

モジュール制御による自動プログラミングへの接近

仙台電波工専

大林 久人

㈱ソフト工学研究所

伊賀 武

プログラミングの自動化は、待望久しいものでありながら、その実現は霧の中を模索するような不確実な道程でもあります。

ここに、われわれはCOMP S (Controlled Modular Programming System) と呼ぶプログラミング自動化ツールを作ったので、ご批評を仰ぎたく報告いたします。

COMP Sは、オンラインで使用し、システム設計を省力化しコーディングの半分以上を自動化しようとするパッケージとして作られ、このバックボーンは階層化フローチャート (Hierarchy Organization Flow Chart: HOF) 手法と呼ばれるプログラミング手法とプログラムに関する1つの考え方——プログラムはすべて機能を実行するモジュールその間の流れを抑制するコントロールに分けることができるという概念——であります。

1 使用手順

1. プロセス設計が終わったところで、1工程 (1プログラム) ずつの入出力ファイル、処理概要を条件として整理します。 (※1図ヘッダシート参照)
2. パターン生成のためCOMP Sを呼出します。

COMP S側からは、マシン名、プログラム名などの入力要求からはじめて使用するファイル (入出力) の編成、アクセス方法、その他の指定、処理の概要に関する質問に応答します。

さきのプロセス設計の終了時には、この質問に答えられる程度に条件を整理しておいていただきたいということであり、また、パターン生成不能ということで設計結果の見直しをCOMP S側から要求することもあります。

3. パラメータの入力が終わるとCOMP Sは要件を解析し、あらかじめ作り出せる範囲でデータ部、及び階層化して展開するモジュールを制御する手続部を生成し、ソース・コーディングのリスト (※2図オートコーディング参照) を作り出します。
この階層化して展開する方法は別に説明しますが、図の表記法は階層化フロチャートと呼んでいます。上記の手続部をこの図法でプリントアウトすることもできます。
(※3図オートフロー参照)

4. ソース・コーディング中には*で囲んだコメント行があります。
この部分が処理モジュールで、ここへ実際に行なわせたい処理をコーディングして追加するとプログラムが完成します。
この処理モジュールのモジュール名とコメントのみを抜き出してリストアップしたものが Spec. List (※4図オートスペック参照) です。
これは、そのまま仕様書を書くための原稿用紙となります。
また、これを使うことで、仕様書の書式、記入順序などが機械的に統一されていきます。

5. プログラムの詳細条件は、この仕様書の枠どり（モジュール単位）にあわせて記述していくようにします。

コーディングもモジュール別に行ないます。 ユーザー・コーディングの必要なものは、ほぼ次のとおりです。

a. データ部

FD句の内容、各入出力レコードの形式（COPY句で代用可）、カウンタ、ワークエリア、定数のテーブルなど。

マッチキー、グループ・コントロールキーなどがあるときは、桁数を既製コーディングのリストとつき合わせてチェックする必要があります。

b. 手続き部

各処理モジュールの内容（実行させたい処理）

なお、入力命令は原則として**COMPS**側が順次処理をするファイルについてのみ生成します。

処理内容によって違いますが単純な処理モジュールでは、1～2行の追加ですむこともあります。複雑なモジュールの場合はモジュールをさらに下位へユーザーが展開して数百行のコーディングをすることになる場合もあります。

6. モジュール単位にコーディングされたソースコードを追加するとき、一般のライブラリ編集機能を使うと行番号指定となります。

このそう入作業を簡単にするため、順序とは無関係（モジュールの出現順でモジュール内の配列は入力順）に、モジュール名（またはワークレコード名）を指定してそう入できるエディタ機能があります。

この**COMPS**のエディタ機能は、次のライブラリからの多段COPY機能も含んでいます。

7. そう入されたステートメントの中に***COPY**ステートメントがあると**COMPS**ライブラリの中からCOPYすべきモジュール（データ記述または手続き）を探します。

