

5G-8

パソコンにおけるYACⅡエディタ

林 義雄 野沢 隆
(富士通株式会社)

1. はじめに

YACⅡチャートは設計支援の構造化表記法として考案され、現在ではそれを利用したコード生成・ビジュアルテストツールが開発されている。[1][2]

YACⅡエディタはプログラム設計工程のうち、処理手順を詳細化していく過程を支援するコンポーネントとして位置付けられる。この過程は、いわば試行錯誤の過程であり、これを支援するには、木構造を意識した専用エディタが必須である。論理を率直に操作できるエディタによって、設計者は、(字下げ、レイアウトといった)余計なことに気をを使うことなく論理の組立に専念できる。

YACⅡエディタの基本要件、必要な機能については既に述べられているところである。[3]

ここでは、パーソナルコンピュータ上で実現した、YACⅡ専用エディタにおける木構造指向の編集機能という観点から以下の点について紹介する。

- 図の領域と文書の領域を区別した編集機能
- 段階的詳細化支援機能
- 再利用支援機能
- 日本語構文入力の簡略化

2. 図の領域と文書の領域を区別した編集機能

YACⅡは左側に記号による木構造、右側に構造をもたない文という具合に両者が混在しており、それぞれの編集要件が異なる。これを解決するために、図の領域と文書の領域を区別し、前者を指示されたときは木構造を意識した編集、後者を指示されたときは文書の編集というように切り分けている。(図1)

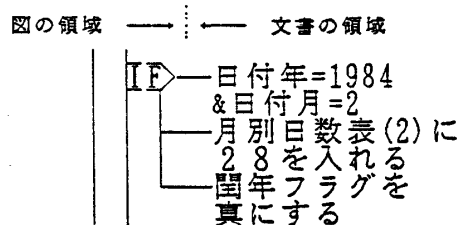


図 1 図領域と文書領域の区別

YACⅡエディタは従来のテキストエディタ同様、個々の文を編集操作の対象とするとともに、論理記号を指示することにより、それを節として部分木をなす一連の文を木構造の任意の階層に複写・移動する機能をサポートする。

この時、常に正しい構造のチャートとなるように木の整合性が自動的に保たれる(図2)。

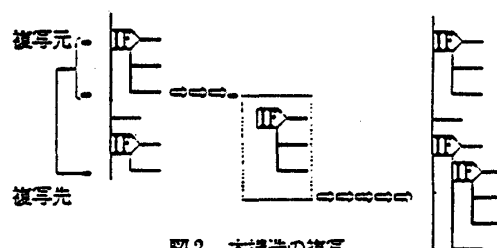


図2 木構造の複写

3. 段階的詳細化支援機能

トップダウン的に処理を詳細化していく操作を支援することでチャートのモジュール化が期待できる。

利用者は、一定の抽象レベルで概要を記述し処理全体を把握した後、個々の概要記述を詳細記述にブレークダウンするという操作が簡単にできる。(図3)

