

「情報処理学会論文誌：プログラミング」の編集について

プログラミング研究会論文誌編集委員会

情報処理学会では、研究会の活性化を目指して様々な改革を進めている。プログラミング研究会はこの流れを受けて、研究会のあるべき姿について徹底的な討論を行ってきた。その帰結として、研究会独自の論文誌の編集にいち早く踏み切ることを決定した。

研究会論文誌「情報処理学会論文誌：プログラミング」の特徴と意義は大きく三つある。第一は、従来の「論文」に対して想定されてきた対象分野や査読基準では必ずしもカバーしきれない、多様な成果の公表の場を提供することである。第二は、投稿論文の内容を研究会で発表することを義務づけることによって、迅速で的確な査読を実現するとともに、議論の結果の最終稿へのフィードバックを可能にすることである。第三は、研究内容の表現に必要なであると認められれば、長大な論文も採録可能としている点である。

本論文誌を通じて、日本のプログラミング分野の研究活動を盛り上げてゆきたい。読者諸氏からの多くの論文投稿を期待する。

1. 対象分野

プログラミングは、コンピュータの誕生と同時に生まれた伝統的な分野であるが、コンピュータがある限り不可欠な技術である。並列分散処理やマルチメディア応用など処理内容が高度になるにつれて、プログラミングの重要性は増すことがあっても減ることはないであろう。

「情報処理学会論文誌：プログラミング」は、プログラミングに関するテーマ全般を専門に扱う論文誌である。具体例として次のようなテーマが挙げられる。

- プログラミング言語の設計、処理系の実装
- プログラミングの理論、基本概念
- プログラミング環境、支援システム
- プログラミング方法論、パラダイム

これらを応用したシステムの開発事例も対象に含まれる。また、上記以外でも、プログラミングに関する面白い話題であれば対象となる。

2. 編集方針

本論文誌は、プログラミング研究会における発表と論文誌投稿が密接にリンクされている点に特徴がある。

論文誌への投稿者が用意する研究会発表用の資料が、そのまま本論文誌への投稿論文となる。

研究会発表をせずに本論文誌に投稿することはできないが、逆に、本論文誌への投稿を伴わない研究会発表は可能である。そのような発表や、論文が不採録となった発表については、アブストラクトが本論文誌に掲載される。従来のプログラミング研究会の研究報告は廃止し、その代わりとして、研究会登録者には本論文誌が配布される。

本論文誌に掲載する論文は、通常のオリジナル論文と、サーベイ論文の2種類とする。どちらの種類であるかは、著者自身の指定によって決まる。論文の記述言語は日本語、英語のいずれかとする。論文の長さに制限は設けない。

3. 査読基準

基本的に、減点法に陥ることを避け、論文のよい点を積極的に評価するという方針を貫く。具体的には、新規性、有効性などの評価項目のうち、どれか一つの点で特に優れていると認められれば採録する。体裁のみが整った論文より、若干の不備があっても技術的な貢献の大きい論文を積極的に受け入れる。

このような観点から、たとえば次に挙げるような、従来は論文としてまとめることが難しかった内容について論じた論文もできるだけ受け入れる。

- プログラミング言語の設計論
- システムの開発経験に関する報告
- 斬新なアイデアの提案
- 概念の整理、分類法、尺度の提案
- 複数のシステムその他の比較

4. 投稿から掲載までの流れ

本論文誌への投稿希望者、および研究会での発表希望者は、発表会開催日の2~3か月前までに発表申込みをする。具体的な方法は研究会ホームページ <http://www.ipsj.or.jp/sig/pro/> を参照していただきたい。申込みの際には、本論文誌への投稿の有無、オリジナル論文とサーベイ論文の種別指定を明記する。また、アブストラクト（和英両方、和文は600字程度）を添付する。

論文投稿を希望した場合は、研究発表会の3週間前までに、別に定めるスタイル基準に従ったカメラレディ形式で論文を提出する。

毎回の研究発表会の直後、編集委員会が開催され、各論文について1名の査読者が決定される。査読報告をもとに、編集委員会は採録、条件付き採録、不採録のいずれかの判定を行ない、発表会開催後3週間程度で発表者に採否通知を行なう。照会の手続きはないが、論文改善のための付帯意見が添付される場合がある。この場合は、3週間以内に改良版を作成する。

5. 研究発表会

プログラミング研究会では、発表会ごとに特集テーマを設けている。ただし各発表会では、特集以外の一般の発表も常に受け付けている。

2000年度の発表会予定は次のとおりであり、各発表会の特集テーマは、今後数年間はそのまま維持する予定である。

- 6月15～16日 [プログラミング言語の設計と実装]
- 8月2～5日 [SWoPP - 並列/分散/協調プログラミング言語と処理系]
- 11月1～2日 [理論]
- 1月頃 [並列・分散処理]
- 3月頃 [プログラミング言語一般]

