

## 10. PASCAL コンパイラの portability について

武 市 正 人 (東京大学工学部)

### § 1. はじめに PASCAL について

プログラミング言語 PASCAL は、1970年に、N. Wirth によって設計されたものである ([1])。その後、いくつかの点が改訂され、1973年に Revised Report ([2]) が出された。

ALGOL 60 をもとに、データ構造の記述能力を高めたものであるが、ALGOL 60 の own の概念などは取り入れられていない。

N. Wirth は、PASCAL が、

- (1) プログラミングの教育の便利な道具であり、また、
- (2) 大きいプログラムを書くための、効率のよい道具である

ことを意図して、

- (a) 基本的な概念は少数に厳選し、
- (b) 言語の構造は簡潔で、系統的であり、しかも
- (c) コンパイラが容易に作製できる

ように設計したと述べている。

(1)については、[3]、[4]などに論じられ、N. Wirth のテキスト ([5]) もある。

いくつかの、PASCAL コンパイラが、PASCAL で書かれていて、(2)に述べられているプログラムの実例とみることができる。

C. A. R. Hoare 等によって、PASCAL の Axiomatic な定義も行われ ([6])、[7] では、アルゴリズムを記述する際に PASCAL が用いられている。

一方で、PASCAL に対する注目すべき批判もあり ([8])、その回答は行われていない状況であるが、Revised Report をさらに検討する必要がある。

### § 2. PASCAL の コンパイラ

PASCAL のコンパイラは、まず、N. Wirth によって、CDC 6000 シリーズの SCOPE モータのもとに作製された ([9])。

PASCAL で書かれているコンパイラは、約 4000 行で、ブートストラップを行って、完成したものである。CDC を用いている大学を中心に、このコンパイラが使用され、改良されてきた。

1974年1月に、これらの大学間の連絡および他機種のコンパイラについての情報を伝える PASCAL NEWSLETTER ([10]) が出され、PASCAL コンパイラの利用度も知ることができる。

U. Ammann は、PASCAL コンパイラを structured に作製する試みを行い ([11])、このコンパイラも、いくつかの大学で使用されている。コンパイラは、PASCAL で書かれ (約 3400 行) CDC の PASCAL コンパイラのもとで作製されたもので、一部を除いて、Revised Report に従っている。このコンパイラの出力する目的プログラムは、仮想のスタック機械の命令語で書かれたものであり、実行の際には、特定の計算機で、

各命令をシミュレートすることになる。このコンパイラは、PASCAL-P (P-コンパイラ) と呼ばれている。

### § 3. Portable コンパイラ

コンパイラの、異機種間移行については、いくつかの方法が考えられるが、ここでは、つぎのような方法を考える。

計算機Mで動いている言語LのコンパイラCを、計算機M'に移行してコンパイラC'を作製することを考える。

コンパイラCが、M、M'の双方に処理系の備っている言語Pで書かれているならば、目的プログラム生成の部分をもM'向きに書き直せば、コンパイラの移行を行うことができる。現状では、PとしてFORTRAN語、COBOL等が考えられるであろう。しかし、これらの言語は、コンパイラの記述作製には向いているとは言えない。

ここでは、言語Pの処理系が、Mには備っているが、M'にはない場合を扱う。このとき、計算機Mで、M'向きのLのコンパイラを作製すれば、コンパイラC'が出来ることになるが、コンパイラC'はPで書かれているので、PとLが異なるときには手直しの際にはMの上で行うことになる。もちろん、PとLが同じ言語であるならば、C'はM'の上で容易に保守できる。

PASCAL コンパイラの移行に際しては、P、Lとして、PASCALを用いるのが便利である。

[12]には、

M …… CDC 6000            M' …… ICL 1900

の場合が報告されている。

この方法で、PASCAL コンパイラを移行するには、PASCALで、M'の命令語を生成するコンパイラを書き、Mを用いてC'を作製することになる。

しかし、Mを用いることが不可能な場合には、別的手段を講じる必要がある。M'の上でMをシミュレートし、コンパイラCを動かして、上と同様のことを行えば解決する。

§ 2にあげたP-コンパイラは、Mとして、実在の計算機ではなく、むしろPASCALで書かれたプログラムの目的プログラムの生成が簡単であるような仮想の計算機を用いている。