このCOPY処理のさいに次のことができます。

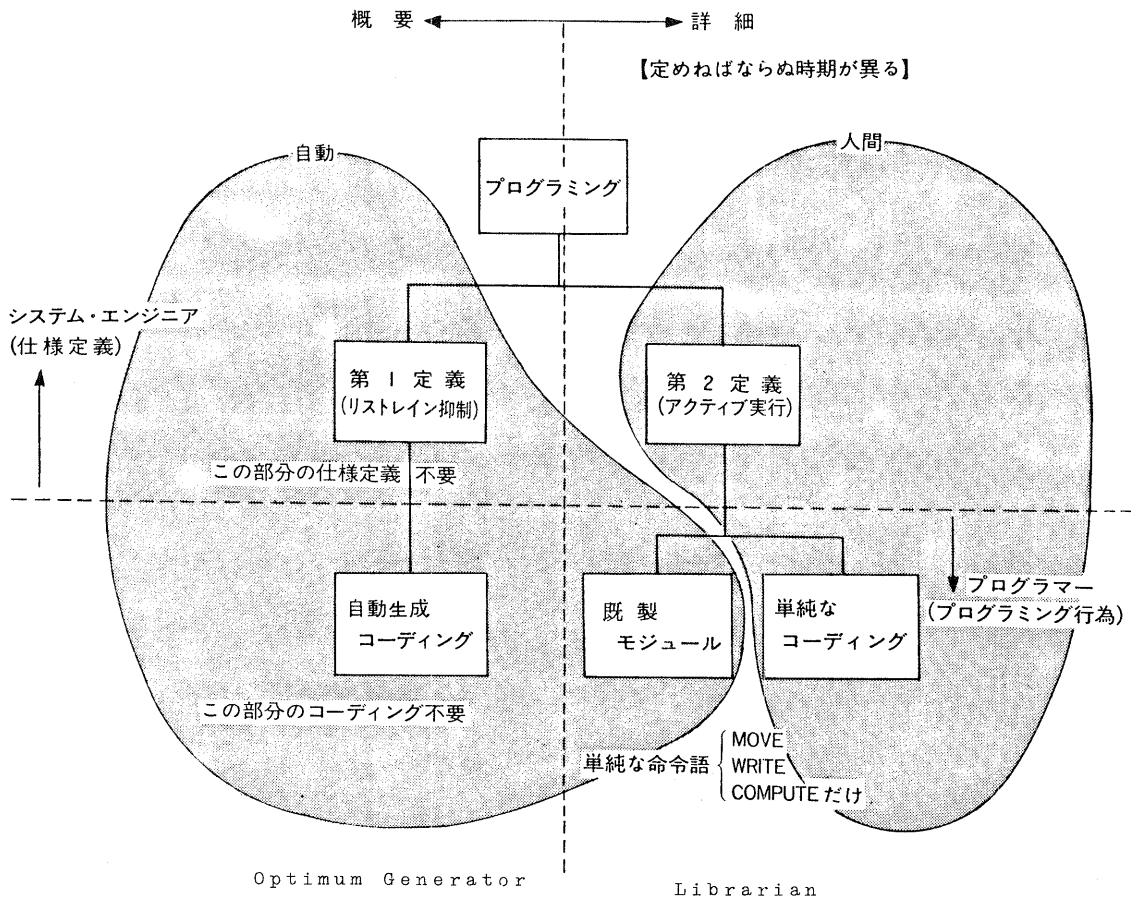
- a. データ名などに接頭語（S-など）または接尾語（-しなど）をつける。

- b. 再COPY（さらに別のモジュールをCOPY）

COPYは、**COMPS**のCOPY機能を使わなくてもかまいませんが、上記のようなことをしたければこの機能を使う必要があります。

2 コンセプト

COMP Sの自動化への接近は、プログラムを制御と処理の2面からとらえ、前者はパターン化によって最適生成を計り、後者はライブラリー化によって再利用する、ことによっています。



