

---

 書 評
 

---

Kernighan and Plauger 著\*

## 『The Elements of Programming Style』\*\*

才 所 敏 明\*\*\*

従来のプログラミングにおいては、記憶領域が小さいとか、演算速度が遅い等のハードウェア上の制約のため、その製品であるプログラムの実行効率が、最大のそしてほぼ唯一の評価項と考えられてきた。

ところが、最近のハードウェア上のめざましい発展は、プログラミング評価項としての実行効率の重みを下げ、むしろ短期間のプログラム開発や、修正、変更の容易なプログラムの開発が求められてきた。

構造化プログラミングは、このような要求を満たすための一つのプログラミング方法で、Dijkstraにより提案されて以来多くの反響を呼び、プログラミング革命とまでいわれている。実際、構造化プログラミングを採用して大きな効果を得た例が数多く報告されている。

しかし、構造化プログラミングの問題は、それですべてではないということである。構造化プログラミングの約束を守っているプログラムでも非常にわかりにくいものがあるし、また逆に、構造化プログラミングの約束に違反していてもわかりやすいプログラムはたくさんある。

本書の著者達は、プログラムのわかりやすさが、プログラムのスタイル（表現の仕方）に強く依存していることを指摘している。また、プログラムのスタイル上の原則は、すべてのプログラム言語に共通であることを主張している。

本書は、よいプログラミング、よいスタイルのプログラムの書き方を、教えることを目的としている。プログラム例 (FORTRAN または PL/I) をあげ、批判し、よいスタイルのものへ書き換えるという過程を詳細に記述することにより、スタイルをよくすることがいかにプログラムをわかりやすくするかを認識させ、さらに、よいスタイルのプログラムを書くための指針

を与えている。

第一章では、プログラミング全般にわたる基本的な注意として、プログラムは明瞭に記述すべきであり、巧みさにおぼれてはならないということをあげている。

第二章は、式の表現上の注意で、変数名の慎重な選択、演算順序の明確化、共通関数の利用、テンポラリ変数使用の否定、をあげている。

第三章は、プログラム構造、データ構造上の注意で、データ構造の慎重な決定、制御構造上の注意 (GOTO の使用を限定)、モジュール化をあげている。

第四章は、入出力に関する注意で、入力データの正当性と妥当性の検査の必要性、プログラムの入力データからの保護、入力データ・エラーの明示とその回復、データ作成の容易な入力形式、それだけで意味のわかる出力、入力データのエコーの必要性、をあげている。

第五章は、一般に多い誤り、変数の初期化のミス、プログラムの特異点における処理のミス、添字付変数の添字のミス、実数値の扱いを注意している。

第六章は、プログラムの実行効率向上をはかるときの注意で、実行効率よりも正しさ、わかりやすさを優先すること、プログラムの詳細なコード上ではなく、アルゴリズム上で実行効率向上をはかると、効率向上のためには、まずプログラムの測定が必要であることをあげている。

第七章は、プログラムの説明書に関する注意である。プログラムが正しくなければ、あるいは説明書とプログラムに違いがあれば、説明書の意味はないこと、プログラム自体、説明書の役割を果し得るし、そうすべきであること、注釈は新しい情報を含まねばならないこと、変数名の適切な選択、プログラム・コードの適切な配置は、プログラムの説明書としての役割を果すのを助けること、をあげている。

この本が与えている多くの指針は、実際のプログラミング作業により影響を与えると思われる。

(昭和 50 年 4 月 22 日受付)

\* Bell Telephone Laboratories, Incorporated Murray Hill, New Jersey

\*\* A 5 版/160 p/1974 年/McGraw-Hill/\$ 3.95

\*\*\* 東京芝浦電気 (株) 経営情報システム部

---

 文献紹介
 

---

## 75-29 データの抽象化のための詳述化技術

Barbara H. Liskov and Stephen N. Zilles: Specification Techniques for Data Abstractions [IEEE Transaction on Software Engineering, Vol. SE-1, No. 1, pp. 7~19 (March 1975)] Key: data abstractions, programming methodology, proofs of correctness, specifications, specification techniques.

