Computer science contains all of the theories and studies for design and use of computers. The science may include such considerations such as the efficiency and structures for computer algorithms. Software engineering, rather literally, is the engineering of software. It tends to involve the actual creation of software, applying topics of computer science to create software for various purposes. Information technology, further builds upon that. It tends to involve the use of computers to interact with data in various ways. It could include the use of databases, which are used to collect, store, and provide information. Various pieces of software could be built to compile the stored data into more useful information.

Theoretical computer science is a more abstract level, looking more at properties, structures, and abilities of algorithms than at the actual code itself. Here you could consider things like Turing machines, lambda calculus, or the Halting Problem. These problems are less about actually building algorithms, and more of what is needed to do so, or what can and can not be done. (Although, you could actually create an algorithm for use on a Turing machine, or lambda calculus processor.) Other questions under this are include "P vs NP," a long-standing question of whether problems whose answers can be tested in polynomial time (NP), must therefore be solvable in polynomial time(P).

Computer Systems goes more into the structure of computers, either physical or digital. This could include the actual logic inside a computer, that makes it function, or the structure and organization of algorithms running on a computer. Here you may look at organizing a computer for parallel computation, considering how best to design a computer, or computer component to meet certain specifications, or perhaps finding efficient ways for it to communicate with another computer.

Computer Applications goes into the use of computers for various applications. This could include the use of computers to store and show images, audio, or other information digitally, the use of computers to simulate physical phenomena, or to create algorithms to solve a problem, but more fluidly, and possibly learning all the details on its own. This is where computers actually become useful on a larger scale, being able to receive, process, and send information for various purposes.

I like both Theoretical Computer Science, and Computer Applications. The first, because I like math, and proofs, and this region is closer to those than to actual computers. The second, because this is where computers actually derive their usefulness. Here is where you could automate various processes, or find ways to let a computer learn on its own. I am particularly curious about the outcomes of machine learning, and how that will apply.