Programming languages for biology

Andrew Phillips
Microsoft Research Cambridge, UK

This tutorial presents programming languages for three separate application areas in biology: DNA Computing, Synthetic Biology and Immunology. First, a programming language for designing molecular devices in DNA is presented. The language allows a range of molecular devices to be designed and simulated on a computer, and then compiled to DNA sequences. In future, such languages could be used to design a universal computer made solely of DNA. Second, a programming language for engineering living cells is presented. The language allows a range of genetic devices to be designed on a computer and compiled to genetic parts. Given a computational model and an extensive library of genetic parts, a compiler automatically selects the parts that satisfy the design constraints, allowing cells to be programmed more efficiently and reliably. In future, such languages could be used to program cells to address societal challenges in areas of food, medicine, energy and the environment. Third, a programming language for modelling existing biological systems is presented. The language is used to model some of the basic processes of viral detection in living cells, and to make preliminary predictions about how the immune system can be reprogrammed to improve its response to a given virus. In future, such languages could be used to program computational models of the immune system, in order to understand how it works and how to improve its ability to combat disease. The tutorial also describes basic principles of model simulation that are common to all three languages, and discusses how these three rather different languages can be unified under a common modelling engine.