プログラムの正当性の証明とか、構成により正しいプログラムに導くプログラミング方式論にとって、形式的詳述化は重要であり、特にデータに関する抽象化の詳述化はそうである。

本論文では、詳述化技術の実際的なポテンシャルあるいは質といったものを評価するいくつかの基準と、データの抽象化の詳述を構成するために用いられる詳述化の技術の特に興味あるクラスが示される。

評価基準として、(1)数学的な取扱いができるほど形式的であるか、(2)困難なしに詳述が構成できるか、(3)理解しやすいか、(4)最小表現であるか、(5)適用範囲が広いか、(6)拡張可能であるか、等が問われる。また詳述化の単位の大きさが、その質を左右することの重要性が述べられている。

一般に関数の抽象化の詳述は、入出力の詳述で与えられるが、スタックのようなデータの抽象化には必ずしも十分でない。ここではそのような詳述化技術として、(1)V-graph を用いるような固定された規則の使用、(2)適当な規則を用いる方法、(3)状態機械のモデルを用いる方法、(4)公理的表現を用いる方法、(5)代数的定義を用いる方法、等があげられ比較評価されている。特定の詳述化技術が優れているという訳ではないが、各技術の大きな違いは、その詳述化が定義された抽象に対する実現を示す程度にあるとしている。

信頼すべきソフトウェアを作るのに形式的詳述化が基本的役割を果すのは、(1)信頼できるプログラムに対する経済的圧力が大きくなることにより、この方向の努力が正当化される、(2)新しい種類の単位(例えば multi procedure module)を認めることにより、詳述化が実行可能である詳述単位の同一視ができる、等の理由による。今後の興味ある問題としては、種々

の詳述化技術のエラーの取扱いについての考察があるが、筆者らはデータの抽象化の興味ある性質を示すことと、これらの性質を表現する詳述化の技術を発達させることが重要であると述べている。(米崎 直樹)

## 75-30 自動化ソフトウェア評価システムによる大規模なソフトウェアのテスト法

C. V. Ramamoorthy and Siu-Bun F. Ho: Testing Large Software with Automated Software Evaluation Systems [IEEE Transactions on Software Engineering, Vol. SE-1, No. 1, pp. 46~58 (March, 1975)] Key: Automated tools, correctness, performance, software evaluation systems, software reliability

ここ数年、大規模なソフトウェアシステムの質および信頼性を高めようという研究が盛んに行われている。しかしながら、規則的な方法により大規模なソフトウェアの正当性を証明するには、なおかなりの年月を必要とする。自動化ソフトウェアツールは、ソフトウェアの信頼性の向上およびそのコストを下げるための必要欠くべからざる道具であることがわかってきた。

本論文は、現在、利用されている自動化ソフトウェアツールの主な形態およびソフトウェア評価システムについて述べたものである。

自動化ツールは、大きく次の3つの項目に基づき分類される。(1)プログラムの解析が静的(非実行時)か動的(実行時)か。(2)開発工程(設計、テスト、メンテナンス)のどの段階で用いるのか。(3)ツールの機能。

(3)の機能はさらに、システム設計の解析(自動化設計ツールや自動化シミュレーションツール)、ソースプログラムの静的解析(コード解析、構造解析、モジュール間インターフェイスのチェック、事象順序のチェックなど)、ソースプログラムの動的解析(実行時の動作監視、自動的テストケース生成、relay runner 方式によるプログラムの保護など)、メンテナンス(ドキュメント生成、修正の正当性チェック)、効率の向上(プログラム再構成、並列処理の抽出および正当化)およびソフトウェアの品質評価ツールに分

けることができる。

ソフトウェア評価システムは、上記に示したいくつかの自動化ツールから構成されるシステムであり、どのツールを選択するかは、プログラミング言語、設計方法、コンパイラやローダの能力による。

自動化ソフトウェア評価システムの成功例として、FACES (Fortran Automatic Code Evaluation System), ACES (Automatic Evaluation System) および PACE(Product Assurance Confidence Evaluator) などについて述べられている。(伊東 祐蔵)

### 75-31 実行時におけるソフトウェア誤り検出

J. Richard Kane, Stephen S. Yau: Concurrent Software Fault Detection (IEEE Transactions on Software Engineering, Vol. SE-1, No. 1, pp. 87~99 (March 1975)), Key: algorithm, concurrent fault detection, control faults, fault model, implementation, performance, simulation

ハードウェア障害やプログラムの“虫”などによって発生するソフトウェア誤りは、データ値の誤りとかモジュールの実行シーケンス(系列)の誤りを招く。このような実行系列の誤りとなるコントロール誤りをシステムの実行時に検出する方法について述べている。

