It Is an interesting fact that Python is an interpreter that was a written in the C coding language. However, Python is the most popular programming language since it is good for people who are learning to code. It is especially popular for intro programmers in embedded systems. Python also has a leg up when it comes to importing modules fast and easy. When it comes to this current assignment 2 I found it worlds easier to code since creating an array in python is very intuitive versus it's C/C++ counterparts. It was very easy to initiate the vector and its size. Manipulating the vector, and calling it's values were very simple due to the ease of access.

C/C++ cannot be contested with when it comes to fast compact code. C/C++ is used by most of the CS community for embedded systems. The biggest difference is that Python is an interpreter which is very slow because interpreters read each code line by line whereas C/C++ is compiled and ran which runs more efficiently, and can be optimized well. Thus, in certain situations Python will have slower runtimes and higher energy consumption. However, when it came to this assignment 2 I was not a fan of C++. When it came to initiating the array for the quicksort algorithm there were a lot of errors that I encountered (this is partly due to the fact that I finished the assignment in C++ before the assignment page was updated with helper code). Since C++ data needs to be cleared out, whereas Python does not require you to do so, generating a random number, having it be different every iteration, the time header (time.h) had to be included to access "srand" and "rand" solving all the basic problems that were encountered. Besides the random numbers I personally did not have any other issues with C++.