a really, really hacky thing I threw together in probably about the same amount of time as this thing here that tries to measure as a number between epsilon and one. There is no beat strength of zero, but how strong the position of the note within its measure is.

I cannot believe how many times I've seen it referenced in papers now. I've used it in papers and things like that, and now the world knows that this was a terrible hack that could probably be done much, much, much better, as you will realize when you do beat strengths on triplets and things because you can see I'm doing something about the log2 of the proportion through the measure, and therefore triplets won't work as well or anything. So I think we do something separate for 6/8 so it works.

But this could be used to give you a little bit more of a sense of the kind of problem that Sophia Son was talking about in the video at the beginning of the class, are certain things happen more on strong places versus others. OK. Can you close what you're working on for just a second? And that's also the cue that I'm going to let you go just so I can have some eyes and things.

So the next problem set-- you're working on this one, but just don't pick up for one second-- is going to be trying to find yourself in interesting corpus. Whether or not you decide to end working on a corpus study at the end of the semester or not, it's probably good to know that we do not want to just continue working with the Bach *Chorales,* and the Irish folk songs, and the Chinese folk songs-- these three data sets that are everywhere-- for the rest of our lives. So we want to have that out.

You're going to be asked to do that and to write up what you find about something, and you're going to be asked to do it in teams of two. Your only condition is that you cannot be on a team with somebody that you did the last group problem set with. So find somebody who is not in that problem set, and do that. And I wanted to give you some extra time for that. Great. It's a little bit more on the Sofia Sun and my article online that you can watch today in the Canvas or later, and thank you.