取扱うソフトウェアシステムはモジュールMからなるものとする。各トランザクションについて、これらのモジュールがある順序で実行される。システム内のモジュール間には順序関係Bが存在しており、その関係は方向性グラフで表わされる。コントロールのフローFは一般に4組  $F=(M, B, S, T)$  で表示される。S, T は、系列の開始および終了モジュールの集合である。コントロール誤りがソフトウェアシステムに及ぼす影響を調べるために、誤りなしのシステムを想定し、許容実行系列を定義する。さらに、それとの比較において correct な系列, invalid 系列を定義している。

本論文で対象とする誤りとコントロールに関しては一定の仮定を設けているが、任意のシステムをこのような仮定を満たすシステムに変換するアルゴリズムも与えている。

一般に、誤りの検出はモジュールの実行時間、トランザクションの処理時間あるいは実行系列を検査することによって可能である。システムおよびそのコントロールフローが検査可能である: 全ての incorrect な

系列が invalid となる条件を示し、各ブランチがコントロールされている well-controlled flow (w. c. f.) が少なくとも 1-検査可能であることを明らかにしている。また、任意のシステムから w. c. f. の構造をもつシステムへ変換するアルゴリズムを与えることによって、少なくとも 1-検査可能な等価フローが存在することを結論づけている。

このような誤り検出能力のインプリメントにあたっては、必要なストレージ、時間などについて考慮すべきである。その解決策として、連想メモリを用いた機構およびそれに基づいたコントロール誤りの検出手順(アルゴリズム)を提唱している。

最後に、この論文で展開された理論を裏付けるシミュレーション結果についても報告している。それによれば、先に述べたような仮定を満たすように設計されたシステムでは、この種のコントロール誤り検出が極めて有効であることが例証されている。(浦野 義頼)

### 75-32 マシンと構成に独立な Fortran: ポータブルな Fortran (PFortran)

D. E. Whitten and P. A. D. deMaine: A Machine and Configuration Independent Fortran: Portable Fortran (PFortran) (IEEE Transactions on Software Engineering, Vol. SE-1, No. 1, pp. 111~124 (March 1975)) Key: program portability, machine independent programming, Fortran

高級言語が、ホスト・マシンおよびその構成が変わっても同じ結果を導びくときポータブルであるという。本論文は Fortran をベースとするポータブルな高級言語 (PFortran) の設計思想と機能について述べている。

PFortran の言語仕様は、広い範囲のマシン上で受け入れられるように、7種の Fortran 言語仕様の共通部分集合として設定されている。PFortran 固有のポータブルな機能としては、核 (Kernel) と呼ばれるマシン独立なデータ単位、仮想配列、可変精度の算術演算、マシン独立な I/O 等があり、PFortran サポート・パッケージと呼ばれる 40 個のサブプログラムとして実現されている。

核のサイズはユーザによって指定できる。可変精度算術演算は核ストリングに対する演算として実現され、核内のビットに対するオペレーションも可能である。仮想配列は 50 次元まで指定可能で、添字として任意の算術表現を取り得る。その要素としては通常のマシン従属な語と同様に核が指定できる。サポート・

パッケージにはページング機構が含まれ、ユーザのデータ量は割当てられるメモリ量の制限を受けない。I/O 操作はサポート・パッケージにより行われ、デバイスやフォーマットの指定はイニシャリゼーション時に行われる。

本論文には、いくつかのサポート・パッケージ中のサブプログラムについてその実行時間を示している。上記のような言語でポータブルなプログラムを記述す

ると、サポート・パッケージ中のサブプログラムに対する参照が多くなり、プログラムの論理構造が不明確となる。また、既存の Fortran プログラムの PFortran への変換は人手で行わねばならない。このような弊害を除くには、PFortran 固有の機能を文法的に含む新しい言語を設定し、上記の言語に変換するトランスレータを実現すれば良い。(熊野 喜一)

---

---

## ニ ュ ー ス

---

---

### McCarthy 教授による MTC および人工知能の特別講義

Stanford 大学の John McCarthy 教授が、6月4日～6日の3日間、電子技術総合研究所で、MTC および人工知能に関する特別講義を行なった。これは京都大学滞在中の同教授を大型プロジェクト「パターン情報処理システム」の海外流動研究員として同所が招いたものである。

