Theoretical Computer Science 4 (1977) 351–353.
© North-Holland Publishing Company

ANNOUNCEMENT

ADVANCED SEMINAR ON SEMANTICS

September 26–October 7, 1977, Sophia–Antipolis, France

This seminar is organized by the Institut de Recherche d'Informatique et d'Automatique (I.R.I.A.), the Ecole Nationale Supérieure des Mines de Paris (E.N.S.M.P.), the University of Oxford and the University of Warwick.

Scientific Organizers

M. Nivat (University Paris VII and I.R.I.A.)
D.M.R. Park (University of Warwick)
G. Berry (E.N.S.M.P. and I.R.I.A.)
J.E. Stoy (University of Oxford)

Secretariat

Miss T. Bricheteau (I.R.I.A.)

Local Organization

Y. Rouchaleau (Centre de Mathématiques Appliquées, E.N.S.M.P.)

Presentation of the Seminar

This will be a Franco-British seminar on recent theories and results about the semantics of programming languages.

The form of semantic definition which has come to be known as "denotational" or "Scott–Strachey" semantics is one rigorous, concise and elegant technique for specifying the semantics of a programming language. The original development of this technique was principally the work of the late Cristopher Strachey, and at the same time the work of Dana Scott provided a sound mathematical theory. Other workers, both in Britain and elsewhere, have since helped to deepen and extend the method, covering areas such as implementation correctness and language design. Half of the seminar will be devoted to a connected and comprehensive exposition of denotational semantics, to be given by Dana Scott and a number of other workers in this field.

A number of other problems arising in connection with semantics have also recently been given partial solutions of a mathematical nature. The theory of program schemes originated in Russia and then spread out; a rather large group has been working in France on recursive schemes, which provide an algebraic framework in which many questions can be easily formulated and sometimes solved—equivalence problems, for example, and the definition and implementation of "good" computations of recursive programs. These concepts and ideas form a coherent theory, known as "algebraic semantics", which will also be presented during the seminar, mainly by French speakers from I.R.I.A.

Other active fields closely related to these theories include the definition of data structures, proofs of properties of programs and the study of transformations of programs. Several French and British scientists are working in areas such as these, and some of them will also be presenting their results.

The seminar is intended for established research workers, having a good basic knowledge of programming languages and computation theory, and also for some graduate students. We append below synopses for the expository lectures on "denotational semantics" (30 lectures) and "algebraic semantics" (16 lectures). There will also be about 14 other lectures on related topics, to be fixed later.

Denotational Semantics

Programming Language Description

The language definition techniques of denotational semantics, and their use for discussing various language concepts—control structures, the store, procedures, declarations, types, jumps, etc.... Worked examples.

General Computability Theory

An introduction to the mathematical theory underlying the techniques of denotational semantics: abstract domains, continuous functions, universal domains and retracts, fixed point induction, semantic equations.

Properties of Programs

The derivation of some results about program equivalences, and tools for establishing program correctness.

Implementation Semantics

An introduction to the role of denotational semantics in demonstrating the correctness of a language implementation.

Speakers

R.E. Milne (National Physical Laboratory)
P.D. Mosses (University of Aarhus; formerly Oxford)
D.M.R. Park (University of Warwick)
G.D. Plotkin (University of Edinburgh)
D.S. Scott (University of Oxford)
J.E. Stoy (University of Oxford)
R.D. Tennent (Grenoble and Oxford; on leave from Queen's University, Kingston, Ontario)

Algebraic Semantics

The Free Magma

The free magma and its completion. Systems of algebraic equations and algebraic infinite trees. Recursive program schemes and their interpretations.

Classes of Interpretations

Equivalence of programs in a class of interpretations. Herbrand interpretations. Algebraic classes of interpretations. Examples.

Relations between Operational Semantics, Denotational Semantics and Algebraic Semantics

Operational Semantics and program contexts. Equivalence of the three semantics.

Computations of Recursive Programs

Structure of the set of derivations. Minimal and optimal computations.

Speakers

C. Berry (E.N.S.M.P. and I.R.I.A.)
B. Courcelle (I.R.I.A.)
I. Guessarian (C.N.R.S.)
J.J. Levy (I.R.I.A.)
M. Nivat (University Paris VII and I.R.I.A.)
J.C. Raoult (University of Orsay Paris-Sud and I.R.I.A.)
J. Vuillemin (University of Orsay Paris-Sud and I.R.I.A.)

Related Topics

Transformation of Programs

A study of a system of syntatic transformations without any new axiomatic, and an introduction to the completeness of such a system.

Speaker

B. Robinet (University Paris VI)

Coroutines and concrete data types

One uses ideas in denotational semantics to understand an intricate mechanism existing in many programming languages.

Speaker

G. Kahn (I.R.I.A.)

General Information

Location

This seminar will be held at:

Centre de Mathématiques Appliquées, E.N.S.M.P.
Sophia–Antipolis
Valbonne
France