ESSAY

Introduction

My south-Indian parents, who moved to the United States from Andhra Pradesh, provided me a colourful cultural experience while growing up. My childhood was filled with a unique blend of visits to Buddhist and Hindu temples, eating pizza and curries, and listening to a mix of South Asian, American and Mexican music. Those diverse cultural experiences made me the Man that I am today and I share my one-of-a-kind background through my professional and social work.

The aptitude for the enhancement of this organism and its easy to access the common man lead to a violent shattering of a multi kinds of ProgrammingLanguages, Hardware Architectures, Compiler Designs, Operating Systems, DataBases, designing a web pageall the while reducing the size of a daily essential kinds of machine. While the increased sophistication of mathematical, formal expressions and automata theory gave the machine it's Mathematical underpinning, the advent of the *integrated circuit* gavethese intangible algorithmic constructs the framework to operate in. The combination of these two enitity's (one abstract the other concrete) paved way for this organism to go from Charles Babbage's steam run *Difference Engine* to the Turing Complete *Analytical Engine* to the first *Super Computer* to the first *Personal Computer*, *Lap-top* and finally, an exponentially more powerful *cell-phone* fitting easily in the palm of our hands. And maybe a likely eventuality: a chip in our heads controlling everything we do in our lives!

The incredibly fascinating and really spectacular facet of this ever growing, ever burgeoning entity, distinguishing it from other *Sciences* is that it is a pure invention. An algorithm or procedure exists in the abstract; it is devised by humans to address real time problems. The problem may be concrete but its solution is not. The robustness, veracity and flexibility of these procedures or algorithms are done not by the traditional scientific method (Observation-Hypothesis-Experimentation-Result), but by mathematical reasoning, logic and testing, making *Computer Science* closer to mathematics than other natural sciences like Chemistry, Biology, Physics or the Social Sciences like History, Geography and the various spin-offs of these rather conventional scientific fields . Computer Science can be viewed under the same evolutionary light as Mathematics, for Math also is an instrument in the *art* of Problem Solving.

Human Computer Interaction

With Computers becoming a standard in every household, it has become imperative for them to be user friendly. Gone are the days when Complex Instructions were written to control Jacques looms. Nowadays, people are just a click away from doing tasks that would have been unfathomable back in the old days. Not all of these people are Computer programmers; most of them are end users using these systems to maybe check an email, update facebook, see a YouTube video, perfectly oblivious (well, maybe they are aware, just sounds better when I say that) to the mind boggling plethora of languages and code that are put to work by

the programmers at the *back end* so that user experience in their work at the *front end* is engaging and purposeful. In both these cases, the individual, be it the regular user or the geeky programmer, are both communicating with the machine for a desired outcome.

But there is a world of difference between what these two are trying to do. A programmer communicates with the computer to enable it to perform tasks/functions/computations in the right way, while the end user engages with the computer to use the features it offers to get his/her work done(be it work and/or entertainment). The programmer uses a language to do so as opposed to a user who would use icons and not write even a line of code. A perusal into the history of programming languages, makes it quite obvious that an innumerable number of them have been made to tackle a number of specific real-time problems. The general problem which has remained though is that the ideal desideratum, one language to rule them all has not yet been achieved. The advancement of Artificial Intelligence and Cognitive Machine Learning (optimistically speaking) might create a panacea for the language wars we have seen on Reedit threads these days making HCI more like having a conversation with your friend.

The Future of Computing

The ubiquity of data in this day and age and its monstrous growth in the coming years will force Computer Science to come up with Computation Models and Logical Paradigms to manage this extreme outburst of data, besides the obvious need to simplify and unify. The recent super accelerated growth of Hadoop proves this beyond doubt. Computer Science can be used to solve problems from simple arithmetic to a complex one like predicting a meteor crash. In order to solve the huge variety of problems it is meant to, it's very *raison d'être*, it needs data, it needs to organize and analyze this data, it needs to establish different patterns and process this data so as to extract useful information that will (if the right steps are taken) facilitate improvement of various human life situations, thereby making the Earth a better, safer and more importantly a simpler and happier place to live in.

(VODNALA KRANTHI KUMAR)