McCarthy 教授は、LISP の生みの親として広く知られており、MTC の分野でも著名である。同教授はまた、人工知能の基礎的研究にも深い見識を有し、

MTC を人工知能の研究における一つの重要な応用分野とみなして両者の橋わたしを試みている。

第1日目の講義は、拡張一階述語論理 (Dana Scott により導入された束論に基づくプログラムと一階述語論理の融合を目的としたもの) についての話で、残りの2日間は人工知能における認識論的問題と題して、主に知能の表現方法に焦点をあてて講義が行なわれた。同教授の話は、概して哲学的でかなり難解な部分も多かったが、認識論的アプローチが今後の人工知能研究での重要な課題の一つであることが強調され、具体的研究課題も示された。

---



---

## 今月の筆者紹介

---



---

**矢野秀一郎 (学生会員)**

昭和26年生。昭和49年大阪大学基礎工学部情報工学科卒業。現在同大学院修士課程在学中。プログラミング技法やプログラム理論に興味をもつ。電子通信学会会員。

**奥井 順 (正会員)**

昭和20年生。昭和45年大阪大学基礎工学部制御工学科卒業。昭和47年同大学院修士課程修了。現在同大学院博士課程在学中。オペレーティングシステムなどに関して、プログラム理論やデッドロックの面からの研究に従事している。電子通信学会、IEEE 各会員。

**都倉 信樹 (正会員)**

昭和14年生。昭和38年大阪大学工学部電子工学科卒業。昭和43年同大学院博士課程修了。工学博士。現在、大阪大学基礎工学部情報工学科助教授。主としてプログラムの技法と理論に関する研究を行なっている。電子通信学会、IEEE 各会員。

**柴谷 善郎 (正会員)**

昭和8年生。昭和31年京都大学工学部電気工学科卒業。川崎航空機工業(株)、京都大学工学部助手、神戸工業(株)を経て、38年より大阪工業大学工学部電子工学教室に勤務。電子通信学会会員。

**久津輪敏郎 (正会員)**

昭和16年生。昭和41年大阪工業大学電子工学科卒業。同年同大学助手、43年大阪府立大学大学院入学、48年同博士課程修了。工学博士。同年大阪工業大学へ復職。電子工学科講師、確率順序機械、セル論理回路、故障診断の研究に従事している。電子通信学会会員。

**江端 克彦 (正会員)**

昭和18年生。昭和41年大阪工業大学電子工学科卒業。43年同大学院修士課程修了。同年より広島電機大学電子工学科助手を経て45年講師。可変論理回路、故障診断に関する研究を行なっている。電子通信学会会員。

**トラン・ティン・アム**

昭和22年生。昭和44年 The National Institute of Technology (Saigon, 南ベトナム) 電気工学科卒業。昭和48年電気通信大学応用電子工科大学院修

士課程修了。現在、大阪大学電子工学科博士課程在学中。グラフ理論、情報処理などに興味をもつ。電子通信学会会員。

**築山 修治**

昭和24年生。昭和47年大阪大学工学部電子工学科卒業。昭和49年同大学院修士課程修了。現在、同大学院博士課程在学中。グラフ理論の応用に関する研究に従事。電子通信学会会員。

**白川 功**

昭和14年生。昭和38年大阪大学工学部電子工学科卒業。昭和43年同大学院博士課程修了。工学博士。同年4月大阪大学助手。昭和48年同助教工学部電子工学科勤務。グラフ理論の応用に興味をもつ。著書「情報回路論I」(共著)、「グラフとネットワークの理論」(共著)、「グラフ理論」(共著)(以上コロナ社)、電子通信学会、IEEE 各会員。

**尾崎 弘 (正会員)**

大正9年生。昭和17年9月、大阪帝国大学工学部通信工学科卒業。母校勤務となるも、昭和17年9月～20年8月まで技術科士官として海軍に服役。現在、演算電子工学講座担当教授。工学博士。RC回路論、多変数正実関数論、グラフ理論などの回路理論、ならびに演算電子工学の研究と教育に従事。著書「回路網理論I」、「情報回路論I」、「ディジタル代数学」、「グラフとネットワークの理論」など。電子通信学会会員。現在副会長。IEEE Fellow。

**渡辺 豊英 (正会員)**

昭和23年生。昭和47年京都大学理学部物理学系卒業。昭和49年京都大学大学院工学研究科数理工学専攻修士課程修了。現在、博士課程在学。プログラミング言語、計算機システムを主として、データの構造、自然言語、生体情報機構に興味をもつ。電子通信学会、日本ME学会、日本行動計量学会各会員。

**宮脇富士男 (正会員)**

昭和10年生。昭和37年姫路工業大学電気科卒業。現在、同科助手、昭和47年より京都大学工学部情報工学教室萩原研究室において研修員としてAPL処理システムの作成に従事し、その後、ファームウェアによる言語処理の研究にとりくんでいる。著書「汎用土木工学プログラムシステム(ICES)概説」(共訳、

北尾書籍). 電気学会, 電子通信学会各会員.