M'の上でMの動きをシミュレートする際には、一般には、効率が問題とされるが、この場合には、適度な効率さえ保証されれば、以後は、C'が動くことになるので問題でない。むしろ、Mのシミュレータ作製の手間が重要である。

次項では、P-コンパイラ移行の第1段階としてのシミュレータ作製の実際について述べる。

### § 4. PASCAL コンパイラ移行の第1段階

前出のP-コンパイラは、いくつかの計算機で、仮想機械のシミュレータが作製され、実際に用いられている。

IBM 370 OS/VS IIのもとで作製されたP-コンパイラおよびシミュレータのプログラムを入手し、東京大学教育用計算機センターのMELCOM 7700 BPMのもとで動作するコンパイラを作製した報告を行う。

○利用した文書等

- (1) IBM 370 版 PASCAL コンパイラ/インタプリタ ([13])
- (2) 上記 interpretive PASCAL の概容 ([14])

#### ○作製期間

上記資料を入手したのが 1974 年 6 月下旬。

教育用計算機センターを利用する学生が少数になる 7 月、8 月を予定した。

#### ○作製にたずさわった人数

7 月 …… 1 人

8 月 …… 4 人

#### ○経 過

入手した IBM 370 版 PASCAL コンパイラ/インタプリタは、U. Ammann の、P-コンパイラを用い、FORTRAN、アセンブラ語で IBM 370 向けに、仮想機械のシミュレータを書いたものである。

仮想機械は、60 種類の演算命令と、約 30 種の標準手続き（入出力、各種関数）を備えている。

IBM 370 では、コンパイラに対する制御文の解析を行うプログラムをも含む膨大な（60 K byte 余）ものであり、アセンブラ語で書かれたシミュレータ部と、それを制御する FORTRAN で書かれたプログラムから成っている。入手した磁気テープには、これらのプログラムのほか、PASCAL で書かれたコンパイラのソースプログラム（欠除部分あり）と、それが翻訳された仮想機械命令（2 進）、上記シミュレータ等のロードモジュールが入っている。

十分な資料を入手する時間がないことと、こうして作製したコンパイラは、以後の開発のための過渡的なものであるという考えにより、IBM 版の方式には従わずに、必要最小限のプログラムを作製することにした。たのために、実際に、そのまま利用するのは、仮想機械語で書かれたコンパイラ（59080 byte）だけとし、他は、シミュレータ作製の際の資料にとどめた。

以下、作製の過程を順に列挙する。

1. FORTRAN 語で、仮想機械のシミュレータを作製。FORTRAN 語で約 1000 行。
2. P-コンパイラの目的プログラムは、仮想機械のアセンブラ語であるので、FORTRAN 語で、アセンブラおよび、1. のシミュレータに手渡すプログラムを作製。FORTRAN 語で約 400 行。
3. 1、2 の作業を MELCOM 7700 アセンブラ語で行う。翻訳する際に必要な標準手続きだけを組み込み、他は後で考慮する。

アセンブラ語で約 2000 ステップ。

1、2 の作業を行ったのは、仮想機械語で書かれたコンパイラの動作確認のためで、約 10 日間で、一応、PASCAL プログラムの翻訳、実行が可能になった。もちろん、速度が問題になるが、それ以上に、必要とされる記憶場所の大きさが予想より大きく、早急に 3. の作業を行う必要があった。

最終的な、MELCOM 7700 の機械語命令を生成するコンパイラ作製を待たずに、能率はあまり良くないが、学生の実習にも耐えられる形にする作業も行われ、コンパイラに対する指令も処理できるようになった。このコンパイラをインタプリティブ・コンパイラ（I-コンパイラ）と呼ぶことにする。

