God made integers ‘and algorithms’; all else is the work of man

The title is a (controversial) twist over the famous quote attributed to Leopold Kronecker “God made integers; all else is the work of man.” The biggest fundamental ‘abstraction’ in the human thought process that has emerged in the twentieth century is arguably that of an ‘algorithm’. Mathematics epitomizes modern day thought and has undoubtedly been the driver behind most of the remarkable technological and scientific advances over the last few centuries. As Eugene Wigner put it in his famous “The miracle of the appropriateness of the language of mathematics for the formulation of the laws of physics is a wonderful gift which we neither understand nor deserve. We should be grateful for it and hope that it will remain valid in future research and that it will extend, for better or for worse, to our pleasure, even though perhaps also to our bafflement, to wide branches of learning.” Computer Science however seems different from the other ‘natural sciences’ in that it is more like mathematics –- there are no ‘natural laws’ to be discovered, and almost all of computer science is a collection of man-made abstractions, just the way mathematics is. It also seems different from traditional mathematics –- it is ‘applied’ and often results in something tangible. We intend to trigger a thought provoking discussion among the eminent panelists –- both mathematicians and computer scientists –- on the interplay between these two disciplines and their impact on mathematics and computer science education; on the role of ‘algorithms’ and computer science methods in general in modern day mathematics and conversely the role of advanced mathematics in computer science. The latter is obviously much better understood than the former.

The idea is to trigger a constructive debate among the eminent panelists to enrich our attempts at seeking answers to questions such as (i) Is the ‘Lemma-Theorem-Proof’ paradigm the best way to inculcate mathematical thinking among students, computer science in particular? (ii) Is Calculus relevant for computer science education? (iii) Should traditional mathematics curricula be modified to enable the budding mathematicians inculcate algorithmic thinking? (iv) Is ‘Experimental Mathematics’ just experimental? (v) To quote from Donald Knuth’s essay on Computer Science and its Relation to Mathematics, “The difference is in the subject matter and approach –- mathematics dealing more-or-less with theorems, infinite process, static relationships, and computer science dealing more or less with algorithms, finitary constructions, dynamic relationships” --- clearly computer science is ‘constructivist’ in spirit. Is computer science ‘just’ constructive mathematics?

The ideas in this panel discussion will be communicated to the competent authorities concerned with education, and will hopefully have a long lasting impact.