**渡辺 勝正 (正会員)**

昭和15年生. 昭和38年京都大学工学部数理工学科卒業. 京都大学工学博士. 京都大学工学部助教授. コンパイラならびにコンパイラ・コンパイラの研究を経て, 現在, 高級言語計算機, およびプログラム作成の自動化の研究に従事している. 電子通信学会, 日本自動制御協会各会員.

**萩原 宏 (正会員)**

大正15年生. 昭和25年京都大学工学部電気工学科卒業, NHKを経て, 昭和32年京都大学工学部助教授, 36年同教授となり, 同学部にて計算機システム, 特にその性能評価, 設計の自動化, データ構造, マイクロプログラミング方式などの研究に従事. 現在同学部情報工学教室に勤務している. 工学博士. 著書に「電子計算機通論 1, 2, 3」ほか数冊がある. 電子通信学会, 電気学会, ACM 各会員.

**小林富士男 (正会員)**

昭和12年生. 昭和35年神奈川大学工学部電気工学科卒業, 36年東京工業大学助手となり現在に至る. 主としてオプト・エレクトロニクス, データ処理関係の研究に従事. 電気学会, 電子通信学会, 照明学会各会員.

**山口昌一郎**

昭和3年生. 昭和26年東京工業大学電気工学科卒

業, 同年4月同大学助手, 35年同助教授, 40年同教授現在に至る. この間カナダ国立科学研究所客員研究員, 光電工学, レーザとその応用, 視覚情報, パターン認識の研究に従事. 工学博士. 著書「基礎電磁気学」ほか. 電気学会, 電子通信学会, 照明学会, 応用物理学会, テレビジョン学会各会員.

**小笠原謙蔵 (第16巻7号参照)**

**黒崎 悦明 (正会員)**

昭和14年生. 昭和38年名古屋大学工学部電子工学科卒業. 同年, 沖電気工業(株)入社後, DC増幅器, パターン認識, 言語処理の研究に従事. この間, 昭和40年より2年間九州大学に国内留学. 現在, 同社ソフトウェア事業部制御システム第1G長. コンピュータ制御システムの実用化に従事している.

**長谷川実郎 (正会員)**

昭和5年生. 昭和28年北海道大学工学部電気工学科卒業. 八咫電機(株)を経て, 昭和35年に創設された日本電子産業(株)(8月20日日本電気漢字システム株に社名変更)に入社, セメント工場のプロセス・コンピュータ・コントロールの開発に従事. 昭和39年より, 漢字情報処理システムの開発を担当し, 各種漢字プリンタ, ディスプレイ, 植字装置, COM, 編集ソフトウェアの開発に従事し, 現在に至る. 現在同社取締役システム開発部長. 画像電子学会会員.

---

 研究会
 

---

## ○第2回計算言語学研究会

{昭和50年7月25日(金), 於電子技術総合研究所  
A会議室, 出席者30名}

## (1) 日本語のコンコーダンス

坂本義行(電総研), 岡本哲也(電通大)

## 〔内容梗概〕

自然言語の調査, 研究のため, 計算機を用いて原データの自動的な組織化ならびに統計資料の作成システムの開発を行ってきた。

ここで紹介するのは, 日本語の漢字かな混り文を標本とし, 自動分かち書き, KWIC Index File, 語の統計資料の蓄積, 検索, 出力システムに関する報告である。

高速の漢字プリンタを使用することによって, 多量の日本語のデータを処理することが可能となった。ま

た, 他の外国語の処理にも応用できる。

(計算言語学研資料 75-2)

## (2) 日本語構文解析システムやまと

西村 恕彦(電総研)

