Ian Cropper
CS 330-2
January 5, 2012
Music and Programming Languages

---

Musical notation and programming languages, though not intuitively similar, are inherently related because both creations come that same part in our being. With something so emotional and expressive as music, and something else so technical and formulated as a programming language, it might be hard to believe that these two things are as related as they are. With deeper insight; however, we can see that these two things carry some very fundamental foundational pieces that related the two.

True musical notation was only first recognized around 900 A.D. Up to that point, music indicated only what pitch to play/sing; there was no way of describing rhythm. As mankind traversed though time, music was eventually given a time signature, and rhythm, and gradually a grammar for how music should be played and notated. By the time Mozart arrived, he could write down a song from his mind and have someone else play it just as he imagined it. Clearly, time signatures are important (try humming Mary Had a Little Lamb to some other song's beat … it's barely the same song!) which goes to say that having that grammar is also very important. Without it, music would be entirely subjective and impossible to replicate.

Let's shift now to programming languages

Any language be it Hindi, English, or Java, comes with a set of rules that define the syntax for a language. When those rules are followed, we accomplish our goals. When they're disobeyed, we end of with an incoherent mess that no one (let alone a computer) can follow. When creating (composing) or implementing (playing) a language, (musical piece) we use the set standard and rules of that language so that by the time we are finished, we actually have something that we can use. In Java, (and many other languages) for example, an object's elements are accessed by the syntax `object.element`. Without this grammar, mass coding languages would have been impossible to develop because they would be different for each user. Just as with a Gregorian chant, the pitches might have been on, but there was no time notation (ever wonder why Gregorian chants have such long, inconsistent notes? Don't blame them, they didn't know any better.). Having that grammar allows other users to use your system properly, be it a sonata or a programming language.

Music and programming languages share a common thread in us because they both take an immense amount of creativity, but also an immense amount of structure and discipline. Put those two things together; however, and you have a beautiful, workable system.