Inquiry 1

Algorithm analysis, algorithm correctness, and distributed algorithms are three important areas of Computer Science. Algorithm analysis provides a common terminology and means for determining performance aspects of a given algorithm independent from programming languages or computer architectures. Algorithm correctness is a growing field that aims to formally prove that a program or algorithm is bug free with the same definitive nature as a mathematical theorem proof. Distributed algorithms are becoming an increasingly important field of study because they take full advantage of common server architectures and provide the theoretical underlying of most Internet protocols.

Most intellectual fields agree on a set of terminology that condenses heavy concepts into words or symbols. Computer Science is no different, and algorithm analysis provides the common underpinning for determining the performance of an algorithm through asymptotic notations. The most common asymptotic notation is Big O notation. The Big O notation shares the upper bound performance characteristic of an algorithm. In other words, it tells us the worst case of an algorithm's performance given a number of inputs (n). The common, but by no means comprehensive, orders of Big O notation, in order from most performant to least, are constant, O(1), logarithmic, O(log n), linear, O(n), quadratic, O(n^2), and polynomial, O(n^c).

Algorithm correctness is an attempt to bridge the world of software and hardware development into the world of pure mathematics. It attempts to prove a program or algorithm is correct through a rigorously defined specification and software that implements the specification to test the algorithm. It's commonly used in the development of microprocessors and highly mission critical software such as the Mars Rover by NASA. Despite the field of algorithm correctness having such an attractive promise of bug-free software, it doesn't see much application in everyday software development where "mostly correct" is often good enough.

Distributed algorithms are algorithms that take advantage of having multiple processors available to perform computations. This is often done in a divide and conquer approach where each actor can contribute to a final solution. The rise of cloud computing has elevated the interest in distributed algorithms due to the ease of provisioning another processor to speed up computation. Distributed algorithms also play a critical role in the Internet by providing algorithms for different Internet and routing protocols.

Algorithm analysis, algorithm correctness, and distributed algorithms are three topics any budding computer scientist would do well to know. Algorithm analysis is a classic subject that will always have great import in communicating efficiently in the field of computer science. Algorithm correctness and distributed algorithms are going to become increasingly important over time as the increase in computing power compels us to take advantage of it fully; something both subjects contribute to in unique ways.

Inquiry 2

There are many hot computer science topics that apply to a mobile developer that creates applications on native platforms. Some of the most salient topics are security of computer systems, programming languages, multimedia, human-computer interactions, artificial intelligence, and abundant-data applications algorithms, and architectures.

Mobile devices are becoming very powerful, and, as a consequence, mobile development is increasingly evolving from toy or companion apps to full-featured, robust computing applications equivalent or greater than traditional desktop and web counter-parts. This is evident in a trend we've noticed in our employment as a creator of apps for people who come to us with requests: many people are starting mobile-first and adding web or desktop apps later. This trend has increased steadily over the years and will continue to do so as the proliferation of mobile devices all but guarantees that the largest consumer market can only be tapped through mobile platforms.

The data sets our apps generate grow more abundant as the apps become more robust. The more robust an app is, the greater the expectation of a user for it to be smart and learn from the data sets the app is generating. Abundant-data algorithms and architectures are becoming more entwined with day-to-day mobile development. The use of a research-based approach towards abundant data can help an app tailor itself to a specific user's usage patterns.

Artificial intelligence is another way to make an app smarter. It can go hand and hand with abundant data sets through machine learning to create a more compelling experience. A common problem that a research-based approach to artificial intelligence is used to solve is some form of recommendation system. It could also be used for more novel approaches, such as optimizing an algorithm based on real-life results. For example, one could make an app that detects concussions and tunes its concussion detection algorithm based on its accuracy compared to a professional neurologist.

Human-computer interaction is another field that is pertinent to mobile development. All software benefits from knowledge in the HCI field, but mobile apps have a wider range of senses at their disposal to allow for a more nuanced approach to HCI. A research-based approach to allowing a user to customize an app's experience for their preferences and use cases is one problem that could be solved through applications of human-computer interaction.

Security of computer systems and support for digital democracy are also very important to mobile computing platforms. As an average user places an increasing amount of data, from pictures of their kids to their bank accounts, we must place great import on computer security. A research-based approach to digital democracy could be powered by voting platforms ran on mobile devices. Research in the security field and cryptography are particularly important and applicable to making sure all the sensitive information we store on mobile devices is resistant to eavesdropping and intrusion.

Programming languages are another important topic in mobile development. Mobile development has been the impetus for new developments in programming languages, as seen in Apple's Swift programming language, and the proliferation of novel languages running on the Android Runtime (whose progenitor is Java's JVM). It is important for mobile applications to have access to fast programming languages due to their lag behind desktop and server computing power. However, it is also important for these languages to be productive and easy to use to enable development to keep up the pace with demand. A research-based approach to developing new programming languages that takes full advantage of the mobile environment is going to be a continual problem needing to be solved over the next decade or more.

Multimedia is turning into an increasingly applicable topic on mobile platforms. Whether it's watching videos on the device, creating them, or delivering them to a TV through technologies like AirPlay or Chromecast, mobile devices are becoming the epicenter of multimedia creation and consumption. The touch based interface intrinsic to mobile development has created a rise in content editing, a practice traditionally limited to traditional computers, as the devices have grown in power. This can be seen in high end devices such as the Surface tablet, or iPad Pro. Mobile devices have been a driving force in developing new standards such as HTTP Live Streaming that allow a device to dynamically adjust video quality based on the bandwidth available. A research-based approach to multimedia could improve live streaming's responsiveness and quality. It could also help solve the problem of being able to participate in multimedia rendering and editing anywhere without having to carry around a bulky laptop or desktop computer.

Mobile development is an exciting field that grows each year as people increasingly prefer to interact with a computing device wherever they happen to be. This has traditionally been limited by the computing power of mobile devices, but we have finally entered an era where our phones can match the performance of a low-end laptop. It will be exciting to see how all the latest advances in computer science will be tuned and applied to mobile platforms in areas such as security of computer systems, programming languages, multimedia, human-computer interactions, artificial intelligence, and abundant-data applications algorithms, and architectures.