### § 5. I-コンパイラの評価

前項で述べたように、I-コンパイラは比較的短期間で小人数で作製することができる。1～4 人・月の程度

であろう。しかし、入手した P-コンパイラ/インタプリタの欠陥がそのまま残されていて、改良を要する点は多い。

#### 1. 記憶場所の大きさ

I-コンパイラの致命的欠陥である。データの取扱いが、すべて 8 byte (64 bit) 単位になっていて、1 byte で扱える“文字”型データに対してさえ 8 byte を与えている。これは、P-コンパイラが最初 CDC 6000 (1 word = 60 bit) で設計され、仮想機械に対する注意が欠除していたものと考えられる。このために、コンパイラの持つ表を作製するだけでも、最低 48 K byte 必要で(もちろん、コンパイラ的设计もあまり良いとはいえない)、仮想機械語命令のコンパイラとあわせて、108 K byte は必要である。さらに、シミュレータのために約 10 K byte 必要である。

#### 2. 実行時間

仮想機械の 1 命令の実行速度は、それをシミュレートする機械にも依存するが、特殊な命令を除いて、1 命令あたり、約 10 ステップの MELCOM 7700 の機械語命令を実行する。コンパイラの大きさ、Portability の代価として、この程度の速度は、妥当なものと考えられる。

### § 6. PASCAL コンパイラ移行の第 2 段階

[12]には、PASCAL コンパイラを、4人・月で、異機種に移行した際の報告がある。I-コンパイラによって、同様の手続きで、手元にある計算機でコンパイラを開発できる状況にある。10月現在、MELCOM 7700 および、FACOM 230-38 に対してコンパイラを作製中である。

### § 7. Portable コンパイラの問題点

§ 5でも触れたように、Portable コンパイラは、とくに、データの表現に対して充分注意を払わなければ、実用的でない。PASCAL の、豊富なデータ構造の表現能力から、現在の P-コンパイラを改良して、新しい Portable コンパイラを設計することは可能であると考えられる。コンパイラの使用目的にもよるが、PASCAL の 1 つの目的である教育用の道具としては、インタプリティブに実行するコンパイラでも(当面は)十分な役割を果たすと思われる。

### § 8. おわりに

本発表は、主に、PASCAL コンパイラ作製の 1 つの例の報告である。

§ 4 に述べた、I-コンパイラ作製にあたっては、三菱電機の中村和也、斎藤弘謨、鈴木美知子、吉沢恵子の各氏の協力を得た。

PASCAL コンパイラ作製と並行して、実験的にミニコン向けの ALGOL-like 言語 PLAN ([15]) の、Portable コンパイラの作製も、上と同様の方法で行っている。PLAN コンパイラ/インタプリタの作製は、東京大学大学院情報工学専門課程の松田元男、庵原信男両氏による。

協力された方々に感謝いたします。

## § 9. 参考文献

- [1] N. Wirth, "The Programming Language PASCAL", Acta Informatica 1, 35 - 63, (1971).
- [2] N. Wirth, "The Programming Language PASCAL (Revised Report)", Berichte der Fachgruppe Computer-Wissenschaften, ETH, № 5, Eidgenössische Technische Hochschule Zürich, July 1973.
- [3] Q. Lecarme, "What Programming Language Should We Use for Teaching Programming?" in Program Teaching Techniques (Proceedings of the IFIP TC-2) ed. by W. M. Turski, 61 - 71, North-Holland, (1973).
- [4] A. Ralston, "The Future of Higher Level Languages (in Teaching)" in Proceedings of the International Computing Symposium 1973, ed. by A. Günter et al., 1 - 10, North-Holland, (1974).
- [5] N. Wirth, Systematic Programming: An Introduction, Prentice-Hall, (1973).
- [6] C. A. R. Hoare, N. Wirth, "An Axiomatic Definition of the Programming Language PASCAL", Bericht der Fachgruppe Computer-Wissenschaften, ETH, № 6, Eidgenössische Technische Hochschule Zürich, Dec. 1972.
- [7] P. B. Hansen, Operating System Principles, Prentice-Hall, (1973).
- [8] A. N. Habermann, "Critical Comments on the Programming Language PASCAL", Acta Informatica 3, 47 - 57, (1973).
- [9] N. Wirth, "The Design of a PASCAL Compiler", Software - Practice and Experience 1, 309 - 333, (1971).
- [10] PASCAL NEWSLETTER № 1, ed. by G. H. Richmond, Jan. 1974.
- [11] U. Ammann, "The Method of Structured Programming Applied to the Development of a Compiler" in Proceedings of the International Computing Symposium 1973, ed. by A. Günter et al., 93 - 100, North-Holland, (1974).
- [12] J. Welsh, C. Quinn, "A PASCAL Compiler for ICL 1900 Series Computers", Software - Practice and Experience 2, 73 - 77, (1972).
- [13] U. Ammann, "Step-wise Development of a PASCAL Compiler, Step 5: Syntax Analysis Including Error Handling; Checks Based on Declarations; Address and Code Generation for a Hypothetical Stack Computer", 1973 - 03 - 15.
- [14] R. S. Deverill, A. C. Hartmann, "Interpretive PASCAL for the IBM 370", Information Science Technical Report № 6, Dec. 1973, California Institute of Technology.
- [15] 武市正人, "ミニ言語のミニコンパイラ① ⑤", bit 6, 8月号~12月号(1974)に連載。