従ってツールとしては、Optimum Generator と Librarian の2大機能が作られております。

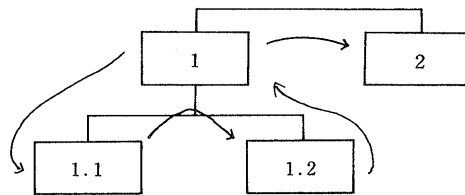
3 階層化フローチャート法

ルール 1 プログラムは全体から下位方向へ展開する。

ルール 2 下位の処理がくり返し処理となるときは、くり返し前処理、くり返し後処理を加えて展開する。またそのくり返しを制御する上位のモジュールには、くり返し打ちりの条件を付記しておく。

ルール 3 特定の条件が成立したときだけ実行する処理は、条件を付記して下位に展開する。

ルール 4 展開されたモジュールの実行順は、左から右、下位にモジュールが展開されているときは、下位の部分の左から右へ実行し、上位へ戻ることを原則とする。

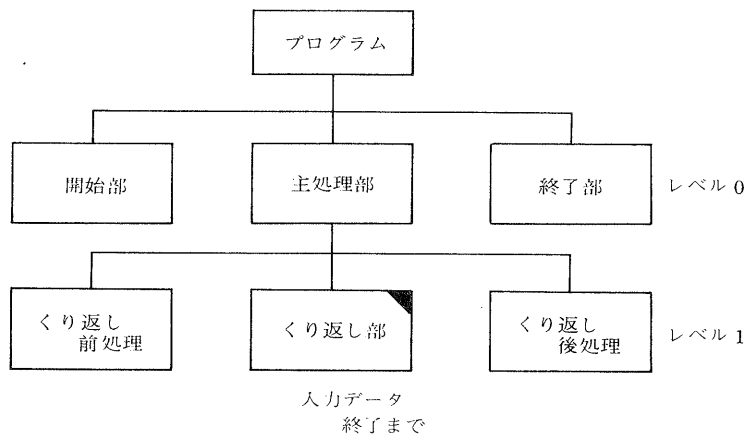


ルール 5 下位にモジュールの展開されているモジュールは制御モジュールとし原則としてこのモジュールは、実行を制御する命令のみをコーディングする。

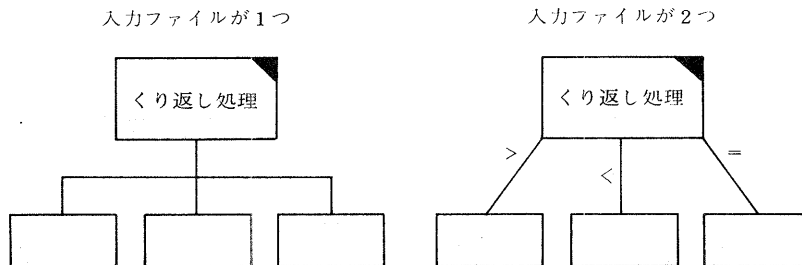
次にバッチ処理プログラムの展開は次の基準によって行なうこととする。

基準 1 プログラム全体を開始部、主処理部、終了部に分ける。

基準 2 主処理部は、データの入力が終了するまでくり返すことを打ち切り条件とする。くり返し処理部を中心に前処理、くり返し後処理に展開する。

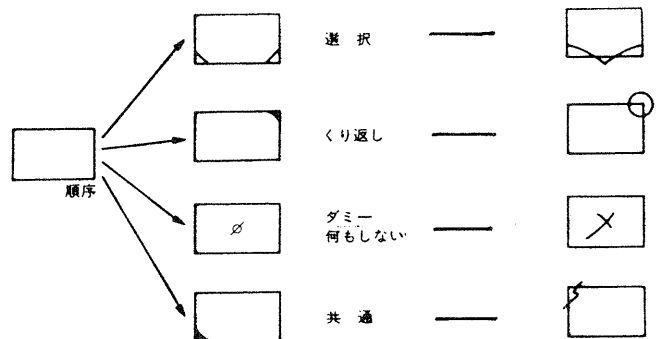


バッチ処理の場合、レベル0、レベル1の展開は常に一定で、レベル2以下は入力ファイル（順次処理をする）の数によって次のように分かれる。

