■7 群（コンピュータ-ソフトウェア）

1編 ソフトウェア基礎

Foundation of Software

（執筆者：坂部俊樹）［2009 年 6 月 受領］

■概要■

ソフトウェア基礎の分野は，ソフトウェアの信頼性，性能，開発手法などの基礎となっている理論を中心とした分野である．本編ではこの分野のなかで特に研究が盛んである，あるいは，応用が広がっている分野として，計算モデル，ソフトウェア検証及びモデル検査を取り上げる．

計算モデルは，計算を表現する抽象モデルであり，プログラミング言語やプログラム検証の基礎となるだけなく，計算量を解析するための土台として用いられるものもある．本編では 1 章において，計算の性質を解析することを主たる目的とするモデルを中心に取り上げ，解説する．

ソフトウェア検証は，ソフトウェアの振る舞いが所望の性質を満たすことを形式的に検証する手法に関する分野であり，ソフトウェアの信頼性向上のための技術の基礎を研究する分野である．所望の性質や振る舞いを表現するために特定の論理の論理式を用いる場合は，その論理における定理自動証明やモデル検査技法が応用される．性能の向上が著しい充足可能性判定ツールは，モデル検査などのほかの検証手法と組み合わされてソフトウェア検証手法の適用範囲の拡大に大きく貢献している．一方，ソフトウェアの設計を記述する枠組みの標準化が進められており，その枠組みにおけるソフトウェア検証の研究も進められている．本編では，2 章においてプログラム検証の論理，定理自動証明，充足可能性判定ツール，設計記述法を解説する

モデル検査の目的は，システムが所望の性質を満たすことを検証することである．検証される性質は時間に関するものが主であり，多くの場合，時相論理などの様相論理に基づく．近年，コンピュータの記憶量と速度の急速な向上に伴ってモデル検査の適用範囲が飛躍的に拡大している．モデル検査では，システムの抽象化モデルとしてオートマトンなどの状態遷移系を与え，そのモデルが論理式として記述された所望の性質を満たすことを機械的に判定する．検証の成否はシステムの抽象化の仕方に強く依存するため，よりよい抽象化を行うことが鍵となる．また，検証したい性質に適したモデルの枠組みを選ぶことも重要である．本編では，3 章において，モデル検査における抽象化，各種オートマトン，高速化技法について解説する．

【本編の構成】

本編では，計算モデル（1 章），ソフトウェア検証（2 章）及びモデル検査（3 章）について解説する．1 章では，木言語，並行計算，項書換え系，代数仕様について，2 章では，ホーア論理，Isabelle/HOL，SAT，モデル検査（総論），UML・ステートチャートについて，3 章では，抽象化，時間オートマトン，ハイブリッドオートマトン，高速化技法について，それらの基礎となる理論を中心に述べる．

【1 編 知識ベース委員会】