## § 10. プログラムの例

[5] の P. 137 にあるプログラムを I-コンパイラで翻訳実行した結果を示す。

- ✓ 問題：  $x = a^3 + b^3 = c^3 + d^3$  ( $a \neq c, a \neq d, a, b, c, d$  は自然数) のように表わされる自然数  $x$  のうち、最小のものを求める。

I-コンパイラによるもの、シミュレータが FORTRAN で書かれているもの (§ 4 参照) によるもの、それと、同じアルゴリズムで、FORTRAN によってプログラムを書き、MELCOM 7000 EXTENDED FORTRAN IV 処理系で処理したもの、FLAG (FORTRAN Load And Go) 処理系で処理したものの所要時間の比較を示す。

| 処 理 系                       | 所要時間(分) |
|-----------------------------|---------|
| PASCAL<br>I-Compiler        | 0.291   |
| PASCAL<br>FORTRAN Simulator | 1.372   |
| FORTAN IV                   | 0.698   |
| FLAG                        | 0.169   |

```

6
6 ER
6 "FINDING THE LEAST NUMBRT THAT IS EQUAL
6 TO TWO DIFFERENT SUMS OF TWO NATURAL
6 NUMBERS RAISED TO THE THIRD POWER"
6
6 VAR I,IL,IH,MIN,A,B,K : INTEGER;
13 J,P,S : ARRAY (. 1..12 .) OF INTEGER;
49 BEGIN I:=1; IL:=1; IH:=2;
47 J(.1.):=1; P(.1.):=1; S(.1.):=2;
68 J(.2.):=1; P(.2.):=8; S(.2.):=9;
89 REPEAT MIN:=S(.1.); A:=I; B:=J(.I.);
105 IF J(.I.)=I THEN IL:=IL+1
115 ELSE
119 BEGIN IF J(.I.)=1 THEN
128 BEGIN IH:=IH+1; P(.IH.):=IH*IH*IH;
143 J(.IH.):=1; S(.IH.):=P(.IH.)+1
161 END;
164 J(.I.):=J(.I.)+1; S(.I.):=P(.I.)+P(.J(.I).)
199 END;
202 I:=IL; K:=I;
206 WHILE K<IH DO
210 BEGIN K:=K+1; IF S(.K.)<S(.I.) THEN I:=K END
230 UNTIL S(.I.)=MIN;
240 WRITE(MIN:10,'=',A:2,'**3+',B:2,'**3=',
260 I:2,'**3+',J(.I.):2,'**3',EOL)
282 END.

```

```

40 ENT 48
LDCI 1
SRO 12
LDCI 1
SRO 11 1
LDCI 2
SRO 10
LAO 37
LDCI 1
CHK 1 12
50 DEC 1
IXA 1
LDCI 1
STO
LAO 25

```

```

LDCI 0
280 LDCI 1
CSP WRC
RETP
0 MST 0
CUP 0 40
STP

```

$$1729=10^{**3}+ 9^{**3}=12^{**3}+ 1^{**3}$$