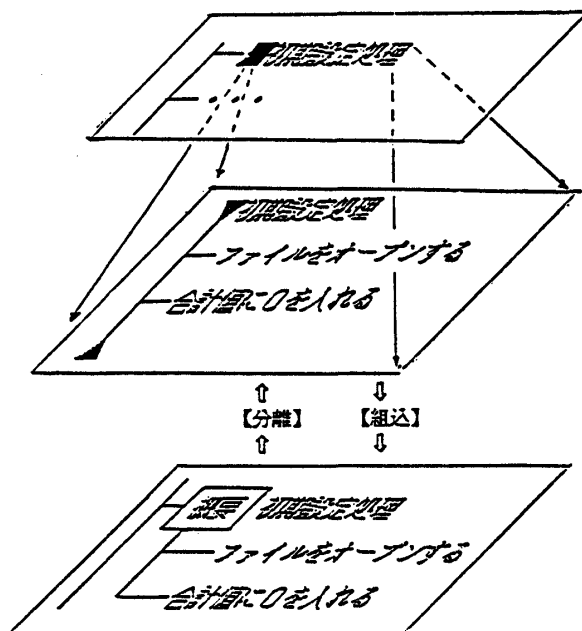


図 3 分離型と組込型

この時、概要記述と詳細記述を、分離型・組込型の2通りの形式で考慮し、両者の相互変換を容易にしている。

① 分離型……概要と詳細を分離して記述する

詳細記述が長くなる時、又は、その処理ブロックが共通性を持ち、プログラムの複数個所で参照される可能性がある時にはこの形式が良い。

② 組込型……概要記述の直後に組込んで記述する

詳細記述が短い時、又は、その処理がプログラムのなかで局所的である時などは、分離して記述するよりも分り易い。

この相互変換が容易にできることにより、以下のような問題が解決できる。

★ 部分的な修正を繰り返した結果、プログラム構造が望ましくなくなったとしても、大幅な修正が面倒で構造を適正に再編成しない

★ コーディング時に構造を手直しして仕様書と異なるプログラムになってしまう

4. 再利用支援機能

(1) メモ機能

YAC II エディタは、メモ領域と呼ばれる一定の作業用バッファを有しており、必要に応じてYAC II チャートの一部を一時的に格納しておくことができる。

これを随時とりだし任意の位置に展開することができる。

(2) 部品登録/展開機能

ひとつのプログラム中で頻繁に使用されるロジックや一定のプログラム構造にきまって現れるパターンなどはYAC II チャートの形式でファイルに登録しておくことができ、必要な時に任意の位置に展開し再利用することができる。(図4)

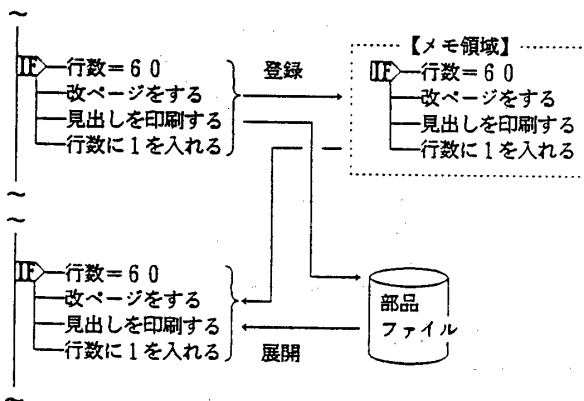


図4 メモ領域と部品ファイル

5. 日本語構文入力簡略化

YAC II チャートの文書部分は、任意の日本語文が記述できる。しかし、その表現が利用者によって異なるのは設計書の規格化・標準化という点から望ましくない。

そこで、YAC II エディタでは、しばしば使用する日本語の表現を用意し、メニューにより選択できるようにした。

利用者は、ポップアップメニュー上をマウスで指示し、基本構文をYAC II チャートの文書部分に表示させたのち、可変部分のみを入力すればよい。(図5)

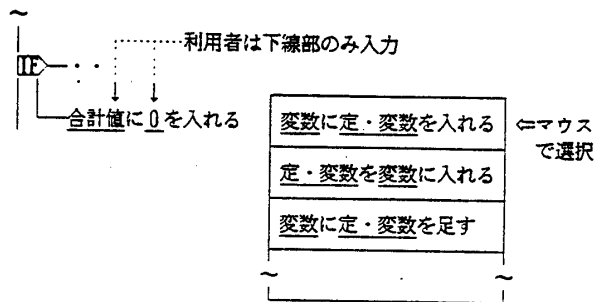


図5 日本語構文メニュー

6. おわりに

プログラム設計において、単にYAC II チャートを記述するだけであれば、「紙とエンピツ」の手軽さにはかなわないであろう。しかしプログラム設計作業は試行錯誤の過程であり、最適なロジックを組立てるまでに何回となく設計書の変更を余儀なくされる（あるいは変更せずに保守性を低下させることになる）。

YAC II エディタにより、この設計書変更過程が効率化され、保守性も向上すると考えている。さらに、以下の点について機能強化を検討していきたい。

- ① チャートレベルでのデバッグ・テスト
- ② コンパイラとの連携
- ③ ホスト計算機との連携

参考文献

- [1] 村上他：プログラム生成のためのYAC II 表記法のあり方、
情報処理学会第31回全国大会予稿集
- [2] 藤井他：YAC II 仕様書をベースとしたソフトウェア開発の自動化とその評価、
情報処理学会第31回全国大会予稿集
- [3] 富田：ワークステーションにおけるYAC II エディタ、
情報処理学会第32回全国大会予稿集