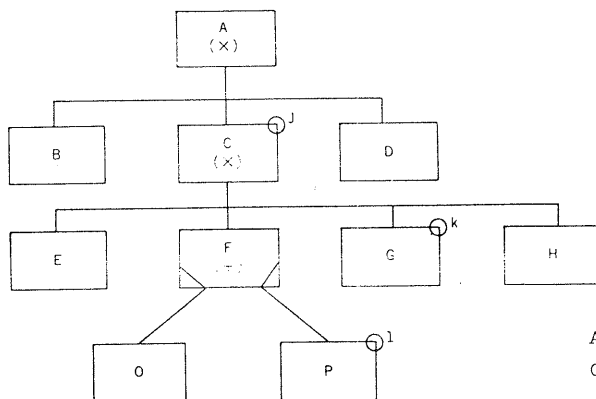


レベル3以下の展開は条件によって変化する。

モジュール記号



算 式



$$A = B \times C^j \times D \quad (1)$$

$$C = E \times F \times G^k \times H \quad (2)$$

$$F = O + P^l \quad (3)$$

(3)を(2)に代入

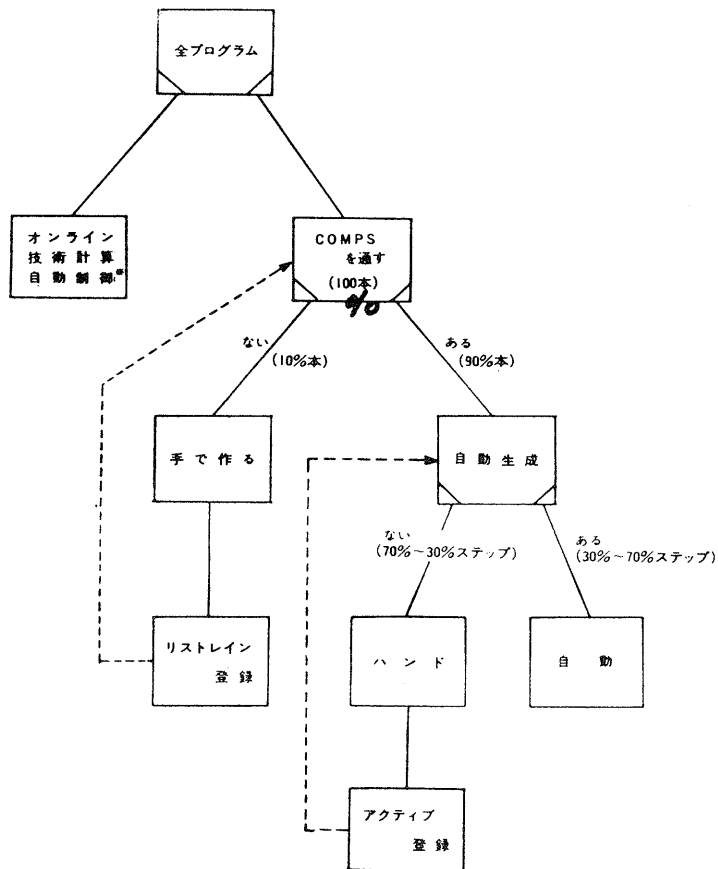
$$C = E \times (O + P^l) \times G^k \times H \quad (4)$$

(4)を(1)に代入

$$A = B \times (E \times (O + P^l) \times G^k \times H)^j \times D$$

(意味) ブランチがリストレインの項目を果たす。
リーフがアクティビティの項目を果たす。

4 自動化率



むすび

COMPSは呼名のごとく、他のプログラムをモジュールとして制御することのできる、外枠であります。すなわち既存の応用ルーチンだけでなく、すでにパッケージとしてでているプログラムを包含することもできるかもしれません。

したがって、このようなツールが発展して社会的に統合され、標準化されて、開放的なプログラミング活動になることが望まれます。

PROGRAM ID MK1320	スタートメッセージ なし	エンドメッセージ なし	特殊目的のセグメント なし	担当者氏名 高岡 元
----------------------	-----------------	----------------	------------------	---------------

