Naturalness of Turing machines

Somenath Biswas
IIT Kanpur

(One dictionary entry for 'natural' is 'characterized by spontaneity and freedom from artificiality, affectation, or inhibitions'.)

Prior to Turing's proposal to define computability in terms of what we call today Turing machines, there existed two other proposals for the same purpose: lambda calculus (Church-Kleene) and general recursive functions (Herbrand-Gödel-Church-Kleene). Although, all the three proposals capture precisely the same class of functions to be computable, yet it is the notion of Turing machines which is credited (by Gödel, e.g.,) to be the first notion to have captured computability 'convincingly'. We discuss why. We attempt to get a glimpse of Alan Turing the person in reading through his 1936 paper which defined his machines. We also discuss how naturally the notion of Turing machine can lead us to the two great discoveries of 1930's, viz., Gödel's incompleteness theorem on the one hand, and the undecidability of the set of theorems of the predicate calculus, on the other.