6. 編集母体

本論文誌は、下記のプログラミング研究会論文誌編集委員会の責任で編集を行なう。各研究発表会ごとに担当編集委員が割り当てられ、投稿論文の査読プロセスを主導する。必要に応じて、副担当編集委員を置いて、編集作業を分担することもできる。副担当編集委員は編集委員会メンバ以外から選任することもある。

1999年度プログラミング研究会論文誌編集委員会

- 委員長 上田和紀 (早稲田大学)
- 委員 天海良治 (NTT)
- 石畑 清 (明治大学)
- 伊知地宏 (富士ゼロックス)
- 久野 靖 (筑波大学)
- 柴山悦哉 (東京工業大学)
- 寺田 実 (東京大学)
- 西崎真也 (東京工業大学)
- 松岡 聡 (東京工業大学)
- 村上昌己 (岡山大学)
- 八杉昌宏 (京都大学)

この号の編集は、上記の1999年度論文誌編集委員会メンバにより行なわれた。2000年度の編集委員会は以下のメンバにより構成される。

2000年度プログラミング研究会論文誌編集委員会

- 委員長 柴山悦哉 (東京工業大学)
- 委員 天海良治 (NTT)
- 石畑 清 (明治大学)
- 伊知地宏 (富士ゼロックス)
- 岩崎秀哉 (東京大学)
- 上田和紀 (早稲田大学)
- 小川瑞史 (NTT)
- 小野寺民也 (日本IBM)
- 久野 靖 (筑波大学)
- 高木浩光 (電総研)
- 寺田 実 (東京大学)
- 富樫 敦 (静岡大学)
- 西崎真也 (東京工業大学)
- 松岡 聡 (東京工業大学)
- 村上昌己 (岡山大学)
- 八杉昌宏 (京都大学)

本号の編集にあたって

西崎 真也, 小川 瑞史

1999年度第3回プログラミング研究会は、1999年10月29日、30日に静岡県田方郡伊豆長岡町のNTTスイートヴィラ踊り子において開催された。

今回の研究会は特集「理論」として、学会会告などを通じて研究会発表を募集した。但し、特集のテーマだけには限定しなかった。また、これまでの研究発表会と同様に、研究会論文誌への投稿を伴う発表の他に、論文投稿を伴わない発表についても歓迎した。

発表申込には、理論に関するもの他、さまざまなテーマの研究発表が寄せられ、発表件数は12件となった。当日の発表のスケジュールは、これまでと同様に、発表25分、質疑20分とし、聴衆の必要に応じて適宜発表中に質問してもよいこととした。どの発表においても、活発な議論がなされた。

編集委員会は、10月30日の昼休み中と研究発表会終了後、2回開催した。編集委員の他、編集委員会が出席を依頼したメンバーにより、研究会論文誌に投稿を希望しているものについて、投稿論文ならびに研究会における発表内容、質疑応答にもとづいて詳細な検討をおこなったうえ、査読者の選定がおこなわれた。編集委員会の席上では発表論文に関して、どのような点が評価できるのかということを中心に議論し、問題点

や総合的な評価などについて意見が交換された。各査読者は、編集委員会での議論をふまえ査読をおこなった。論文誌投稿を希望した論文の中から、8件(すべて通常論文)が採録された。

掲載した論文について、以下に紹介する。

「浅い束縛による動的スコープ変数が存在する時の末尾再帰呼び出し」では、浅い束縛を用いて動的スコープ変数が実装されている場合において、空間計算量 $O(1)$ で実行できるような末尾再帰呼び出しを実現するための手法が提案され、その性能評価がおこなわれている。

「直観主義時相線形論理における論理プログラミングについて」では、直観主義時相線形論理に対する論理型プログラミング言語の設計がおこなわれ、その計算モデルが与えられている。

「世代別ごみ集めでのプログラムの文脈に基づくシンボルの配置法」では、世代別ごみ集めにおいて、古い世代でのオブジェクトの書き換えの局所性を向上させる方法として、プログラムの文脈に基づいてシンボルオブジェクトを配置する手法が提案され、その性能評価がおこなわれている。

「家電向け仮想マシンアーキテクチャ」では、家電

向けの Java の仮想マシンアーキテクチャ CVM が提案され、その実装が紹介され、その性能評価がおこなわれている。

「古典線形論理型プログラミング言語の静的解析の一手法について」では、古典線形論理に基づいた論理型プログラミング言語の静的解析方法が提案されている。

「Java アプレットのためのアニメーションヘルプシステム」では、Java アプレットの操作を説明するアニメーションヘルプを作成、編集、再生するためのシステムとその実装が紹介されている。

「On Proving AC-Termination by Argument Filtering Method」では、AC 項書き換え系の AC 停止性を証明する際に有用となる、“Argument Filtering Method” が提案され、その拡張が紹介されている。

「Java のクラスローダ制約の定式化」では、Java2 で型安全性を保証するために導入されたクラスローダ制約という概念を定式化する手法が、Java のサブセットを与えることにより紹介されている。

最後に、研究会開催、論文誌編集に御協力を賜ったみなさまに感謝いたします。