JOB概要

11 郵便コード

12 回線店 229

13 回線店 ファイル

14 回線店 229

01

L1

11	12	13	14
回線店 229	回線店 ファイル	部門 ファイル	回線店 229

ページ当り最高行数: 28
 明細とサマリの別: 無
 見出しの有無: 有
 ページフッタイングの有無: 無

処理概要

- (1) 11をキーアップするまでテーブルにため込む
- (2) 11と12のマジKEYとの比較
 - ① 11 < 12の時
テーブルの内容でフォロイする
 - ② 11 > 12の時
・12のキーアップまで読みとばし (DISPLAYする)
 - ③ 11 = 12の時
・12のキーアップまでテーブルにため込み
・テーブルの内容でフォロイする

使用ファイル概要

ファイル	用途	構成	アクセス	配列キー/アクセスキー	特記事項	レーベル	BF	レコード長
11	① T R・U	⑤ IS D	⑤ D	管業所>管業員>優先順位	REPEAT:PRESET MULTI SELECT	STANDARD	13	150
12	D	S	S	管業所>管業員>優先順位	SELECT	STANDARD	25	80
13	R	S	S	X	PRESET	STANDARD	13	80
L1	D	S	S	X	X	STANDARD	12	133
14	R	IS	D	Y	Y	STANDARD	13	150