## 〔内容梗概〕

日本語文章を構文解析するシステムを作り, 実験した。辞書引きは最長一致によるので, 事前の分かち書きの必要はない。一つの文字列は長さ0以上の一つの品詞列に書き換えられる。品詞には, 類と族がある。類は, 語および連語の集合の素なる類別(90+80)である。類の集合が族(20)であって, 一つの類は一つ以上の族に属することができる。品詞の列が文脈評価文法によって一つの品詞に書き換えられる。50個の文を処理して, ほぼ正しいもの33, 複文のかかりの悪いもの13, 省略文2, その他2となった。

(計算言語学研資料 75-2)

---

 本 会 記 事
 

---

## ○入会者

昭和50年8月の理事会で入会を承認された方々は以下のとおりです(会員番号順, 敬称略)。

【正会員】 小鳥久幸, 沢井正彦, 渋谷大海, 伊東佐禮, 村元良昭, 西堀 齊, 川上雅昭, 佐藤 滋, 室信治, 服部文夫, 宇野陽一, 昔農正敏, 小林健三, 吉川良一, 池田耕治, 杉本英二, 岡田博夫, 松岡健次, 鈴木 裕, 坂藤繁夫, 平山正治, 金吉雅人, 深海規, 石井正純, 工藤峰利, 倉田信夫, 脊井孝昭, 荒田一郎, 森 成司, 林 利弘, 阿草清滋, 塚本耕平, 米澤宣義, 樺沢一之, 野田芳真郎, 小野直紀, 化生順治, 土師総一, 前田隆正, 刑部 稔, 大倉良昭, 池内克史, 浅原陸男, 柳瀬毅彦, 高橋 寛, 宮沢慎介, 堀田敬志, 三沢常男, 芳川真文, 朽見真一, 斎藤市郎,

竹内恒夫, 梁川博生, 山本 明, 国立 勉, 松岡 毅, 保坂峰夫, 根岸和義, 川辺秀一, 天沼義政, 小川哲夫(以上61名)

【学生会員】 高木雅史, 幡野 仁, 山下真澄, 杉田雅彦, 本位田真一, 古村 高, 鈴木 潔, 豊島 淳, 深沢良彰, 八重田茂, 新井利明, 石井和喜, 坂前和市, 三嶋良武, 長田格, 檜山重雄, 斎藤俊治, 小原永, 島 健一, 飯田憲治, 入沢寿美(以上21名)

## ○採用原稿

昭和50年7月に採用された原稿は以下のとおりです(採用順, カッコ内は寄稿年月日)。

## 論 文

▶市田浩三, 吉本富士市, 清野 武: ワンパス法によるデータ平滑化の安定性 (50.4.23)

▶益田隆司, 塩田博行: 仮想メモリシステムにおけるプログラムの局所性とその最大化 (50. 1. 30)

▶中野康明, 安田道夫, 中島 晃: 印刷文字認識における位置ずれ補正の効果 (50. 3. 14)

#### 資 料

▶加藤誠巳, 杉本隆重: デジタル計算機を用いた多

分岐信号伝送線路の過渡解析手法 (50. 5. 22)

▶尾上守夫, 柴田義文: 紙テープを利用した文字および画像の表示 (50. 6. 27)

#### ショート・ノート

▶斎藤信男: 腕木システムの一般化 (50. 5. 13)

▶上田和宏: 論理回路の配置アルゴリズム (49. 8. 15)

### 昭和 50 年度役員

会 長	北川敏男
副 会 長	猪瀬 博, 廣田憲一郎
常 務 理 事	相磯秀夫, 稲田伸一, 後藤英一, 鈴木錠造, 高橋延匡
理 事	山本卓真, 伊吹公夫, 大前義次, 落合 進, 佐川俊一, 三浦武雄, 山本欣子, 渡部 和
監 事	海宝 顕, 長森享三
関西支部長	田中幸吉
東北支部長	高橋 理

### 編 集 委 員 会

担当常務理事	相磯秀夫
担 当 理 事	伊吹公夫, 渡部 和
委 員	石黒栄一, 石野福彌, 宇都宮公訓, 小野欽司, 大畑 徹, 岡田康行, 片山卓也, 亀田寿夫, 木村 泉, 岸 慎, 首藤 勝, 田中穂積, 高橋義造, 武田俊男, 棟上昭男, 名取 亮, 中西正和, 西木俊彦, 野末尚次, 発田 弘, 服部幸英, 藤田輝昭, 古川康一, 益田隆司, 松尾益次郎, 松下 温, 三上 徹, 三木彬生, 村上国男, 森 敬, 山下真一郎, 山田邦雄, 米田英一