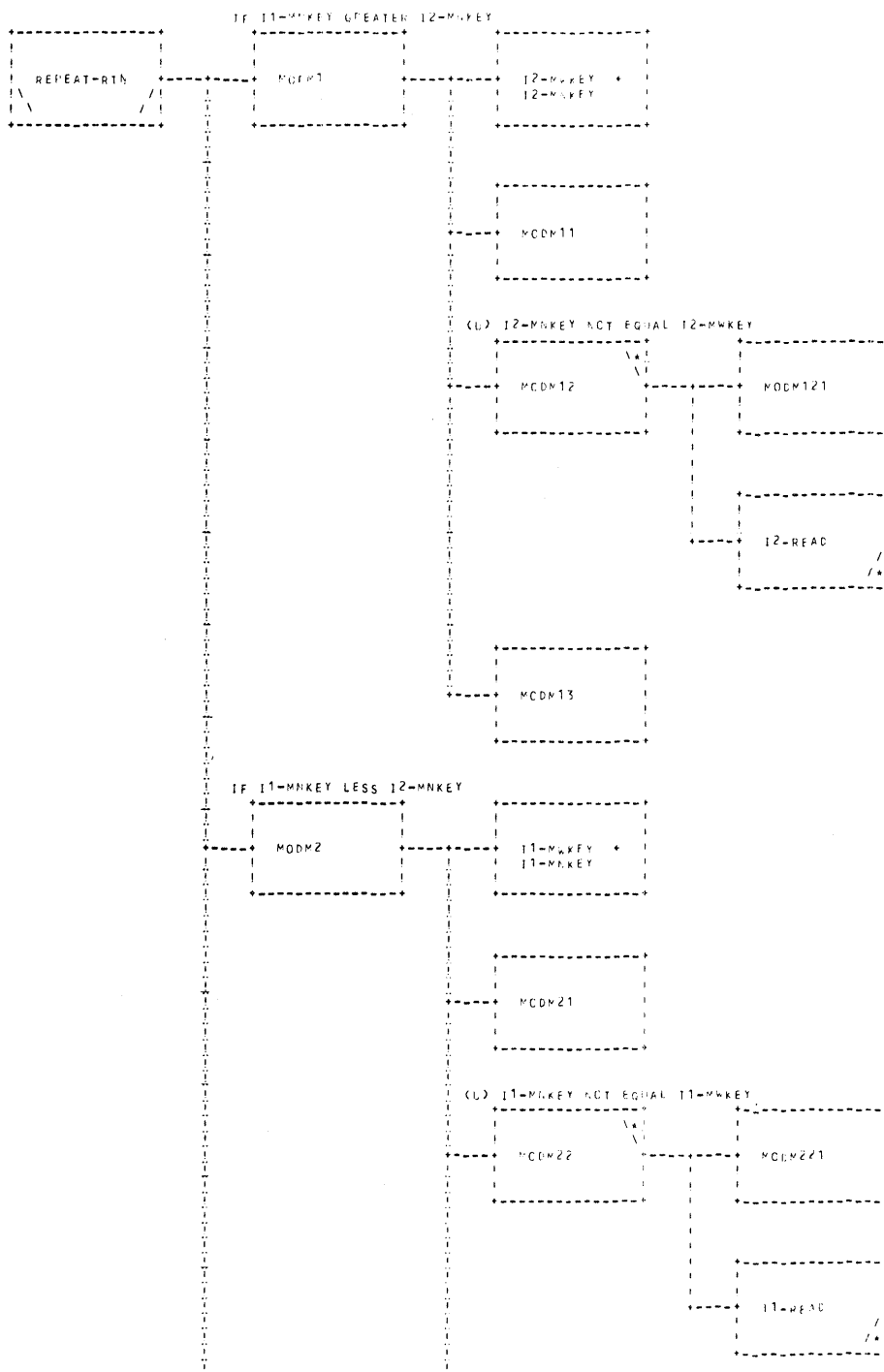
オ1図 ヘッダシート

```

105050      GO TO EX-TERM-DISPLAY.
105060 EX-TERM-DISPLAY.      EXIT.
106010*****
106020      PRE REPEAT PROCEDURES *****
106030*****
106040      (WORK-AREA-SET)   コーディングのワークエリア、ワークシートノイニシアイズ -----
106050      -----
106060 WORK-AREA-SET.
106070      GO TO EX-WORK-AREA-SET.
106090 EX-WORK-AREA-SET.      EXIT.
107010*****
107020      (CONSTANT-PRESET) システムのファイルのメモリに設定する -----
107030      -----
107040 CONSTANT-PRESET.
107050      GO TO EX-CONSTANT-PRESET.
107090 EX-CONSTANT-PRESET.    EXIT.
108010*****
108020      (USERS-DATE-SET)   システムのユーザー、プログラマー、プログラマーの -----
108030      -----
108040 USERS-DATE-SET.
108050      GO TO EX-USERS-DATE-SET.
108090 EX-USERS-DATE-SET.     EXIT.
109010*****
109020      INPUT FILE --- SUB ROUTINES ***
109030*****
110010      (11-EOFRTN)       11 ファイルの EOF のとき -----
110040      -----
110050 11-EOFRTN.
110070      MOVE HIGH-VALUE TO 11-MNKEY.
110080      GO TO EX-11-EOFRTN.
110090 EX-11-EOFRTN.          EXIT.
111010      (12-EOFRTN)       12 ファイルの EOF のとき -----
111040      -----
111050 12-EOFRTN.
111070      MOVE HIGH-VALUE TO 12-MNKEY.
111080      GO TO EX-12-EOFRTN.
111090 EX-12-EOFRTN.          EXIT.
112010      (11-KEYEDIT)      11 ファイルのコントロールのキー (マッシュアップキー) の -----
112020      -----
112040      -----
112050 11-KEYEDIT.
112060      GO TO EX-11-KEYEDIT.
112090 EX-11-KEYEDIT.         EXIT.
113010      (12-KEYEDIT)      12 ファイルのコントロールのキー (マッシュアップキー) の -----
113020      -----
113040      -----
113050 12-KEYEDIT.
113060      GO TO EX-12-KEYEDIT.
113090 EX-12-KEYEDIT.         EXIT.
114010      (11-RECORD-SELECT) 11 ファイルのレコードの SELECTION の -----
114020      -----
114030      -----
114040 11-RECORD-SELECT.
114050      GO TO EX-11-RECORD-SELECT.
114090 EX-11-RECORD-SELECT.    EXIT.

```

オ2図 オートコーディング



オ3図 オートフロー

SPECIFICATION LIST

※4図 オートスペック